Attributes-Guided and Pure-Visual Attention Alignment for Few-Shot Recognition

Siteng Huang,[1,2] **Min Zhang,**[2] **Yachen Kang,**[2] **Donglin Wang**[2*]
[1] Zhejiang University
[2] Machine Intelligence Lab (MiLAB), AI Division, School of Engineering, Westlake University
{huangsiteng, zhangmin, kangyachen, wangdonglin}@westlake.edu.cn

Abstract

The purpose of few-shot recognition is to recognize novel categories with a limited number of labeled examples in each class. To encourage learning from a supplementary view, recent approaches have introduced auxiliary semantic modalities into effective metric-learning frameworks that aim to learn a feature similarity between training samples (support set) and test samples (query set). However, these approaches only augment the representations of samples with available semantics while ignoring the query set, which loses the potential for the improvement and may lead to a shift between the modalities combination and the pure-visual representation. In this paper, we devise an attributes-guided attention module (AGAM) to utilize human-annotated attributes and learn more discriminative features. This plug-and-play module enables visual contents and corresponding attributes to collectively focus on important channels and regions for the support set. And the feature selection is also achieved for query set with only visual information while the attributes are not available. Therefore, representations from both sets are improved in a fine-grained manner. Moreover, an attention alignment mechanism is proposed to distill knowledge from the guidance of attributes to the pure-visual branch for samples without attributes. Extensive experiments and analysis show that our proposed module can significantly improve simple metric-based approaches to achieve state-of-the-art performance on different datasets and settings.

Introduction

The recent success of visual recognition tasks commonly relies on supervised learning from a large number of labeled samples. However, in many practical applications, it is expensive and time-consuming to collect sufficient labeled samples for each category. Inspired by the fact that humans are good at learning to identify objects with very little direct supervision, *few-shot learning* (FSL) has attracted considerable attention. Trained on sufficient labeled samples from known categories (*seen classes*) and given very few labeled samples (*support set*) of a set of new categories (*unseen classes*), few-shot recognition methods aim at classifying unlabeled samples (*query set*) into these new categories. To imitate the process of learning new concepts, seen and unseen classes do not overlap, which makes classical deep learning methods to have generalization issues. Meanwhile, only very few labeled samples are available for the test unseen classes, which may cause severe overfitting when trying common fine-tuning strategies.

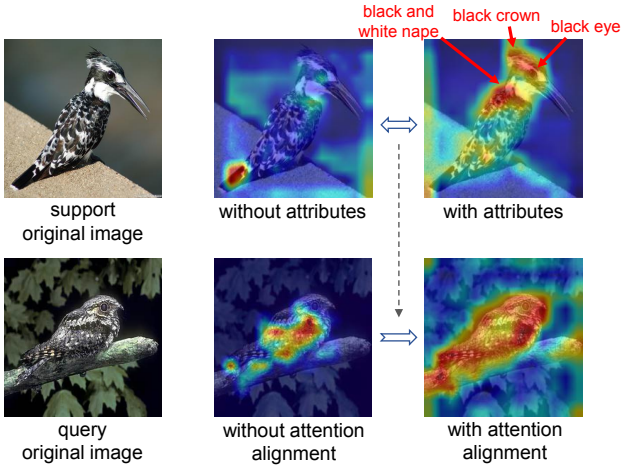

Figure 1: An illustration of the effect of our proposed attention alignment mechanism. The network learns to focus more on discriminative features of support samples with the guidance of auxiliary attributes. And the attention alignment mechanism helps the self-guided branch to learn to select important features without attributes.

An effective approach to the few-shot recognition problem is to train a neural network to embed support and query samples into a smaller embedding space, where categories can distinguish with each other based on a distance metric (Vinyals et al. 2016; Sung et al. 2018). Existing works have achieved promising results by improving the informativeness and discriminability of the learned representations. Ulteriorly, inspired by the hypothesis that language helps infants to learn to recognize new visual objects (Jackendoff 1987), some recent approaches introduce auxiliary semantic modalities such as label embeddings (Xing et al. 2019) and attribute annotations (Tokmakov, Wang, and Hebert 2019) to compensate for the lack of supervision. These approaches assume that auxiliary semantic information is only avail-

*Corresponding author.

able for support set, but not for query set that is regarded as the prediction object. However, while following this realistic setting, these approaches only focus on the learning of support representations via information mixture or constraint with the help of semantics. The necessity of explicitly designing special mechanisms for query samples has been ignored, resulting in a potential loss of performance. Moreover, as visual and semantic feature spaces naturally have heterogeneous structures, query representations directly obtained from visual contents may shift from same-labeled support representations mixed of both visual and semantic modalities. This is shown as the failure of increasing the intra-class similarity and reducing the inter-class similarity, which damages the accuracy of recognition.

In this paper, we propose a novel **attributes-guided attention module (AGAM)** to utilize human-annotated attributes as auxiliary semantics and learn more discriminative features. AGAM contains two parallel branches, *i.e.*, the **attributes-guided branch** and the **self-guided branch**. Each branch sequentially applies two attention modules, first a channel-wise attention module to blend cross-channel information and learn which channels to focus, then a spatial-wise attention module to learn which areas to focus. The difference between the two branches is that corresponding attributes of support samples can guide the feature selection in the attributes-guided branch, leading to more representative and discriminative representations due to the prominence of relevant elements and noise reduction of irrelevant clutters. And the self-guided branch also helps to refine the pure-visual representations of samples when attributes are not available. Different from existing modality mixture approaches (Xing et al. 2019; Schwartz et al. 2019) that directly mix multiple modalities with an adaptive proportion, we use the attention mechanism to enhance the informativeness of representations more finely, while ensuring the support representations modified with attributes live in the same space of pure-visual query representations.

Although query representations output by the self-guided branch go through a similar process to support ones, the lack of semantic information may lead to an inaccurate focus on important channels or regions, which increases the distance between same-labeled support and query samples. To handle the issue, we propose an **attention alignment mechanism** for AGAM, which aligns the attention weights from both branches with a specially-designed attention alignment loss during the learning of support representations. As the features to be emphasized or suppressed by the two branches tend to be similar, the alignment can be regarded as a special case of knowledge distillation (Hinton, Vinyals, and Dean 2015), which means the branch with less information can learn from the branch with more information. Therefore, as shown in Figure 1, the self-guided branch can better locate informative features without the guidance of attributes. Note that our AGAM can be viewed as a plug-and-play module, making existing metric-learning approaches more effective. To summarize, our main contributions are in several folds:

1. We utilize powerful channel-wise and spatial-wise attention to learn what information to emphasize or suppress. While considerably improving the representativeness and discriminability of representations in a fine-grained manner, features extracted by both visual contents and corresponding attributes share the same space with pure-visual features.

2. We propose an attention alignment mechanism between the attributes-guided and self-guided branches. The mechanism contributes to learning the query representations by matching the focus of two branches, so that the supervision signal from the attributes-guided branch promotes the self-guided branch to concentrate on more important features even without attributes.

3. We conduct extensive experiments to demonstrate that the performance of various metric-based methods is greatly improved by plugging our light-weight module.

Related Work

Few-Shot Recognition

Few-shot recognition aims to learn to classify unseen data examples into a set of new categories given only a few labeled samples. Having made significant progress, most meta-learning approaches can be roughly divided into two categories. The first is *optimization-based methods*, which learn a meta-learner to adjust the optimization algorithm so that the model can be good at learning with a few examples, usually by providing the search steps (Ravi and Larochelle 2017) or a good initialization to begin the search (Finn, Abbeel, and Levine 2017; Nichol, Achiam, and Schulman 2018). The second is *metric-based methods* (Vinyals et al. 2016; Snell, Swersky, and Zemel 2017; Sung et al. 2018; Oreshkin, López, and Lacoste 2018), which learn a generalizable embedding model to transform all instances into a common metric space, and in this metric space, simple classifiers can be executed directly.

Learning with Semantic Modalities

With the rapid growth of multimedia data, multimodal analysis has attracted a lot of attention in recent years. In particular, zero-shot learning methods use various semantic modalities to recognize unseen classes without any available labeled samples (Reed et al. 2016; Xian et al. 2019). The common practice in zero-shot learning is to train a projection between visual and semantic feature spaces with labeled samples in seen classes, and apply the learned projection to unseen classes when inferring. Although the setting of zero-shot learning seems similar to that of few-shot learning, simply fine-tuning zero-shot methods with few samples in few-shot problems may lead to overfitting.

Recently, building upon existing metric-based meta-learning methods, some few-shot learning works propose to utilize auxiliary semantic modalities in a quite different manner from zero-shot learning. (Chen et al. 2019b) maps samples into a concept space and synthesizes instance features by interpolating among the concepts. (Tokmakov, Wang, and Hebert 2019) proposes a simple attribute-based regularization approach to learn compositional image representations. (Xing et al. 2019) models the representation as a convex combination of the two modalities. And (Schwartz et al. 2019) proposes a benchmark for few-shot learning with multiple semantics. In our work, with the help of attributes

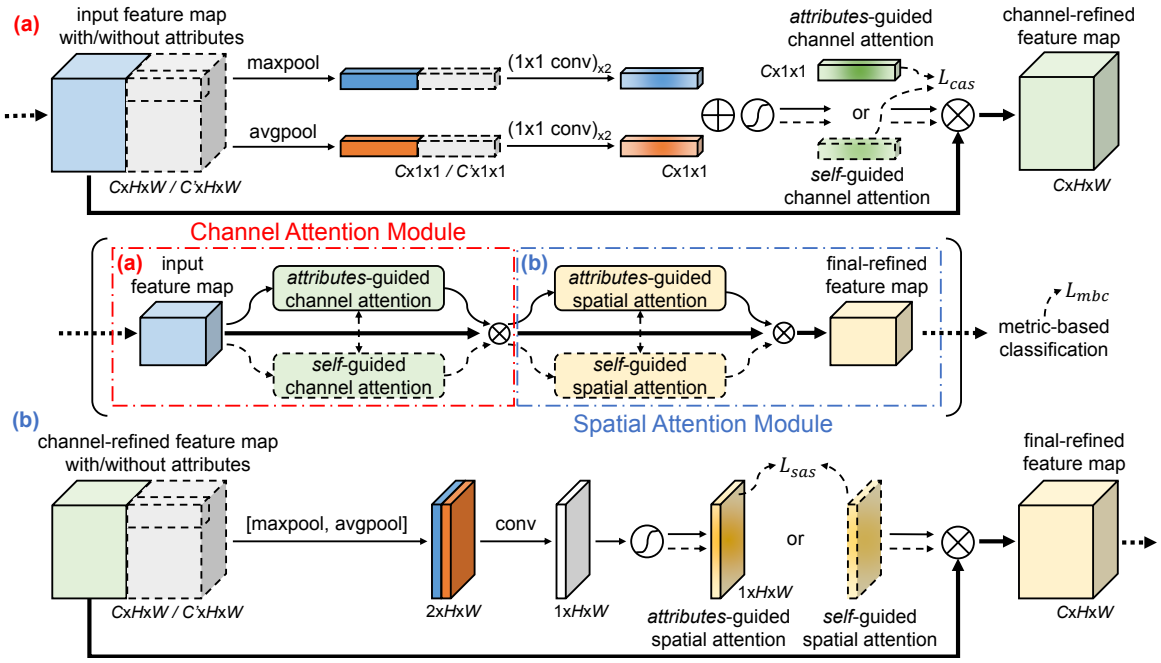

Figure 2: The overall framework of AGAM. Based on whether attributes to the image are available, one of the attributes-guided branch and the self-guided branch is selected. The input features sequentially pass a channel-wise attention module (a) and a spatial-wise attention module (b) to obtain the final-refined features.

as the only semantic modality, we utilize channel-wise and spatial-wise attention to learn a better metric space in a fine-grained manner. Furthermore, we design an attention alignment mechanism to align the focus of the attributes-guided and self-guided branches, helping to reduce mismatches of same-labeled query and support samples.

Methodology

Preliminaries

As only a few labeled samples are available in each unseen class, all approaches in our experiments follow the *episodic training* paradigm, which has been demonstrated as an effective approach for few-shot recognition (Snell, Swersky, and Zemel 2017; Sung et al. 2018). In general, models are trained on K-shot N-way episodes, and each episode can be seen as an independent task. An episode is created by first randomly sampling N categories from seen classes and then randomly sampling support and query samples from these categories. Our method hypothesizes that both visual contents and attributes as semantic information can be useful for few-shot learning. Therefore, the support set $\mathcal{S} = \{(s_i, a_i, y_i)\}_{i=1}^{N\times K}$ contains K labeled examples for each of the N categories. Here, s_i is the i-th image, a_i denotes the attributes vector to the image, and $y_i \in \{1, \ldots, N\}$ denotes the class label to the image. However, the attributes for query samples are considered to be unavailable, and the query set $\mathcal{Q} = \{(q_i, y_i)\}_{i=1}^{Q}$. Here, q_i is the i-th image, and Q denotes the number of query samples. The training phase aims to minimize the loss of the prediction in the query set for each episode, and the performance of the method is measured by the prediction accuracy of new episodes sampling from unseen classes. Note that attributes are not used in some of the experimental comparison approaches.

Algorithm Overview

In this work, we resort to metric-based methods to obtain proper feature representations for support and query samples, and propose an **attributes-guided attention module (AGAM)** to modify the features by taking into account the attribute annotations to the images. Figure 2 presents an overview of our proposed AGAM. Inspired by (Woo et al. 2018), we utilize channel-wise attention and spatial-wise attention modules to obtain the final refined features. However, different from the previous work, we design two parallel branches, *i.e.*, **attributes-guided branch** (denoted by ag) and **self-guided branch** (denoted by sg). For samples with attributes annotations, the attributes-guided branch learns the attention weights by incorporating both attributes and visual contents. And the self-guided branch is designed for the inference of samples without the guidance of attributes. Furthermore, we propose an **attention alignment mechanism** in AGAM, which aims to pull the focus of the two branches closer, so that the self-guided branch can capture more informative features for query samples without the guidance of attributes. Note that AGAM is a flexible module and can be easily added into any part of convolutional neural networks.

Channel-Wise Attention Module

Firstly, as each channel of a feature map can be considered as a feature detector (Zeiler and Fergus 2014), we produce a

1D channel-wise attention map to focus on "what" is meaningful in the given image, as shown in Figure 2(a). Given an intermediate feature map $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ output by an established convolutional backbone network, based on whether the attributes vector $\mathbf{a} \in \mathbb{R}^{D}$ corresponding to the original image is available, the input of the channel-wise attention module can be different. As the attributes vector is not available in the self-guided branch, the input $\mathbf{F}^{sg}_{c\_inp}$ is the same as $\mathbf{F}$, where c_inp denotes the *inp*ut of the *c*hannel attention module. And for samples with attributes, we firstly broadcast $\mathbf{a}$ along height and width dimension of $\mathbf{F}$ to obtain a tensor $\mathbf{A} \in \mathbb{R}^{D\times H\times W}$, then concatenate $\mathbf{F}$ and $\mathbf{A}$ on the channel dimension to get the input of the attributes-guided branch $\mathbf{F}^{ag}_{c\_inp} = [\mathbf{F}; \mathbf{A}] \in \mathbb{R}^{C^{'}\times H\times W}$, where $C^{'} = C + D$ and $[;]$ denotes the concatenation.

To compute the channel-wise attention efficiently, max-pooling and average-pooling are first used in parallel to squeeze the spatial dimension of the input feature. As shown in the later ablation study, using both pooling strategies simultaneously can bring complementary and distinctive features. Here we have $\text{MaxPool}(\mathbf{F}^{sg}_{c\_inp}), \text{AvgPool}(\mathbf{F}^{sg}_{c\_inp}) \in \mathbb{R}^{C\times 1\times 1}$, and $\text{MaxPool}(\mathbf{F}^{ag}_{c\_inp}), \text{AvgPool}(\mathbf{F}^{ag}_{c\_inp}) \in \mathbb{R}^{C^{'}\times 1\times 1}$. For each branch, features pooled by each pooling layer are then forwarded to an attention generating network, which consists of two convolutions with kernel size 1 and can also be seen as two linear transformations with a ReLU activation in between (Lin, Chen, and Yan 2014). The purpose of this attention generating network is to generate channel-wise attention after exploiting the inter-channel relationship of features, and note that parameters of this network are not shared between two branches. The element-wise summation is used to merge the results of the same branch. In short, we have

$$\begin{aligned}\mathbf{M}^{ag}_{c} =& \sigma(\mathbf{W}^{ag}_{1}(\mathbf{W}^{ag}_{0}(\text{MaxPool}(\mathbf{F}^{ag}_{c\_inp})))\\ &+ \mathbf{W}^{ag}_{1}(\mathbf{W}^{ag}_{0}(\text{AvgPool}(\mathbf{F}^{ag}_{c\_inp})))),\end{aligned} \tag{1}$$

$$\begin{aligned}\mathbf{M}^{sg}_{c} =& \sigma(\mathbf{W}^{sg}_{1}(\mathbf{W}^{sg}_{0}(\text{MaxPool}(\mathbf{F}^{sg}_{c\_inp})))\\ &+ \mathbf{W}^{sg}_{1}(\mathbf{W}^{sg}_{0}(\text{AvgPool}(\mathbf{F}^{sg}_{c\_inp})))),\end{aligned} \tag{2}$$

where σ denotes the sigmoid activation function, $\mathbf{W}^{ag}_{0} \in \mathbb{R}^{(C^{'}/r)\times C^{'}}$, $\mathbf{W}^{ag}_{1} \in \mathbb{R}^{C\times (C^{'}/r)}$, $\mathbf{W}^{sg}_{0} \in \mathbb{R}^{(C/r)\times C}$, $\mathbf{W}^{sg}_{1} \in \mathbb{R}^{C\times (C/r)}$ are parameters of convolutions, and r is a reduction ratio to reduce parameter overhead. Note that the ReLU activation followed by $\mathbf{W}_0$ is omitted for clearer expression. To obtain the channel-refined features, we multiply $\mathbf{M}^{ag}_{c}, \mathbf{M}^{sg}_{c} \in \mathbb{R}^{C\times 1\times 1}$ with the feature map $\mathbf{F}$, expressed as

$$\mathbf{F}^{ag}_{c\_out} = \mathbf{M}^{ag}_{c} \otimes \mathbf{F}, \quad \mathbf{F}^{sg}_{c\_out} = \mathbf{M}^{sg}_{c} \otimes \mathbf{F}, \tag{3}$$

where $\mathbf{F}_{c\_out} \in \mathbb{R}^{C\times H\times W}$ represents the output of the channel-wise attention module in the corresponding branch, and $\otimes$ denotes element-wise multiplication. During multiplication, the channel-wise attention values are broadcasted along the spatial dimension.

Spatial-Wise Attention Module

As illustrated in Figure 2(b), we also generate a 2D spatial-wise attention map to focus "where" is an informative region. The input of the module is $\mathbf{F}^{sg}_{s\_inp} = \mathbf{F}^{sg}_{c\_out} \in \mathbb{R}^{C\times H\times W}$ for the self-guided branch, and $\mathbf{F}^{ag}_{s\_inp} = [\mathbf{F}^{ag}_{c\_out}; \mathbf{A}] \in \mathbb{R}^{C^{'}\times H\times W}$ for the attributes-guided branch. For both two branches, we first apply max-pooling and average-pooling operations along the channel dimension and concatenate the pooled features. Then for each branch, a convolution layer is used to generate the spatial-wise attention map. In short, the attention map is computed as

$$\mathbf{M}^{ag}_{s} = \sigma(f^{ag}(\big[\text{AvgPool}(\mathbf{F}^{ag}_{s\_inp}); \text{MaxPool}(\mathbf{F}^{ag}_{s\_inp})\big])), \tag{4}$$

$$\mathbf{M}^{sg}_{s} = \sigma(f^{sg}(\big[\text{AvgPool}(\mathbf{F}^{sg}_{s\_inp}); \text{MaxPool}(\mathbf{F}^{sg}_{s\_inp})\big])), \tag{5}$$

where σ denotes the sigmoid activation function. f represents a convolution operation with the filter size of 7×7 and the number of zero-paddings on both sides of 3, whose parameters are also not shared between the two branches. To obtain the final refined features, we multiply $\mathbf{M}^{ag}_{s}, \mathbf{M}^{sg}_{s} \in \mathbb{R}^{1\times H\times W}$ with the channel-refined features in the corresponding branch, which can be expressed briefly as

$$\mathbf{F}^{ag}_{s\_out} = \mathbf{M}^{ag}_{s} \otimes \mathbf{F}^{ag}_{c\_out}, \quad \mathbf{F}^{sg}_{s\_out} = \mathbf{M}^{sg}_{s} \otimes \mathbf{F}^{sg}_{c\_out}, \tag{6}$$

where $\mathbf{F}_{s\_out} \in \mathbb{R}^{C\times H\times W}$ represents the output of the corresponding branch. During multiplication, we broadcast the spatial-wise attention values along the channel dimension.

Attention Alignment Mechanism

As AGAM works with other metric-learning approaches, these improved feature embeddings are finally fed into a metric-based learner. For a K-shot N-way episode containing Q query samples, the metric-based classification loss can be defined as the negative log-probability according to the true class label $y_n \in \{1, 2, \ldots, N\}$:

$$L_{mbc} = -\sum_{b=1}^{Q} \log p(y = y_n | v^{q}_{b}), \tag{7}$$

where v^{q}_{b} denotes the feature embedding of the b-th query sample. Note that $p(y = y_n | v^{q}_{b})$ is the probability of predicting v^{q}_{b} as the n-th class and can be different in various metric-learning approaches, hence, the specific representation of probability depends on the chosen approach.

Furthermore, as the lack of attributes annotations may lead the self-guided branch to concentrate on suboptimal features, the metric-based learner is likely to make wrong predictions for query images as the located channels and regions are shifted from those of the same-labeled support samples. Therefore, to encourage the self-guided branch to learn to emphasize or suppress the same features as if attributes have participated in learning, we design an attention alignment mechanism between the two branches. This

Method	CUB		SUN	
	5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
MatchingNet (Vinyals et al. 2016), *paper*	61.16 ± 0.89	72.86 ± 0.70	-	-
MatchingNet (Vinyals et al. 2016), *our implementation*	62.82 ± 0.36	73.22 ± 0.23	55.72 ± 0.40	76.59 ± 0.21
MatchingNet (Vinyals et al. 2016) **with AGAM**	**71.58 ± 0.30**	**75.46 ± 0.28**	**64.95 ± 0.35**	**79.06 ± 0.19**
	+8.76	*+2.24*	*+9.23*	*+2.47*
ProtoNet (Snell, Swersky, and Zemel 2017), *paper*	51.31 ± 0.91	70.77 ± 0.69	-	-
ProtoNet (Snell, Swersky, and Zemel 2017), *our implementation*	53.01 ± 0.34	71.91 ± 0.22	57.76 ± 0.29	79.27 ± 0.19
ProtoNet (Snell, Swersky, and Zemel 2017) **with AGAM**	**75.87 ± 0.29**	**81.66 ± 0.25**	**65.15 ± 0.31**	**80.08 ± 0.21**
	+22.86	*+9.75*	*+7.39*	*+0.81*
RelationNet (Sung et al. 2018), *paper*	62.45 ± 0.98	76.11 ± 0.69	-	-
RelationNet (Sung et al. 2018), *our implementation*	58.62 ± 0.37	78.98 ± 0.24	49.58 ± 0.35	76.21 ± 0.19
RelationNet (Sung et al. 2018) **with AGAM**	**66.98 ± 0.31**	**80.33 ± 0.40**	**59.05 ± 0.32**	**77.52 ± 0.18**
	+8.36	*+1.35*	*+9.47*	*+1.31*

Table 1: Average accuracy (%) comparison with 95% confidence intervals before and after incorporating AGAM into existing methods using a Conv-4 backbone. Best results are displayed in boldface, and improvements are displayed in italics.

is achieved by applying an attention alignment loss to the same type of attention maps obtained from the same support sample but different branches. Among various types of losses that can be used to measure the similarity of attention maps, we choose a soft margin loss according to the experimental results. Specifically, the attention alignment loss of the i-th sample can be expressed as

$$l_i^{cas} = \sum_j \log(1 + \exp(-\widetilde{\mathbf{M}}_c^{ag}(j) \otimes \widetilde{\mathbf{M}}_c^{sg}(j))), \quad (8)$$

$$l_i^{sas} = \sum_j \log(1 + \exp(-\widetilde{\mathbf{M}}_s^{ag}(j) \otimes \widetilde{\mathbf{M}}_s^{sg}(j))), \quad (9)$$

where $\widetilde{\mathbf{M}}$ indicates that the attention map is normalized, and (j) denotes the j-th element of the attention map. For each episode, all support samples are taken into account:

$$L_{cas} = \sum_i^{N*K} l_i^{cas}, \quad L_{sas} = \sum_i^{N*K} l_i^{sas}. \quad (10)$$

It is noted that our attention alignment mechanism can be regarded as a special case of knowledge distillation (Hinton, Vinyals, and Dean 2015), where attention maps (referred to the "knowledge") of the attributes-guided branch (viewed as a teacher model) become the distillation targets for the self-guided branch (viewed as a student model). By mimicking the focusing behaviors of the attributes-guided branch, the self-guided branch with only unimodal input is expected to concentrate on more informative features.

Accordingly, the overall loss of each episode is defined as $L = L_{mbc} + \alpha L_{cas} + \beta L_{sas}$, where α, β are the trade-off hyperparameters to balance the effects of different losses. The time complexity of AGAM is $O(C^{'}HW)$, and the space complexity is $O(C^{'2})$. As the complexities vary with the size of the input features, we note that in our experiments, AGAM is inserted after the last convolutional layer of the backbone network to avoid excessive cost.

Experiments

Experimental Setup

Datasets. We use two datasets with high-quality attribute annotations to conduct experiments: Caltech-UCSD-Birds 200-2011 (CUB) (Wah et al. 2011) and SUN Attribute Database (SUN) (Patterson et al. 2014).

Experimental Settings. We experiment with our approach on 5-way 1-shot and 5-way 5-shot settings, and in each episode, 15 query samples per class are used for both training and inference. We report the *average accuracy* (%) and the corresponding 95% *confidence interval* over the 10,000 episodes randomly sampled from the test set.

Implementation Details. Our method is trained from scratch and uses the Adam (Kingma and Ba 2015) optimizer with an initial learning rate 10^{-3}. Following the settings of (Chen et al. 2019a), we apply standard data augmentation including random crop, left-right flip, and color jitter in the meta-training stage. And for meta-learning methods, we train 60,000 episodes for 1-shot and 40,000 episodes for 5-shot settings. For AGAM, we set trade-off hyperparameters $\alpha = 1.0$ and $\beta = 0.1$ for all experiments. Code is available at https://github.com/bighuang624/AGAM.

Adapting AGAM into Existing Frameworks

To verify the effectiveness of our proposed AGAM, we embed it into three metric-based meta-learning approaches: Matching Network (Vinyals et al. 2016), Prototypical Network (Snell, Swersky, and Zemel 2017), and Relation Network (Sung et al. 2018). Table 1 shows the gains obtained by incorporating AGAM into each approach on two datasets, and for all three approaches, incorporating AGAM leads to a significant improvement. An observation is that AGAM boosts the performance of Prototypical Network nearly by 23% on the 1-shot setting and 10% on the 5-shot setting of CUB, which is especially prominent when compared with other metric-based methods. We believe the reason is that when identifying bird species in such a fine-grained dataset as CUB, a very detailed comparison between the support and the query sample is required. While Matching Network and

Method	Backbone	Test Accuracy	
		5-way 1-shot	5-way 5-shot
CUB			
MatchingNet (Vinyals et al. 2016)	Conv-4	61.16 ± 0.89	72.86 ± 0.70
ProtoNet (Snell, Swersky, and Zemel 2017)	Conv-4	51.31 ± 0.91	70.77 ± 0.69
RelationNet (Sung et al. 2018)	Conv-4	62.45 ± 0.98	76.11 ± 0.69
MACO (Hilliard et al. 2018)	Conv-4	60.76	74.96
MAML (Finn, Abbeel, and Levine 2017)	Conv-4	55.92 ± 0.95	72.09 ± 0.76
Baseline (Chen et al. 2019a)	Conv-4	47.12 ± 0.74	64.16 ± 0.71
Baseline++ (Chen et al. 2019a)	Conv-4	60.53 ± 0.83	79.34 ± 0.61
Comp. (Tokmakov, Wang, and Hebert 2019) *	ResNet-10	53.6	74.6
AM3 (Xing et al. 2019) † *	Conv-4	73.78 ± 0.28	81.39 ± 0.26
AGAM (OURS) *	Conv-4	$\mathbf{75.87 \pm 0.29}$	$\mathbf{81.66 \pm 0.25}$
MatchingNet (Vinyals et al. 2016) †	ResNet-12	60.96 ± 0.35	77.31 ± 0.25
ProtoNet (Snell, Swersky, and Zemel 2017)	ResNet-12	68.8	76.4
RelationNet (Sung et al. 2018) †	ResNet-12	60.21 ± 0.35	80.18 ± 0.25
TADAM (Oreshkin, López, and Lacoste 2018)	ResNet-12	69.2	78.6
FEAT (Ye et al. 2020)	ResNet-12	68.87 ± 0.22	82.90 ± 0.15
MAML (Finn, Abbeel, and Levine 2017)	ResNet-18	69.96 ± 1.01	82.70 ± 0.65
Baseline (Chen et al. 2019a)	ResNet-18	65.51 ± 0.87	82.85 ± 0.55
Baseline++ (Chen et al. 2019a)	ResNet-18	67.02 ± 0.90	83.58 ± 0.54
Delta-encoder (Bengio et al. 2018)	ResNet-18	69.8	82.6
Dist. ensemble (Dvornik, Mairal, and Schmid 2019)	ResNet-18	68.7	83.5
SimpleShot (Wang et al. 2019)	ResNet-18	70.28	86.37
AM3 (Xing et al. 2019) *	ResNet-12	73.6	79.9
Multiple-Semantics (Schwartz et al. 2019) * ° •	DenseNet-121	76.1	82.9
Dual TriNet (Chen et al. 2019b) * °	ResNet-18	69.61 ± 0.46	84.10 ± 0.35
AGAM (OURS) *	ResNet-12	$\mathbf{79.58 \pm 0.25}$	$\mathbf{87.17 \pm 0.23}$
SUN			
MatchingNet (Vinyals et al. 2016) †	Conv-4	55.72 ± 0.40	76.59 ± 0.21
ProtoNet (Snell, Swersky, and Zemel 2017) †	Conv-4	57.76 ± 0.29	79.27 ± 0.19
RelationNet (Sung et al. 2018) †	Conv-4	49.58 ± 0.35	76.21 ± 0.19
Comp. (Tokmakov, Wang, and Hebert 2019) *	ResNet-10	45.9	67.1
AM3 (Xing et al. 2019) † *	Conv-4	62.79 ± 0.32	79.69 ± 0.23
AGAM (OURS) *	Conv-4	$\mathbf{65.15 \pm 0.31}$	$\mathbf{80.08 \pm 0.21}$

Table 2: Average accuracy (%) comparison to state-of-the-arts with 95% confidence intervals on both CUB and SUN datasets. † denotes that it is our implementation. * denotes that it uses auxiliary attributes. ° denotes that it uses auxiliary label embeddings. • denotes that it uses auxiliary descriptions of the categories. Best results are displayed in boldface.

Relation Network benefit from inputting and analyzing each support-query pair, the original Prototypical Network separately embeds each sample and thus perform worse. However, AGAM supplements the required fine-grained information for Prototypical Network by concatenating on discriminative features, helping to better solve those challenging recognition tasks.

Comparison with State-of-the-Arts

To prove that simple metric-based methods can surpass the previous state-of-the-art performance after equipping our proposed AGAM, we report the results of our method and others on both CUB and SUN datasets. Note that we choose to display the results of Prototypical Network with our proposed AGAM as our method. For a fair comparison, we split the results achieved by all methods into two groups according to the backbones, and our AGAM uses the same or a smaller backbone network in each group.

As shown in Table 2, our proposed AGAM further improves over Prototypical Network and achieves the best performance among all approaches. It is pointed out that AGAM not only outperforms methods that only use visual contents, but also outperforms methods of utilizing auxiliary semantic information. We attribute this success to two things. The first is that AGAM uses channel-wise and spatial-wise attention to refine the representations from both support and query set in a fine-grained manner, making full use of visual contents and auxiliary attributes. The second is that the attention alignment mechanism matches the focus of two branches, which helps to alleviate the shift between pure-visual query representations and the same-labeled support representations learned with the guidance of attributes.

Ablation Study

To empirically show the effectiveness of our framework design, a careful ablation study is conducted. Specifically, we

Method	Test Accuracy	
	5-way 1-shot	5-way 5-shot
AGAM	**75.87 ± 0.29**	**81.66 ± 0.25**
AGAM_SACA	74.22 ± 0.27	79.72 ± 0.26
w/o avgpool	66.27 ± 0.29	76.58 ± 0.25
w/o maxpool	67.60 ± 0.29	77.09 ± 0.22
w/o CA	54.91 ± 0.36	80.52 ± 0.24
w/o SA	69.66 ± 0.31	76.24 ± 0.27
w/o L_{cas}	74.88 ± 0.26	77.78 ± 0.26
w/o L_{sas}	74.29 ± 0.27	77.87 ± 0.23
w/o L_{cas}&L_{sas}	75.37 ± 0.31	78.92 ± 0.27

Table 3: Ablation test results of AGAM on CUB. Average accuracies (%) with 95% confidence intervals of each model are reported. Best results are displayed in boldface.

do one of the following operations at a time: (1) Exchange the order of two attention modules. (2) Remove one of the two pooling layers in both branches. (3) Remove one of the two attention modules in both branches. (4) Remove one or both L_{cas} and L_{sas}. We evaluate all these models on the CUB dataset based on Prototypical Network with a Conv-4 backbone, and the results are shown in Table 3.

Influence of the Order of Attention Modules. We first exchange the order of channel-wise attention and spatial-wise attention in AGAM, with spatial-wise attention in the front and channel-wise attention in the back (**AGAM_SACA**). This leads to about 2% performance drops on both 1-shot and 5-shot settings. The experimental results show that the design of the module sequence is effective.

Influence of Pooling Layers. Removing either of the average-pooling (**w/o avgpool**) and the max-pooling (**w/o maxpool**) leads to 8% to 9% performance drops on the 1-shot setting and 4% to 5% performance drops on the 5-shot setting. This demonstrates that using both pooling strategies simultaneously captures more useful information.

Influence of Attention Modules. Removing the channel-wise attention (**w/o CA**) leads to about 1% performance drops on the 5-shot setting, and more than 20% drops on the 1-shot setting. This fully demonstrates the importance of channel-wise attention in our proposed AGAM when labeled data is particularly scarce. Removing the spatial-wise attention (**w/o SA**) drops the performance by 5% to 6% on both 1-shot and 5-shot settings, which attests that the spatial-wise attention brings stable and significant improvement.

Influence of Attention Alignment Loss. To prove the effectiveness of our proposed attention alignment mechanism, we also remove one or both L_{cas} and L_{sas}. An observation from the results is that removing both L_{cas} and L_{sas} (**w/o** L_{cas}&L_{sas}) obtains about 1% performance higher than removing one of them alone (**w/o** L_{cas}, **w/o** L_{sas}) on both 1-shot and 5-shot settings. We believe the reason is that when only aligning weights from the channel-wise or spatial-wise attention modules, the important features of query samples can not be consistently selected using only visual information in the other attention module, and therefore it is better to use neither L_{cas} nor L_{sas}. However, compared to the complete AGAM model, all three models still perform less than

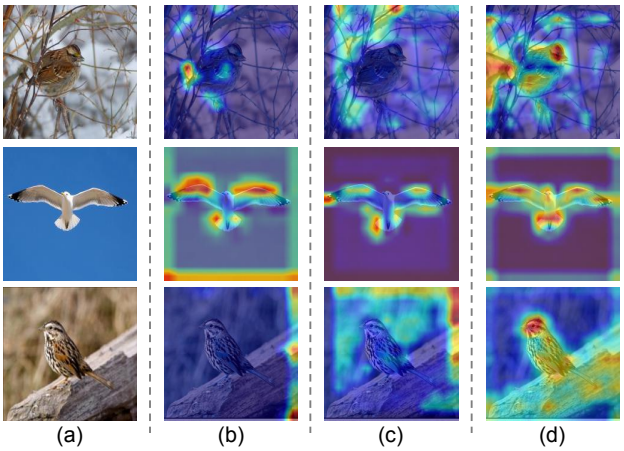

Figure 3: Gradient-weighted class activation mapping (Grad-CAM) visualization of query samples. Each row is the result of the same query sample, and each column is: (a) Original images. (b) Results of Prototypical Network. (c) Results of AGAM but removing the attention alignment mechanism. (d) Results of the complete AGAM.

1% worse on the 1-shot setting and 2% to 3% worse on the 5-shot setting, demonstrating that the joint use of two attention alignment loss items can lead to improvement.

Visualization Analysis

To qualitatively evaluate whether AGAM indeed exploits the important features with the help of the attention alignment mechanism, Figure 3 visualizes the gradient-weighted class activation maps (Selvaraju et al. 2017) from Prototypical Network, with AGAM but removing the attention alignment mechanism, and with the complete AGAM. It is observed that incorporating AGAM helps to attend to more representative local features than the original Prototypical Network, which contributes to better recognition performance. Also, the masks of AGAM-integrated model cover the representative regions better for query samples when using the attention alignment mechanism, indicating that the self-guided branch benefits from the attention alignment mechanism.

Conclusion

In this paper, we propose an attributes-guided attention module (AGAM) to fully utilize manually-encoded attributes in few-shot recognition. Owing to the channel-wise attention and spatial-wise attention, the enhanced representations are more unique and discriminative for both support and query samples. Furthermore, through the well-designed attention alignment mechanism, attention alignment is achieved between the attributes-guided branch and the self-guided branch, which narrows the gap between the representations learned with and without attributes. We have demonstrated that our proposed AGAM boosts the performance of metric-based meta-learning approaches by a large margin, which are superior to that of state-of-the-arts.

Acknowledgments

The authors gratefully acknowledge funding support from the Westlake University and Bright Dream Joint Institute for Intelligent Robotics.

References

Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R. 2018. Delta-encoder: an effective sample synthesis method for few-shot object recognition. In *Advances in Neural Information Processing Systems*.

Chen, W.; Liu, Y.; Kira, Z.; Wang, Y. F.; and Huang, J. 2019a. A Closer Look at Few-shot Classification. In *International Conference on Learning Representations*.

Chen, Z.; Fu, Y.; Zhang, Y.; Jiang, Y.; Xue, X.; and Sigal, L. 2019b. Multi-Level Semantic Feature Augmentation for One-Shot Learning. *IEEE Trans. Image Processing* 28(9): 4594–4605.

Dvornik, N.; Mairal, J.; and Schmid, C. 2019. Diversity With Cooperation: Ensemble Methods for Few-Shot Classification. In *IEEE International Conference on Computer Vision*, 3722–3730.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *International Conference on Machine Learning*, 1126–1135.

Hilliard, N.; Phillips, L.; Howland, S.; Yankov, A.; Corley, C. D.; and Hodas, N. O. 2018. Few-Shot Learning with Metric-Agnostic Conditional Embeddings. *CoRR* abs/1802.04376.

Hinton, G. E.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531.

Jackendoff, R. 1987. On beyond zebra: The relation of linguistic and visual information. *Cognition* 26(2): 89–114.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*.

Lin, M.; Chen, Q.; and Yan, S. 2014. Network In Network. In *International Conference on Learning Representations*.

Nichol, A.; Achiam, J.; and Schulman, J. 2018. On First-Order Meta-Learning Algorithms. *CoRR* abs/1803.02999.

Oreshkin, B. N.; López, P. R.; and Lacoste, A. 2018. TADAM: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, 719–729.

Patterson, G.; Xu, C.; Su, H.; and Hays, J. 2014. The SUN Attribute Database: Beyond Categories for Deeper Scene Understanding. *International Journal of Computer Vision* 108(1-2): 59–81.

Ravi, S.; and Larochelle, H. 2017. Optimization as a Model for Few-Shot Learning. In *International Conference on Learning Representations*.

Reed, S. E.; Akata, Z.; Lee, H.; and Schiele, B. 2016. Learning Deep Representations of Fine-Grained Visual Descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition*, 49–58.

Schwartz, E.; Karlinsky, L.; Feris, R. S.; Giryes, R.; and Bronstein, A. M. 2019. Baby steps towards few-shot learning with multiple semantics. *CoRR* abs/1906.01905.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *IEEE International Conference on Computer Vision*, 618–626.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical Networks for Few-shot Learning. In *Advances in Neural Information Processing Systems*, 4077–4087.

Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P. H. S.; and Hospedales, T. M. 2018. Learning to Compare: Relation Network for Few-Shot Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1199–1208.

Tokmakov, P.; Wang, Y.-X.; and Hebert, M. 2019. Learning compositional representations for few-shot recognition. In *IEEE International Conference on Computer Vision*, 6372–6381.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; and Wierstra, D. 2016. Matching Networks for One Shot Learning. In *Advances in Neural Information Processing Systems*, 3630–3638.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

Wang, Y.; Chao, W.; Weinberger, K. Q.; and van der Maaten, L. 2019. SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning. *CoRR* abs/1911.04623.

Woo, S.; Park, J.; Lee, J.; and Kweon, I. S. 2018. CBAM: Convolutional Block Attention Module. In *European Conference on Computer Vision*, 3–19.

Xian, Y.; Lampert, C. H.; Schiele, B.; and Akata, Z. 2019. Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Trans. Pattern Anal. Mach. Intell.* 41(9): 2251–2265.

Xing, C.; Rostamzadeh, N.; Oreshkin, B. N.; and Pinheiro, P. O. 2019. Adaptive Cross-Modal Few-Shot Learning. In *Advances in Neural Information Processing Systems*, 4848–4858.

Ye, H.; Hu, H.; Zhan, D.; and Sha, F. 2020. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *IEEE Conference on Computer Vision and Pattern Recognition*, 8805–8814.

Zeiler, M. D.; and Fergus, R. 2014. Visualizing and Understanding Convolutional Networks. In *European Conference on Computer Vision*, 818–833.