Rethinking Class Relations: Absolute-relative Supervised and Unsupervised Few-shot Learning

Hongguang Zhang[1,2] Piotr Koniusz[3,2] Songlei Jian[5] Hongdong Li[2] Philip H. S. Torr[4]
[1]Systems Engineering Institute, AMS [2]Australian National University [3]Data61/CSIRO
[4]University of Oxford [5]National University of Defense Technology
firstname.lastname@{anu.edu.au[2], data61.csiro.au[3],eng.ox.ac.uk[4]}

Abstract

The majority of existing few-shot learning methods describe image relations with binary labels. However, such binary relations are insufficient to teach the network complicated real-world relations, due to the lack of decision smoothness. Furthermore, current few-shot learning models capture only the similarity via relation labels, but they are not exposed to class concepts associated with objects, which is likely detrimental to the classification performance due to underutilization of the available class labels. For instance, children learn the concept of tiger *from a few of actual examples as well as from comparisons of* tiger *to other animals. Thus, we hypothesize that both similarity and class concept learning must be occurring simultaneously. With these observations at hand, we study the fundamental problem of simplistic class modeling in current few-shot learning methods. We rethink the relations between class concepts, and propose a novel Absolute-relative Learning paradigm to fully take advantage of label information to refine the image an relation representations in both supervised and unsupervised scenarios. Our proposed paradigm improves the performance of several state-of-the-art models on publicly available datasets.*

1. Introduction

Deep learning, a popular learning paradigm in computer vision, has improved the performance on numerous computer vision tasks, such as category recognition, scene understanding and action recognition. However, deep models heavily rely on large amounts of labeled training data, costly data collection and labelling.

In contrast, humans enjoy the ability to learn and memorize new complex visual concepts from very few examples. Inspired by this observation, researchers have focused on the so-called Few-shot Learning (FSL), for which a network is trained by the use of only few labeled training instances. Recently, deep networks based on relation-learning have gained the popularity [39, 37, 38, 35, 46, 47, 48, 43, 49, 22, 36]. Such approaches often apply a form of metric learning adapted to the few-shot learning task. They learn object relations (similarity learning on query and support images) based on support classes, and can be evaluated on images containing novel classes.

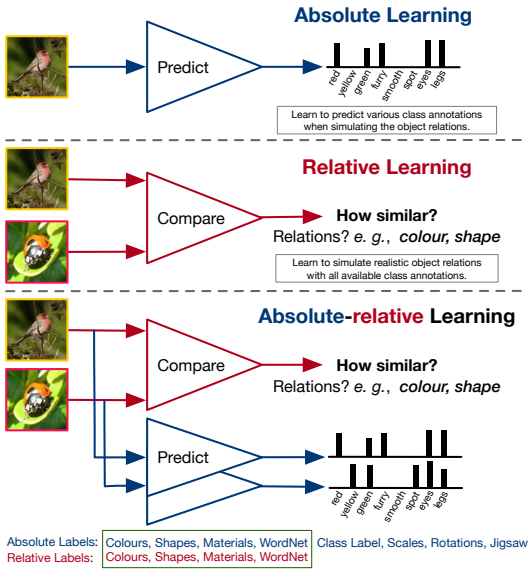

Figure 1: Our few-shot learning paradigm. Absolute Learning (AL) refers to the strategy where a pipeline learns to predict absolute object information *e.g.*, object or concept class. Relative Learning (RL) denotes similarity (relation) learning with the use of binary $\{0, 1\}$ and/or soft $[0; 1]$ similarity labels. Absolute-relative Learning (ArL) is a combination of AL and RL, which is akin to multi-task learning, and unary and pair-wise potentials in semantic segmentation. ArL is also conceptually closer to how humans learn from few examples.

However, there are two major problems in these relation learning pipelines, namely, (i) binary $\{0, 1\}$ labels are used to express the similarity between pairs of images, which cannot capture the similarity nuisances in the real-world setting due to the hardness of such modeling, which leads to biases in

the relation-based models, (ii) only pair-wise relation labels are used in these pipelines, so the models have no knowledge of the actual class concepts. In other words, these models are trained to learn the similarity between image pairs while they discard the explicit object classes despite they are accessible in the training stage.

We conjuncture that these two problems pose inconsistency between current few-shot learning approaches and human's cognitive processes. To this end, we propose the Absolute-relative Learning (ArL) which exposes few-shot learners to both similarity and class labels, and we employ semantic annotations to circumvent the issue with the somewhat rigid binary similarity labels $\{0, 1\}$.

Our ArL consists of two separate learning modules, namely, Absolute Learning (AL) and Relative Learning (RL). AL denotes the strategy in which we learn to predict the actual object categories or class concepts in addition to learning the class relations. In this way, the feature extracting network is exposed to additional object- or concept-related knowledge. RL refers to the similarity learning strategy for which (apart of binary $\{0, 1\}$ labels) we employ semantic annotations to promote the realistic similarity between image pairs. We use attributes or word2vec to obtain the semantic relation labels and learn element-wise similarities *e.g.*, if two objects have same colour, texture, *etc*. Such labels are further used as the supervisory cue in relation learning to capture the realistic soft relations between objects beyond the binary similarity.

By combing AL and RL which constitute on ArL, the relation network is simultaneously taught the class/object concepts together with more realistic class/object relations, thus naturally yielding an improved accuracy. Moreover, we use the predictions from the absolute and relative learners as interpretable features to promote the original relation learning via feedback connections.

Our approach is somewhat related to multi-modal learning which leverages multiple sources of data for training and testing. However, while multi-modal learning combines multiple streams of data on network inputs, our ArL models the semantic annotations in the label space, that is, we use them as the network output. We believe that using multiple abstractions of labels (relative *vs*. absolute) encourages the network to preserve more information about objects relevant to the few-shot learning task. Our strategy benefits from multi-task learning where two tasks learnt simultaneously help each other to outperform a naive fusion of two separate tasks. These tasks somewhat resemble unary and pair-wise potentials in semantic segmentation.

We note that obtaining the semantic information for novel classes (the testing step in few-shot learning) is not always easy or possible. Since our pipeline design is akin to multi-task rather than multi-modal learning, our model does not require additional labeling at the testing stage. Therefore, it is a more realistic setting than that of existing approaches.

In addition to the classic supervised few-shot recognition, we extend our ArL to the unsupervised scenario. Different with approach [12] that merely applies the self-supervised discriminator as an auxiliary task to improve the performance of supervised FSL, we develop an effective unsupervised FSL based on ArL. As there is no annotations for training samples, we rely on augmentation labelling (*e.g.*, rotations, flips and colors) to perform Absolute-relative Learning. Below, we summarize our contributions:

i. We propose so-called Absolute-relative Learning which can be embedded into popular few-shot pipelines to exploit both similarity and object/concept labelling.

ii. We extend our approach to unsupervised FSL, and we show how to create self-supervised annotations for unsupervised Absolute-relative Learning.

iii. We investigate the influence of different types of similarity measures on attributes in Relative Learning to simulate realistic object relations.

iv. We investigate the influence of different Absolute Learning branches on the classification performance.

To the best of our knowledge, we are the first to perform an in-depth analysis of object and class relation modeling in the context of supervised and unsupervised few-shot learning given the Absolute-relative Learning paradigm via class, semantic and augmentation annotations.

2. Related Work

Below, we describe recent one- and few-shot learning algorithms followed by semantic-based approaches.

2.1. Learning From Few Samples

For deep learning algorithms, the ability of *'learning from only a few examples is the desired characteristic to emulate in any brain-like system'* [33] is a desired operating principle which poses a challenge to typical CNNs designed for the large scale visual category recognition [34].

One- and Few-shot Learning has been studied widely in computer vision in both shallow [29, 28, 9, 3, 8, 23] and deep learning scenarios [19, 39, 37, 10, 37, 38, 46].

Early works [8, 23] propose one-shot learning methods motivated by the observation that humans can learn new concepts from very few examples. Siamese Network [19] presents a two-streams convolutional neural network approach which generates image descriptors and learns the similarity between them. Matching Network [39] introduces the concept of support set and L-way Z-shot learning protocols. It captures the similarity between one testing and several support images, thus casting the one-shot learning

problem as set-to-set learning. Prototypical Networks [37] learns a model that computes distances between a datapoint and prototype representations of each class. Model-Agnostic Meta-Learning (MAML) [10] introduces a meta-learning model trained on a variety of different learning tasks. Relation Net [38] is an efficient end-to-end network for learning the relationship between testing and support images. Conceptually, this model is similar to Matching Network [39]. However, Relation Net leverages an additional deep neural network to learn similarity on top of the image descriptor generating network. Second-order Similarity Network (SoSN) [46] is similar to Relation Net [38], which consists of the feature encoder and relation network. However, approach [38] uses first-order representations for similarity learning. In contrast, SoSN investigates second-order representations to capture co-occurrences of features. Graph Neural Networks (GNN) have also been applied to few-shot learning in many recent works [11, 18, 13] achieving promising results. Finally, noteworthy are domain adaptation and related approaches which can also operate in the small sample regime [20, 21, 45, 44, 25, 27, 26].

2.2. Learning from Semantic Labels

Semantic labels are used in various computer vision tasks *e.g.*, object classification, face and emotion recognition, image retrieval, transfer learning, and especially in zero-shot learning. Metric learning often uses semantic information *e.g.*, approach [5] proposes an image retrieval system which uses semantics of images via probabilistic modeling. Approach [41] presents a novel bi-relational graph model that comprises both the data graph and semantic label graph, and connects them by an additional bipartite graph built from label assignments. Approach [32] proposes a classifier based on semantic annotations and provides the theoretical bound linking the error rate of the classifier and the number of instances required for training. Approach [15] improves metric learning via the use of semantic labels with different types of semantic annotations.

Our relative learning is somewhat related to the idea using semantic information to learn metric. However, we use similarity measures to simulate realistic relation labels in supervised and unsupervised few-shot learning.

2.3. Multi-task Learning

Multi-task learning operates on a set of multiple related tasks. Approach [2] treats the multi-task learning as a convex iterative problem. Approach [16] considers the homoscedastic uncertainty of each task to weight multiple loss functions while HallNet [42] learns old-fashioned descriptors as auxiliary tasks for action recognition.

In contrast, we focus on how to refine the backbone by learning from class concepts and relations to address the high-level few-shot learning task.

3. Background

The concept of few-shot learning and the standard pipeline for few-shot learning are described next.

3.1. Relation Learning

Few-shot learning model typically consists of two parts: (i) feature encoder and (ii) relation module *e.g.*, a similarity network or a classifier. Below we take the two-stage 'feature encoder-relation network' [38, 46] as an example to elaborate on main aspects of few-shot learning pipelines.

A basic relation network [38, 46] contains 2-4 convolutional blocks and 2 fully-connected layers. Let us define the feature encoding network as $f\colon (\mathbb{R}^{W\times H};\mathbb{R}^{|\boldsymbol{\mathcal{F}}|})\rightarrow\mathbb{R}^{K\times N}$, where W and H denote the width and height of an input image, K is the length of feature vectors (number of filters), $N=N_W\cdot N_H$ is the total number of spatial locations in the last convolutional feature map. For simplicity, we denote an image descriptor by $\boldsymbol{\Phi}\in\mathbb{R}^{K\times N}$, where $\boldsymbol{\Phi}=f(\mathbf{X};\boldsymbol{\mathcal{F}})$ for an image $\mathbf{X}\in\mathbb{R}^{W\times H}$ and $\boldsymbol{\mathcal{F}}$ are the parameters-to-learn of the encoding network.

The relation network is denoted by $r\colon (\mathbb{R}^{K'};\mathbb{R}^{|\boldsymbol{\mathcal{R}}|})\rightarrow\mathbb{R}$. Typically, we write $r(\boldsymbol{\psi};\boldsymbol{\mathcal{R}})$, where $\boldsymbol{\psi}\in\mathbb{R}^{K'}$, whereas $\boldsymbol{\mathcal{R}}$ are the parameters-to-learn of the relation network.

3.2. Supervised Few-shot Learning

For the supervised L-way Z-shot problem, we assume some support images $\{\mathbf{X}_s\}_{s\in\mathcal{W}}$ from set $\mathcal{W}$ and their corresponding image descriptors $\{\boldsymbol{\Phi}_s\}_{s\in\mathcal{W}}$ which can be considered as a Z-shot descriptor. Moreover, we assume one query image $\mathbf{X}_q$ with its image descriptor $\boldsymbol{\Phi}_q$. Both the Z-shot and the query descriptors belong to one of L classes in the subset $\mathcal{C}^{\ddagger}\equiv\{c_1,...,c_L\}\subset\mathcal{I}_C\equiv\mathcal{C}$. The L-way Z-shot learning step can be defined as learning similarity:

$$\zeta_{sq}=r\left(\vartheta\left(\{\boldsymbol{\Phi}_s\}_{s\in\mathcal{W}},\boldsymbol{\Phi}_q^*\right),\boldsymbol{\mathcal{R}}\right), \tag{1}$$

where ζ refers to similarity prediction of given support-query pair, r refers to the relation network, and $\boldsymbol{\mathcal{R}}$ denotes network parameters that have to be learnt. ϑ is the relation operator on features of image pairs: we simply use concatenation.

Following approaches [38, 46], the Mean Square Error (MSE) is employed as the objective function:

$$L=\sum_{c\in\mathcal{C}^{\ddagger}}\sum_{c'\in\mathcal{C}^{\ddagger}}\left(r\left(\{\boldsymbol{\Phi}_s\}_{s\in\mathcal{W}_c},\boldsymbol{\Phi}_{q\in\mathcal{Q}:\ell(q)=c'},\boldsymbol{\mathcal{R}}\right)-\delta(c-c')\right)^2,$$
$$\text{where}\ \ \boldsymbol{\Phi}_s=f(\mathbf{X}_s;\boldsymbol{\mathcal{F}})\ \ \text{and}\ \ \boldsymbol{\Phi}_q=f(\mathbf{X}_q;\boldsymbol{\mathcal{F}}). \tag{2}$$

In the above equation, $\mathcal{W}_c$ is a randomly chosen set of support image descriptors of class $c\in\mathcal{C}^{\ddagger}$, $\mathcal{Q}$ is a randomly chosen set of L query image descriptors so that its consecutive elements belong to the consecutive classes in $\mathcal{C}^{\ddagger}\equiv\{c_1,...,c_L\}$. $\ell(q)$ corresponds to the label of $q\in\mathcal{Q}$. Lastly, δ refers to the indicator function equal 1 if its argument is 0.

3.3. Unsupervised Few-shot Learning

There are no class annotations that can be directly used for relation learning in the unsupervised setting. However, the popular self-supervised contrastive learning captures self-object relations by learning the similarity between different augmentations of the same image. Thus, we build our unsupervised few-shot learning pipeline based on contrastive learning. Given two image inputs $\mathbf{X}$ and $\mathbf{Y}$, we apply random augmentations on these images *e.g.*, rotation, flip, resized crop and color adjustment via operator $\text{Aug}(\cdot)$, which samples these transformations according to a uniform distribution. We obtain a set of M augmented images:

$$\hat{\mathbf{X}}_i \sim \text{Aug}(\mathbf{X}),\ \hat{\mathbf{Y}}_i \sim \text{Aug}(\mathbf{Y}),\ i \in \{1, ..., M\}. \quad (3)$$

We pass augmented images to the feature encoder f to get feature descriptors and obtain relation predictions $\zeta, \zeta^* \in \mathbb{R}^{M\times M}$ from relation network r for augmented samples of $\mathbf{X}$ and $\mathbf{Y}$, respectively, as well as relation predictions $\zeta' \in \mathbb{R}^{M\times M}$ evaluated between augmented samples of $\mathbf{X}$ and $\mathbf{Y}$:

$$\boldsymbol{\Phi}_i = f(\hat{\mathbf{X}}_i; \boldsymbol{\mathcal{F}}),\ \boldsymbol{\Phi}^*_j = f(\hat{\mathbf{Y}}_j; \boldsymbol{\mathcal{F}}),\ i, j \in \{1, ..., M\}, \quad (4)$$
$$\zeta_{ij} = r\left(\boldsymbol{\Phi}_i, \boldsymbol{\Phi}_j; \boldsymbol{\mathcal{R}}\right),\ \zeta'_{ij} = r\left(\boldsymbol{\Phi}_i, \boldsymbol{\Phi}^*_j; \boldsymbol{\mathcal{R}}\right),\ \zeta^*_{ij} = r\left(\boldsymbol{\Phi}^*_i, \boldsymbol{\Phi}^*_j; \boldsymbol{\mathcal{R}}\right).$$

Lastly, we minimize the contrastive loss L_{urn} w.r.t. $\boldsymbol{\mathcal{F}}$ and $\boldsymbol{\mathcal{R}}$ in order to push closer augmented samples generated from the same image ($\mathbf{X}$ and $\mathbf{Y}$, resp.) and push away augmented samples generated from pairs images $\mathbf{X}$ and $\mathbf{Y}$:

$$L_{urn} = \|\,\zeta - 1\,\|^2_F + \|\,\zeta^* - 1\,\|^2_F + \|\,\zeta'\,\|^2_F\,. \quad (5)$$

In practice, we sample a large number of image pairs $\mathbf{X}$ and $\mathbf{Y}$ with the goal of minimizing Eq. (5).

4. Approach

Below, we firstly explain the Relative Learning and Absolute Learning modules followed by the introduction of the Absolute-relative Learning pipeline. We note that all auxiliary information *e.g.*, attributes and word2vec embeddings are used in the label space (not as extra inputs).

Given images $\mathbf{X}_i$ and $\mathbf{X}_j$, we feed them into the feature encoder f to get image representations $\boldsymbol{\Phi}_i = f(\mathbf{X}_i; \boldsymbol{\mathcal{F}})$ and $\boldsymbol{\Phi}_j = f(\mathbf{X}_j; \boldsymbol{\mathcal{F}})$, where $\boldsymbol{\mathcal{F}}$ are the parameters of feature encoder. Subsequently, we perform our proposed Relative Learning and Absolute Learning on $\boldsymbol{\Phi}_i$ and $\boldsymbol{\Phi}_j$.

4.1. Relative Learning

In conventional few-shot learning, binary class labels are employed to train the CNNs in order to model the relations between pairs of images. However, labeling such pairs as similar/dissimilar (*i.e.*, $\{0, 1\}$) cannot fully reflect the actual relations between objects.

In this paper, we take a deeper look at how to represent relations in the few-shot learning scenario. To better exploit class relations in the label space, we employ semantic annotations *e.g.*, attributes and word2vec. Based on these semantic annotations, we investigate how semantic relation labels influence the final few-shot learning performance.

Figure 2 (bottom right corner) shows that the classic relation learning can be viewed as an intersection (or relation) operation over the original class labels. Thus, we apply intersection on the semantic annotations to obtain the relative semantic information for the relative supervision, which can contribute to obtaining more realistic image relations in the label space. Let us denote the class labels and attributes of image $\mathbf{X}_i$ as $c_i, \mathbf{a}_i$. Given two samples $\mathbf{X}_i$ and $\mathbf{X}_j$ with their class labels c_i, c_j (and one-hot vectors $\mathbf{c}_i, \mathbf{c}_j$) and attributes $\mathbf{a}_i, \mathbf{a}_j$, we obtain the binary relation label $\hat{c}_{ij}$ which represents if the two images are from the same class. We also have semantic relation label $\hat{a}_{ij}$ which represents attributes shared between $\mathbf{X}_i$ and $\mathbf{X}_j$. Semantic annotations often contain continuous rather than binary values. Thus, we use the RBF function with the ℓ^p_p norm. Specifically, we obtain:

$$\hat{c}_{ij} = c_i \wedge c_j = \delta(\mathbf{c}_i - \mathbf{c}_j) \text{ and } \hat{a}_{ij} = e^{-||\mathbf{a}_i - \mathbf{a}_j||^p_p}, \quad (6)$$

If we train the network only with $\hat{c}_{ij}$, it becomes the basic few-shot learning. However, the simultaneous use of $\hat{c}_{ij}$ and $\hat{a}_{ij}$ for similarity learning should yield smoother similarity decision boundaries.

To learn from multi-modal relative supervisions, we apply a two-stage learner consisting of a shared part g and respective parts r. Let us denote the class and semantic relative learners as r_c and r_s. To make relative predictions, we firstly apply the relation operator ϑ over $\boldsymbol{\Phi}_i$ and $\boldsymbol{\Phi}_j$ (concatenation along the channel mode), and feed such a relation descriptor into g (4 blocks of Conv-BN-ReLU-MaxPool) to obtain the refined pair-wise representation $\boldsymbol{\psi}_{ij}$:

$$\boldsymbol{\psi}_{ij} = g(\vartheta(\boldsymbol{\Phi}_i, \boldsymbol{\Phi}_j); \boldsymbol{\mathcal{G}}). \quad (7)$$

Subsequently, we feed $\boldsymbol{\psi}_{ij}$ into learners r_c and r_a to get class- and semantics-wise relation predictions $\hat{c}^*_{ij}$ and $\hat{a}^*_{ij}$:

$$\hat{c}^*_{ij} = r_c(\boldsymbol{\psi}_{ij}; \boldsymbol{\mathcal{R}}_c) \text{ and } \hat{a}^*_{ij} = r_s(\boldsymbol{\psi}_{ij}; \boldsymbol{\mathcal{R}}_s), \quad (8)$$

where $\boldsymbol{\mathcal{R}}_c$ and $\boldsymbol{\mathcal{R}}_s$ refer to the parameters of r_c and r_s, respectively. The objectives for class- and semantic-wise relative learners are:

$$L_{relc} = \sum_i \sum_j \left(r_c\left(\boldsymbol{\psi}_{ij}; \boldsymbol{\mathcal{R}}_c\right) - \hat{c}_{ij}\right)^2, \quad (9)$$

$$L_{rels} = \sum_i \sum_j \left(r_s\left(\boldsymbol{\psi}_{ij}; \boldsymbol{\mathcal{R}}_s\right) - \hat{a}_{ij}\right)^2. \quad (10)$$

4.2. Absolute Learning

In contrast to Relative Learning which applies the relative labels to learn similarity, Absolute Learning refers to

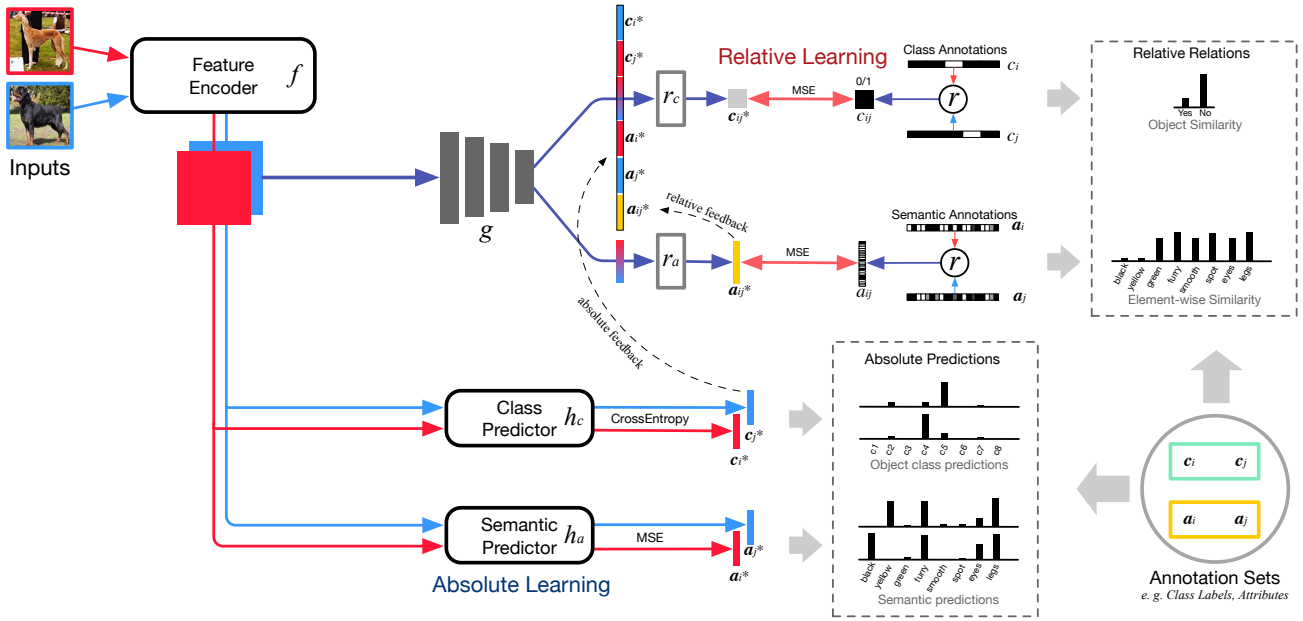

Figure 2: The proposed pipeline for our Absolute-relative Learning (supervised setting). It consists of three blocks, namely (i) feature encoder to extract the image representations, (ii) Absolute Learning module to enhance the feature quality with auxiliary supervision, (iii) Relative Learning module to learn image relations based on multi-modal relation supervisions. With our Absolute-relative Learning, we want to both learn if the two objects share the same label and how similar they are semantically *e.g.*, in terms of shared visual attributes.

the strategy in which the network learns predefined object annotations *e.g.*, class labels, attributes, *etc*. The motivation behind the Absolute Learning is that current few-shot learning pipelines use the relation labels as supervision which prevents the network from capturing objects concepts. In other words, the network knows if the two objects are similar (or not) but it does not know what these objects are.

Branches for Absolute Learning are shown in Figure 2. In this paper, we apply an additional network branch following the feature encoder to learn the absolute object annotations. Firstly, consider the class prediction as an example. Once we obtain the image representation $\boldsymbol{\Phi}_i$ given image $\mathbf{X}_i$, we feed it into the class absolute learner h_c with parameter $\boldsymbol{\mathcal{H}}_c$.:

$$\mathbf{c}_i^* = h_c(\boldsymbol{\Phi}_i; \boldsymbol{\mathcal{H}}_c). \tag{11}$$

Subsequently, we apply the cross-entropy loss to train the class absolute learner (l^c is the target class integer):

$$L_{absc} = -\sum_i^N log(\frac{exp(c_i^*[l_i^c])}{\sum_j exp(c_i^*[j])}). \tag{12}$$

For the semantic absolute learner, we use the MSE loss by feeding $\boldsymbol{\Phi}_i$ into h_s:

$$\mathbf{a}_i^* = h_s(\boldsymbol{\Phi}_i; \boldsymbol{\mathcal{H}}_s), \tag{13}$$

$$L_{abss} = \frac{1}{N}\sum_i^N ||\mathbf{a}_i - \mathbf{a}_i^*||_2^2. \tag{14}$$

The Absolute Learning module may appear somewhat similar to self-supervised learning applied to few-shot learning. However, we use discriminators to classify different types of object annotations while the typical self-supervision recognises the patterns of image transformations. We believe our strategy helps refine the feature encoder to capture both the notion of similarity as well as concrete object concepts.

4.3. Absolute-relative Learning

For our Absolute-relative Learning (ArL), we simultaneously train the relation network with relative object similarity labels, and introduce an auxiliary task which learns specific object labels. The pipeline of ArL is shown in Figure 2 which highlights that the ArL model uses the auxiliary semantic soft labels to train the relation network to capture more realistic image relations while employing auxiliary predictor branches to infer different types of object information, thus refining the feature representations and the feature encoder.

In addition to merging the absolute/relative learners, we introduce several connections from the outputs of absolute and relative learners wired to relative learners to promote the original relation learner, which does not require absolute labels or semantic labels at the testing time. In contrast, multi-modal learning needs all modalities in the testing step.

Figure 3 shows the Absolute-relative Learning pipeline (unsupervised setting). As the supervised ArL, the unsupervised ArL pipeline consists of absolute and relative learners. However, the annotations used during the training phase are self-supervised augmentation keys, not class labels.

Let l denote the number of layers in g. We apply ϑ over the intermediate descriptor $\boldsymbol{\psi}_{ij}^{(l-1)}$, which is the ($l$-1)-th layer

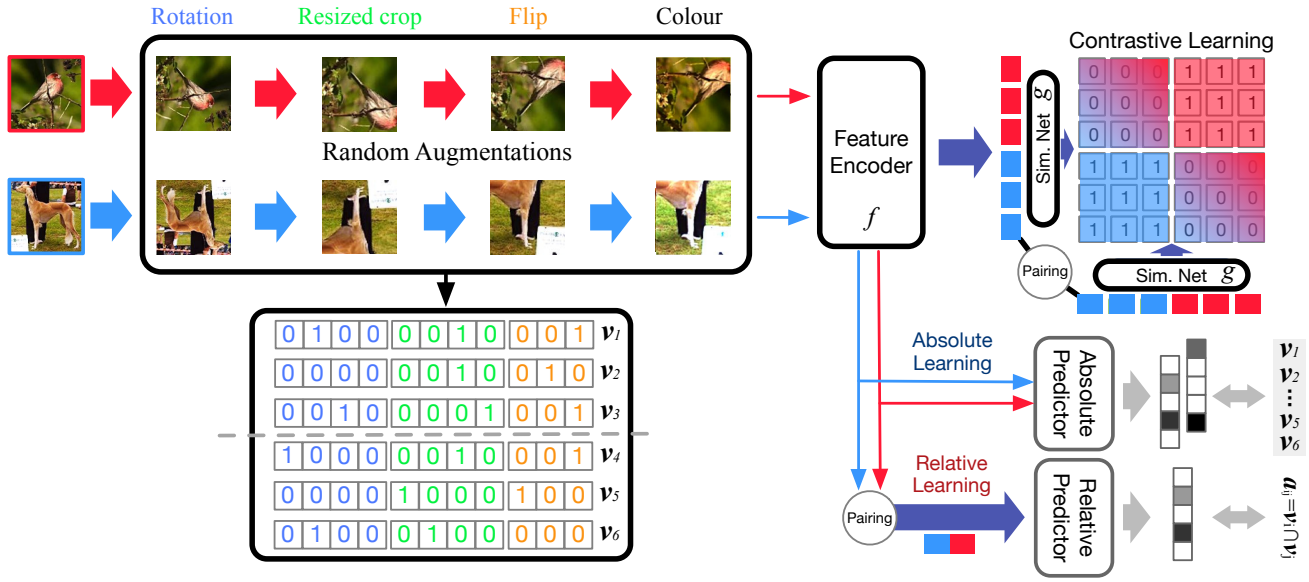

Figure 3: The proposed pipeline for Absolute-relative Learning (unsupervised setting). In contrast to supervised ArL that uses class and semantic annotations in absolute and relative learners, we apply a random augmentation sequence to augment unlabeled datapoints, and we store the augmentation keys as instance annotations.

of g, and absolute predictions $\mathbf{c}_i^*, \mathbf{c}_j^*, \mathbf{a}_i^*, \mathbf{a}_j^*$. We call this operation the absolute feedback:

$$\hat{\boldsymbol{\psi}}_{ij}^{(l-1)} = \vartheta\left(\boldsymbol{\psi}_{ij}^{(l-1)}, \mathbf{c}_i^*), \mathbf{c}_j^*), \mathbf{a}_i^*, \mathbf{a}_j^*\right). \tag{15}$$

We use $\hat{\boldsymbol{\psi}}_{ij}$ from the last layer of g to train the semantic relative learner r_s:

$$\hat{\boldsymbol{\psi}}_{ij} = \hat{\boldsymbol{\psi}}_{ij}^{(l)} = g^{(l)}(\hat{\boldsymbol{\psi}}_{ij}^{(l-1)}), \tag{16}$$

$$L_{rels} = \sum_i \sum_j \left(r_s(\hat{\boldsymbol{\psi}}_{ij}; \boldsymbol{\mathcal{R}}_s) - \hat{a}_{ij}\right)^2, \ \hat{a}_{ij} = e^{-||\mathbf{a}_i - \mathbf{a}_j|_p^p}.$$

Let $\hat{\mathbf{a}}_{ij}^*$ denote the outputs of semantic relative learner. Then we apply the relative feedback by combining $\hat{\boldsymbol{\psi}}_{ij}$ and $\hat{\mathbf{a}}_{ij}^*$ to promote the training of class relative learner r_c:

$$\hat{\boldsymbol{\psi}}_{ij}^* = \vartheta(\hat{\boldsymbol{\psi}}_{ij}, \hat{\mathbf{a}}_{ij}^*), \tag{17}$$

$$L_{relc} = \sum_i \sum_j \left(r_c(\hat{\boldsymbol{\psi}}_{ij}^*; \boldsymbol{\mathcal{R}}_c) - \hat{c}_{ij}\right)^2, \ \hat{c}_{ij} = \delta(\mathbf{c}_i - \mathbf{c}_j).$$

We minimize the following objective for ArL:

$$\min \quad L_{relc} + \alpha L_{rels} + \beta L_{absc} + \gamma L_{abss}. \tag{18}$$

where $(\alpha, \beta, \gamma) \in [0.001; 1]^3$ are hyper-parameters that control the impact of each learner and are estimated with 20 steps of the HyperOpt package [4] on a given validation set. Nullifying α, β or γ disables corresponding losses.

5. Experiments

Below, we demonstrate the usefulness of our approach by evaluating it on the *mini*Imagenet [39], fine-grained CUB-200-2011 [40] and Flower102 [31] datasets. Figure 2 presents our ArL with the two-stage relation learning pipeline but ArL applies to any few-shot learning models with any type of base learners (*e.g.*, nearest neighbour discrimination, relation module, multi-class linear classifier, *etc*). The core objective of ArL is to improve the representation quality. Thus, we employ the classic baseline models, *i.e.*, Prototypical Net (PN) [37], Relation Net [38], SoSN [46], MetaOptNet [24], *etc*, as our baseline models to evaluate our Relative Learning, Absolute Learning and the Absolute-relative Learning in both supervised and unsupervised settings. The Adam solver is used for model training. We set the initial learning rate to be 0.001 and decay it by 0.5 every 50000 iterations. We evaluate ArL on RelationNet (RN) [38], Prototypical Net (PN) [37], Second-order Similarity Network (SoSN) [46] and MetaOptNet [24]. For augmentations in the unsupervised setting, we randomly apply resized crop (scale 0.6–1.0, ratio 0.75–1.33), horizontal+vertical flips, rotations (0–360°), and color jitter.

5.1. Datasets

Below, we describe our setup, standard and fine-grained datasets with semantic annotations and evaluation protocols.

***mini*Imagenet** [39] consists of 60000 RGB images from 100 classes, each class containing 600 samples. We follow the standard protocol [39] and use 80/20 classes for training/testing, and images of size 84×84 for fair comparisons with other methods. For semantic annotations, we manually annotate 31 attributes for each class. We also leverage word2vec extracted from GloVe as the class embedding.

Caltech-UCSD-Birds 200-2011 (CUB-200-2011) [40] has

Table 1: Evaluations on the *mini*Imagenet dataset (5-way acc. given) for the ArL in supervised and unsupervised settings. ('U-' refers to the unsupervised FSL.)

Model	Backbone	1-shot	5-shot
		Supervised Few-shot Learning	
Matching Nets [39]	-	43.56 ± 0.84	55.31 ± 0.73
Meta Nets [30]	-	49.21 ± 0.96	-
PN [37]	Conv-4-64	49.42 ± 0.78	68.20 ± 0.66
MAML [10]	Conv-4-64	48.70 ± 1.84	63.11 ± 0.92
RN [38]	Conv-4-64	51.36 ± 0.82	66.12 ± 0.70
SoSN [46]	Conv-4-64	53.73 ± 0.83	68.58 ± 0.70
SoSN [46]	ResNet-12	59.01 ± 0.83	75.49 ± 0.68
MAML++ [1]	Conv-4-64	52.15 ± 0.26	68.32 ± 0.44
MetaOptNet [24]	ResNet-12	62.64 ± 0.61	78.63 ± 0.46
PN + ArL	Conv-4-64	53.93 ± 0.65	69.68 ± 0.45
RN + ArL	Conv-4-64	53.79 ± 0.68	68.86 ± 0.43
SoSN + ArL	Conv-4-64	57.48 ± 0.65	72.64 ± 0.45
SoSN + ArL	ResNet-12	61.36 ± 0.67	78.95 ± 0.42
MetaOptNet + ArL	ResNet-12	65.21 ± 0.58	80.41 ± 0.49
		Unsupervised Few-shot Learning	
Pixel (Cosine)	-	23.00	26.60
BiGAN (k_{nn}) [7]	-	25.56	31.10
BiGAN (cluster matching) [7]	-	24.63	29.49
DeepCluster (k_{nn}) [6]	-	28.90	42.25
DeepCluster (cluster matching) [6]	-	22.20	23.50
UMTRA [17]	Conv-4-64	39.91	50.70
CACTUs [14]	Conv-4-64	39.94	54.01
U-RN	Conv-4-64	35.14 ± 0.91	44.10 ± 0.88
U-PN	Conv-4-64	35.85 ± 0.85	48.01 ± 0.82
U-SoSN	Conv-4-64	37.94 ± 0.87	50.95 ± 0.81
U-RN + ArL	Conv-4-64	36.37 ± 0.92	46.97 ± 0.86
U-PN + ArL	Conv-4-64	38.76 ± 0.84	51.08 ± 0.84
U-SoSN + ArL	Conv-4-64	41.13 ± 0.84	55.39 ± 0.79
U-SoSN + ArL	ResNet-12	41.08 ± 0.83	$\mathbf{57.01 \pm 0.79}$

11788 images of 200 bird species. 100/50/50 classes are randomly selected for meta-training, meta-validation and meta-testing. 312 attributes are provided for each class.

Table 2: Evaluations on the CUB-200-2011 and Flower102. (5-way acc. given).

Model	CUB-200-2011		Flower102	
	1-shot	5-shot	1-shot	5-shot
	Supervised Few-shot Learning			
PN [37]	37.42	51.57	62.81	82.11
RN [38]	40.56	53.91	68.26	80.94
SoSN [46]	46.72	60.34	71.90	84.87
RN + ArL	44.53	58.76	71.12	83.49
SoSN - RL(cls.) [46]	46.72	60.34	71.90	84.87
SoSN - RL(att.)	49.24	64.04	74.96	87.21
SoSN - AL(cls.)	46.88	60.90	72.97	85.35
SoSN - AL(att.)	48.85	63.64	74.31	86.97
SoSN + ArL	**50.62**	65.87	76.21	88.36
	Unsupervised Few-shot Learning			
BiGAN(k_{nn})[7]	28.02	30.17	44.68	59.12
U-RN	29.36	36.36	55.54	68.86
U-PN	29.87	37.13	55.36	68.49
U-SoSN	36.89	45.81	61.26	75.98
U-RN + ArL	31.27	38.41	57.19	70.23
U-PN + ArL	31.58	39.95	57.61	70.31
U-SoSN + ArL	37.93	51.55	69.14	84.10

Flower102 [31] is a fine-grained category recognition dataset that contains 102 classes of various flowers. Each class consists of 40-258 images. We randomly select 60 meta-train classes, 20 meta-validation classes and 22 meta-test classes. 1024 attributes are provided for each class.

5.2. Performance Analysis

Absolute-relative Learning (ArL). Table 1 shows that Absolute-relative Learning (ArL) effectively improves the performance on all datasets. On *mini*Imagenet, SoSN+ArL improve the 1- and 5-shot performance by 3.6% and 4.1%, MetaOptNet+ArL improves the performance by 2.6% and

Table 3: Ablation study of the impact of different annotations (*e.g.*, class labels, attributes) on ArL.

Baseline	Rel. Learn. cls.	Rel. Learn. att.	Abs. Learn. cls.	Abs. Learn. att.	Top-1 Acc. 1-shot	Top-1 Acc. 5-shot
RN	✓				51.36	65.32
		✓			52.38	66.74
			✓		51.41	66.01
				✓	52.35	66.53
SoSN	✓				53.73	68.58
		✓			55.56	70.97
			✓		55.12	70.91
				✓	55.31	71.03

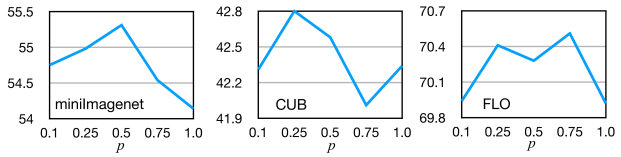

Figure 4: The validation of p given the semantic similarity measure function $e^{-||\mathbf{a}_i - \mathbf{a}'_i||_p^p}$ on selected datasets.

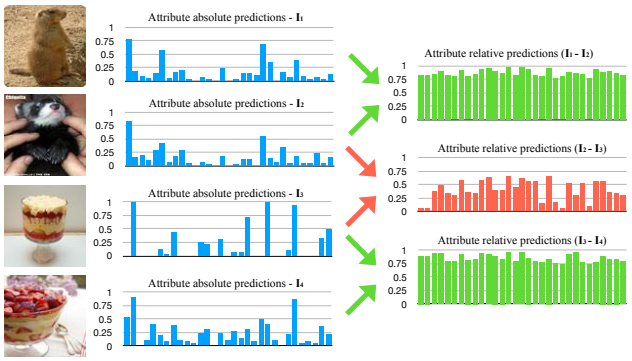

Figure 5: Visualization of semantic absolute and relative predictions which shows how their bins relate.

1.8% respectively. Not in the table, DeepEMD [43] (ResNet-12) and LaplacianFSL [49] (ResNet-18) scored 65.91% and 66.41% (1-shot prot.) In contrast, DeepEMD+ArL and LaplacianFSL+ArL scored 67.24% and 68.07%.

For fine-grained datasets, CUB-200-2011 and Flower102 in Table 2, SoSN+ArL improves the 1- and 5-shot accuracy by 1.4% and 1.6%. For unsupervised learning, ArL with SoSN brings 3.5% and 4.4% gain on *mini*Imagenet, 1.0% and 5.7% gain on CUB-200-2011, 7.9% and 8.1% gain on Flower102 for 1- and 5-shot learning, respectively. Our unsupervised U-SoSN+ArL often outperforms recent supervised methods on fine-grained classification datasets.

Visualization. Below, we visualize absolute and relative semantic predictions to explain how such an information can be used. As shown in Fig. 5, we randomly select 4 images from two classes, among which I_1 and I_2 belong to one class, and I_3 and I_4 belong to another class. Figure 5 shows that the semantic absolute predictions of images from the same class have more consistent distributions, the relative predictions over same-class image pairs have high responses to the same subset of bins. Predictions over images from disjoint classes result in smaller intersection of corresponding peaks.

Ablations on absolute and relative learners. Figure 4 shows results w.r.t. p from Eq. 6. Table 3 shows how different absolute and relative learners affect few-shot learning results on *mini*Imagenet. For example, for the SoSN baseline, the attribute-based absolute and relative learners work the best among all absolute and relative learning modules.

Relative Learning (RL). Table 3 (*mini*Imagenet) illustrates the performance enhanced by the semantic-based relation on Relation Net, SoSN and SalNet. Results in the table

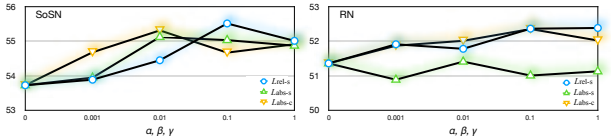

Figure 6: Ablations on α, β, γ for SoSN [46] and RN [38] in the supervised setting. These evaluations are just an illustration as we tune parameters on the validation splits via the HyperOpt package.

indicate that the performance of few-shot similarity learning can be improved by employing the semantic relation labels at the training stage. For instance, SoSN with attribute soft label (*att.*) achieves 0.6% and 1.7% gain for 1- and 5-shot protocols, compared with the baseline (*SoSN*) in Table 1. The results on CUB-200-2011 and Flower102 from Table 2 indicate similar gains.

Absolute Learning (AL). Table 3 shows that different absolute learning modules help improve the performance on *mini*Imagenet. SoSN with the attribute predictor (*SoSN-AL*) achieves the best performance of 55.61% on 1-shot and 71.03% (5-shot). Table 3 shows that applying multiple absolute learning modules does not always further improve the accuracy. The attribute-based predictor (*att.*) also works the best among all variants on CUB-200-2011 and Flower102. For instance, SoSN with the attribute-based predictor achieves 2.1% and 3.3% improvements on CUB-200-2011, and 2.4% and 2.1% improvement on Flower102 for 1- and 5-shot protocols, respectively. We note that the class predictor (*cls.*) does not work well on the fine-grained classification datasets.

6. Conclusions

In this paper, we have demonstrated that binary labels commonly used in few-shot learning cannot capture complex class relations well, leading to inferior results. Thus, we have introduced semantic annotations to aid the modeling of more realistic class relations during network training. Moreover, we have proposed a novel Absolute-relative Learning (ArL) paradigm which combines the similarity learning with the concept learning, and we extend ArL to unsupervised FSL. This surprisingly simple strategy appears to work well on all datasets in both supervised and unsupervised settings, and it perhaps resembles a bit more closely the human learning processes. In contrast to multi-modal learning, we only use semantic annotations as labels in training, and do not use them during testing. Our proposed approach achieves the state-of-the-art performance on all few-shot learning protocols.

Acknowledgements. This work is in part supported by the Equipment Research and Development Fund (no. ZXD2020C2316), NSF Youth Science Fund (no. 62002371), the ANU VC's Travel Grant and CECS Dean's Travel Grant (H. Zhang's stay at the University of Oxford).

References

[1] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your maml. *arXiv preprint arXiv:1810.09502*, 2018. 7

[2] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. In *Advances in neural information processing systems*, pages 41–48, 2007. 3

[3] Evgeniy Bart and Shimon Ullman. Cross-generalization: Learning novel classes from a single example by feature replacement. *CVPR*, pages 672–679, 2005. 2

[4] James Bergstra, Brent Komer, Chris Eliasmith, Dan Yamins, and David D Cox. Hyperopt: a python library for model selection and hyperparameter optimization. *Computational Science & Discovery*, 8(1):014008, 2015. 6

[5] Ben Bradshaw. Semantic based image retrieval: a probabilistic approach. In *Proceedings of the eighth ACM international conference on Multimedia*, pages 167–176. ACM, 2000. 3

[6] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018. 7

[7] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016. 7

[8] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *PAMI*, 28(4):594–611, 2006. 2

[9] Michael Fink. Object classification from a single example utilizing class relevance metrics. *NIPS*, pages 449–456, 2005. 2

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135, 2017. 2, 3, 7

[11] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017. 3

[12] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. *arXiv preprint arXiv:1906.05186*, 2019. 2

[13] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[14] Kyle Hsu, S. Levine, and Chelsea Finn. Unsupervised learning via meta-learning. *ArXiv*, abs/1810.02334, 2019. 7

[15] Mengdi Huai, Chenglin Miao, Yaliang Li, Qiuling Suo, Lu Su, and Aidong Zhang. Metric learning from probabilistic labels. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1541–1550. ACM, 2018. 3

[16] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[17] Siavash Khodadadeh, Ladislau Bölöni, and M. Shah. Unsupervised meta-learning for few-shot image and video classification. *ArXiv*, abs/1811.11819, 2018. 7

[18] Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D. Yoo. Edge-labeling graph neural network for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[19] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2, 2015. 2

[20] Piotr Koniusz, Yusuf Tas, and Fatih Porikli. Domain adaptation by mixture of alignments of second-or higher-order scatter tensors. In *CVPR*, volume 2, 2017. 3

[21] Piotr Koniusz, Yusuf Tas, Hongguang Zhang, Mehrtash Harandi, Fatih Porikli, and Rui Zhang. Museum exhibit identification challenge for the supervised domain adaptation and beyond. *ECCV*, pages 788–804, 2018. 3

[22] Piotr Koniusz and Hongguang Zhang. Power normalizations in fine-grained image, few-shot image and graph classification. *TPAMI*, 2020. 1

[23] Brenden M. Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua B. Tenenbaum. One shot learning of simple visual concepts. *CogSci*, 2011. 2

[24] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10657–10665, 2019. 6, 7

[25] Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1459–1469, 2020. 3

[26] Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Benjamin Swift, Hanna Suominen, and Hongdong Li. Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. In *Advances in Neural Information Processing Systems*, volume 33, pages 12034–12045, 2020. 3

[27] Dongxu Li, Xin Yu, Chenchen Xu, Lars Petersson, and Hongdong Li. Transferring cross-domain knowledge for video sign language recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6205–6214, 2020. 3

[28] Fei Fei Li, Rufin VanRullen, Christof Koch, and Pietro Perona. Rapid natural scene categorization in the near absence of attention. *Proceedings of the National Academy of Sciences*, 99(14):9596–9601, 2002. 2

[29] E. G. Miller, N. E. Matsakis, and P. A. Viola. Learning from one example through shared densities on transforms. *CVPR*, 1:464–471, 2000. 2

[30] Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2554–2563. JMLR. org, 2017. 7

[31] M-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008. 6, 7

[32] Peng Peng, Raymond Chi-Wing Wong, and Philip S Yu. Learning on probabilistic labels. In *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 307–315. SIAM, 2014. 3

[33] Jagath Chandana Rajapakse and Lipo Wang. *Neural Information Processing: Research and Development*. Springer-Verlag Berlin and Heidelberg GmbH & Co. KG, 2004. 2

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 2

[35] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Tim Lillicrap. A simple neural network module for relational reasoning. *NIPS*, 2017. 1

[36] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1

[37] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, pages 4077–4087, 2017. 1, 2, 3, 6, 7

[38] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. *CoRR:1711.06025*, 2017. 1, 2, 3, 6, 7, 8

[39] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *NIPS*, pages 3630–3638, 2016. 1, 2, 3, 6, 7

[40] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 6

[41] Hua Wang, Heng Huang, and Chris Ding. Image annotation using bi-relational graph of images and semantic labels. In *CVPR 2011*, pages 793–800. IEEE, 2011. 3

[42] Lei Wang, Piotr Koniusz, and Du Huynh. Hallucinating IDT descriptors and I3D optical flow features for action recognition with cnns. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV*, pages 8697–8707, 2019. 3

[43] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 8

[44] Hongguang Zhang and Piotr Koniusz. Model selection for generalized zero-shot learning. In *European Conference on Computer Vision*, pages 198–204. Springer, 2018. 3

[45] Hongguang Zhang and Piotr Koniusz. Zero-shot kernel learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[46] Hongguang Zhang and Piotr Koniusz. Power normalizing second-order similarity network for few-shot learning. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1185–1193. IEEE, 2019. 1, 2, 3, 6, 7, 8

[47] Hongguang Zhang, Jing Zhang, and Piotr Koniusz. Few-shot learning via saliency-guided hallucination of samples. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1

[48] H Zhang, L Zhang, X Qi, H Li, PHS Torr, and P Koniusz. Few-shot action recognition with permutation-invariant attention. In *Proceedings of the European Conference on Computer Vision (ECCV 2020)*, volume 12350. Springer, 2020. 1

[49] Imtiaz Masud Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning, 2020. 1, 8