



# Improving Few-shot Learning by Spatially-aware Matching and CrossTransformer

Hongguang Zhang<sup>1</sup> , Philip H. S. Torr<sup>4</sup>, and Piotr Koniusz<sup>2,3</sup> 

<sup>1</sup> Systems Engineering Institute, AMS

[zhang.hongguang@outlook.com](mailto:zhang.hongguang@outlook.com)

<sup>2</sup> Data61/CSIRO

[piotr.koniusz@data61.csiro.au](mailto:piotr.koniusz@data61.csiro.au)

<sup>3</sup> Australian National University

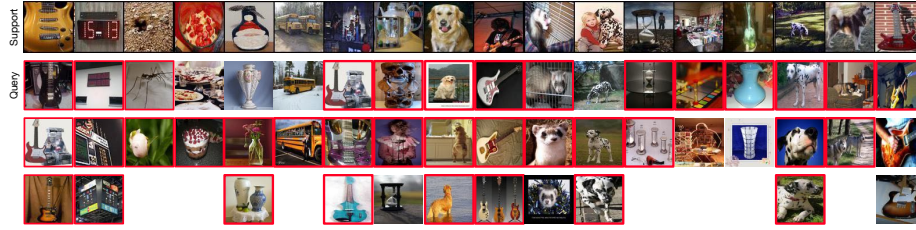
<sup>4</sup> Oxford University

**Abstract.** Current few-shot learning models capture visual object relations in the so-called meta-learning setting under a fixed-resolution input. However, such models have a limited generalization ability under the scale and location mismatch between objects, as only few samples from target classes are provided. Therefore, the lack of a mechanism to match the scale and location between pairs of compared images leads to the performance degradation. The importance of image contents varies across coarse-to-fine scales depending on the object and its class label, *e.g.*, generic objects and scenes rely on their global appearance while fine-grained objects rely more on their localized visual patterns. In this paper, we study the impact of scale and location mismatch in the few-shot learning scenario, and propose a novel Spatially-aware Matching (SM) scheme to effectively perform matching across multiple scales and locations, and learn image relations by giving the highest weights to the best matching pairs. The SM is trained to activate the most related locations and scales between support and query data. We apply and evaluate SM on various few-shot learning models and backbones for comprehensive evaluations. Furthermore, we leverage an auxiliary self-supervisory discriminator to train/predict the spatial- and scale-level index of feature vectors we use. Finally, we develop a novel transformer-based pipeline to exploit self- and cross-attention in a spatially-aware matching process. Our proposed design is orthogonal to the choice of backbone and/or comparator.

**Keywords:** few-shot · multi-scale · transformer · self-supervision.

## 1 Introduction

CNNs are the backbone of object categorization, scene classification and fine-grained image recognition models but they require large amounts of labeled data. In contrast, humans enjoy the ability to learn and recognize novel objects and complex visual concepts from very few samples, which highlights the superiority of biological vision over CNNs. Inspired by the brain ability to learn in the few-samples regime, researchers study the so-called problem of few-shot learning



**Fig. 1.** Scale mismatch in *mini*-ImageNet. Top row: support samples randomly selected from episodes. Remaining rows: failure queries are marked by red boxes. We estimated that  $\sim 30\%$  mismatches are due to the object scale mismatch, which motivates the importance of scale and region matching in few-shot learning.

for which networks are adapted by the use of only few training samples. Several proposed relation-learning deep networks [1,2,3,4] can be viewed as performing a variant of metric learning, which they fail to address the scale- and location-mismatch between support and query samples as shown in Figure 1. We follow such models and focus on studying how to capture the most discriminative object scales and locations to perform accurate matching between the so-called query and support representations.

A typical relational few-shot learning pipeline consists of (i) feature encoder (backbone), (ii) pooling operator which aggregates feature vectors of query and support images of an episode followed by forming a relation descriptor, and (iii) comparator (base learner). In this paper, we investigate how to efficiently apply spatially-aware matching between query and support images across different locations and scales. To this end, we propose a Spatially-aware Matching (SM) scheme which scores the compared locations and scales. The scores can be regularized to induce sparsity and used to re-weight similarity learning loss operating on  $\{0, 1\}$  labels (different/same class label). Note that our SM is orthogonal to the choice of baseline, therefore it is applicable to many existing few-shot learning pipelines.

As the spatial size (height and width) of convolutional representations vary, pooling is required before feeding the representations into the comparator. We compare several pooling strategies, *i.e.*, average, max and second-order pooling (used in object, texture and action recognition, fine-grained recognition, and few-shot learning [5,6,7,8,9,4,10]) which captures covariance of features per region/scale. Second-order Similarity Network (SoSN) [4] is the first work which validates the usefulness of autocorrelation representations in few-shot learning. In this paper, we employ second-order pooling as it is permutation-invariant w.r.t. the spatial location of aggregated vectors while capturing second-order statistics which are more informative than typical average-pooled first-order features. As second-order pooling can aggregate any number of feature vectors into a fixed-size representation, it is useful in describing regions of varying size for spatially-aware matching.

Though multi-scale modeling has been used in low-level vision problems, and matching features between regions is one of the oldest recognition tools [11,12], relation-based few-shot learning (similarity learning between pairs of images) has not used such a mechanism despite clear benefits.

In addition to our matching mechanism, we embed the self-supervisory discriminators into our pipeline whose auxiliary task has to predict scale and spatial annotations, thus promoting a more discriminative training of encoder, attention and comparator. This is achieved by the use of Spatial-aware Discriminator (SD) to learn/predict the location and scale indexes of given features. Such strategies have not been investigated in matching, but they are similar to pretext tasks in self-supervised learning.

Beyond using SM on classic few-shot learning pipelines, we also propose a novel transformer-based pipeline, Spatially-aware Matching CrossTransformer (SmCT), which learns the object correlations over locations and scales via cross-attention. Such a pipeline is effective when being pre-trained on large-scale datasets.

Below we summarize our contributions:

- i. We propose a novel spatially-aware matching few-shot learning strategy, compatible with many existing few-shot learning pipelines. We form possible region- and scale-wise pairs, and we pass them through comparator whose scores are re-weighted according to the matching score of region- and scale-wise pairs obtained from the Spatial Matching unit.
- ii. We propose self-supervisory scale-level pretext tasks using second-order representations and auxiliary label terms for locations/scales, *e.g.*, scale index.
- iii. We investigate various matching strategies, *i.e.*, different formulations of the objective, the use of sparsity-inducing regularization on attention scores, and the use of a balancing term on weighted average scores.
- iv. We propose a novel and effective transformer-based cross-attention matching strategy for few-shot learning, which learns object matching in pairs of images according to their respective locations and scales.

## 2 Related Work

**One- and few-shot learning** has been studied in shallow [13,14,15,16,17,18] and deep learning setting [19,1,2,20,2,3,21,22,23,4,24,10,25,26,27,28,29,30,31,32]. Early works [17,18] employ generative models. Siamese Network [19] is a two-stream CNN which can compare two streams.

Matching Network [1] introduces the concept of support set and  $L$ -way  $Z$ -shot learning protocols to capture the similarity between a query and several support images in the episodic setting which we adopt. Prototypical Net [2] computes distances between a query and prototypes of each class. Model-Agnostic Meta-Learning (MAML) [20] introduces a meta-learning model which can be considered a form of transfer learning. Such a model was extended to Gradient Modulating MAML (ModGrad) [33] to speed up the convergence. Relation

Net [3] learns the relationship between query and support images by a deep comparator that produces relation scores. SoSN [4] extends Relation Net [3] by second-order pooling. SalNet [24] is a saliency-guided end-to-end sample hallucinating model. Graph Neural Networks (GNN) [34,35,36,37,38] can also be combined with few-shot learning [21,25,26,39,40,41]. In CAN [27], PARN [28] and RENet [42], self-correlation and cross-attention are employed to boost the performance. In contrast, our work studies explicitly matching over multiple scales and locations of input patches instead of features. Moreover, our SM is the first work studying how to combine self-supervision with spatial-matching to boost the performance. SAML [29] relies on a relation matrix to improve metric measurements between local region pairs. DN4 [30] proposes the deep nearest neighbor neural network to improve the image-to-class measure via deep local descriptors. Few-shot learning can also be performed in the transductive setting [43] and applied to non-standard problems, *e.g.*, keypoint recognition [44].

**Second-order pooling** has been used in texture recognition [45] by Region Covariance Descriptors (RCD), in tracking [5] and object category recognition [8,9]. Higher-order statistics have been used for action classification [46,47], domain adaptation [48,49], few-shot learning [4,24,10,50,51], few-shot object detection [52,53,54,55] and even manifold-based incremental learning [56].

We employ second-order pooling due to its (i) permutation invariance (the ability to factor out spatial locations of feature vectors) and (ii) ability to capture second-order statistics.

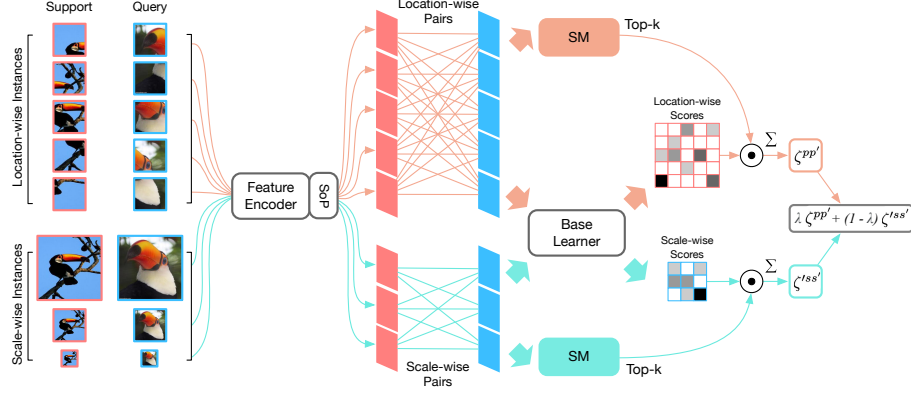
**Notations.** Let  $\mathbf{x} \in \mathbb{R}^d$  be a  $d$ -dimensional feature vector.  $\mathcal{I}_N$  stands for the index set  $\{1, 2, \dots, N\}$ . Capitalized boldface symbols such as  $\mathbf{\Phi}$  denote matrices. Lowercase boldface symbols such as  $\boldsymbol{\phi}$  denote vectors. Regular fonts such as  $\Phi_{ij}$ ,  $\phi_i$ ,  $n$  or  $Z$  denote scalars, *e.g.*,  $\Phi_{ij}$  is the  $(i, j)^{\text{th}}$  coefficient of  $\mathbf{\Phi}$ . Finally,  $\delta(\mathbf{x} - \mathbf{y}) = 1$  if  $\mathbf{x} = \mathbf{y}$  and 0 otherwise.

### 3 Approach

Although spatially-aware representations have been studied in low-level vision, *e.g.*, deblurring, they have not been studied in relation-based learning (few-shot learning). Thus, it is not obvious how to match feature sets formed from pairs of images at different locations/resolutions.

In conventional image classification, high-resolution images are known to be more informative than their low-resolution counterparts. However, extracting the discriminative information depends on the most expressive scale which varies between images. When learning to compare pairs of images (the main mechanism of relation-based few-shot learning), one has to match correctly same/related objects represented at two different locations and/or scales.

Inspired by such issues, we show the importance of spatially-aware matching across locations and scales. To this end, we investigate our strategy on classic few-shot learning pipelines such as Prototypical Net, Relation Net and Second-order Similarity Network which we refer to as  $(PN+SM)$ ,  $(RN+SM)$  and  $(SoSN+SM)$  when combined with our Spatially-aware Matching (SM) mechanism.



**Fig. 2.** The pipeline of *SoSN+SM*. We downsample input images twice (3 scales) and extract 5 sub-regions from the original image. Thus, 8 sub-images are passed through our encoder, and intermediate feature vectors are obtained and aggregated with SoP into matrices (red and blue blocks). We obtain 5+5 location-wise support/query matrices  $\Psi_k^i$  and  $\Psi_q^i$  per support/query images  $k$  and  $q$ , where  $i \in \{1, \dots, 5\}$ . We also obtain 3+3 scale-wise matrices  $\Psi_k^{i_i}$  and  $\Psi_q^{i_i}$ , where  $i \in \{1, \dots, 3\}$ . We pair them via relation operator  $\vartheta$  (e.g., concatenation) into 25 and 9 relation descriptors passed to the attention mechanism and relation network, which produces weight scores  $w_{pp'}$  ( $5 \times 5$ ) and  $w_{ss'}$  ( $3 \times 3$ ), and relation scores  $\zeta_{pp'}$  ( $5 \times 5$ ) and  $\zeta_{ss'}$  ( $3 \times 3$ ), respectively. Finally, relation scores are re-weighted by attention scores and aggregated into the final score.

### 3.1 Spatially-aware Few-shot Learning

Below, we take the *SoSN+SM* pipeline as an example to illustrate (Figure 2) how we apply SM on the SoSN few-shot learning pipeline. We firstly generate spatially-aware image sequences from each original support/query sample, and feed them into the pipeline. Each image sequence includes 8 images, i.e., 5 location-wise crops and 3 scale-wise instances. Matching over such support-query sequences requires computing correlations between  $8 \times 8 = 64$  pairs, which leads to significant training overhead when the model is trained with a large batch size. Thus, we decouple spatially-aware matching into location-wise and scale-wise matching steps to reduce the computational cost.

Specifically, let  $\mathbf{I}^1, \dots, \mathbf{I}^4$  be four corners cropped from  $\mathbf{I}$  of  $84 \times 84$  size without overlap and  $\mathbf{I}^5$  be a center crop of  $\mathbf{I}$ . We refer to such an image sequence by  $\{\mathbf{I}^p\}_{p \in \mathcal{I}_5}$ , and they are of  $42 \times 42$  resolution. Let  $\mathbf{I}^1$  be equal to the input image  $\mathbf{I}$  of  $84 \times 84$  size, and  $\mathbf{I}^2$  and  $\mathbf{I}^3$  be formed by downsampling  $\mathbf{I}$  to resolutions  $42 \times 42$  and  $21 \times 21$ . We refer to these images by  $\{\mathbf{I}^s\}_{s \in \mathcal{I}_3}$ . We pass these images via the encoding network  $f(\cdot)$ :

$$\Phi^p = f(\mathbf{I}^p; \mathcal{F}) \quad \text{and} \quad \Phi'^s = f(\mathbf{I}^s; \mathcal{F}). \quad (1)$$

where  $\mathcal{F}$  denotes parameters of encoder network,  $\Phi^p$  and  $\Phi'^s$  are feature maps at the location  $p$  and scale  $s$ , respectively. As feature maps vary in size, we apply

second-order pooling from Eq. (2) to these maps. We treat the channel mode as  $D$ -dimensional vectors and spatial modes  $H \times W$  as  $HW$  such vectors. As  $N = HW$  varies, we define it to be  $N$  for crops and  $N'^s$  for scales. Then we form

$$\Psi^p = \eta \left( \frac{1}{N} \Phi^p \Phi^{pT} \right) \quad \text{and} \quad \Psi'^s = \eta \left( \frac{1}{N'^s} \Phi'^s \Phi'^{sT} \right). \quad (2)$$

Subsequently, we pass the location- and scale-wise second-order descriptors  $\Psi^p$  and  $\Psi'^s$  to the relation network (comparator) to model image relations. For the  $L$ -way 1-shot problem, we have a support image ( $k^{\text{th}}$  index) with its image descriptors ( $\Phi_k^p, \Phi_k'^s$ ) and a query image ( $q^{\text{th}}$  index) with its image descriptors ( $\Phi_q^p, \Phi_q'^s$ ). Moreover, each of the above descriptors belong to one of  $L$  classes in the subset  $\{c_1, \dots, c_L\} \subset \mathcal{I}_C$  that forms the so-called  $L$ -way learning problem and the class subset  $\{c_1, \dots, c_L\}$  is chosen at random from  $\mathcal{I}_C \equiv \{1, \dots, C\}$ . Then, the  $L$ -way 1-shot relation requires relation scores:

$$\zeta_{kq}^{pp'} = r(\vartheta(\Psi_k^p, \Psi_q^{p'}); \mathcal{R}) \quad \text{and} \quad \zeta_{kq}^{ts s'} = r(\vartheta(\Psi_k'^s, \Psi_q'^{s'}); \mathcal{R}), \quad (3)$$

where  $\zeta$  and  $\zeta'$  are relation scores for a  $(k, q)$  image pair at locations  $(p, p')$  and scales  $(s, s')$ . Moreover,  $r(\cdot)$  is the relation network (comparator),  $\mathcal{R}$  are its trainable parameters,  $\vartheta(\cdot, \cdot)$  is the relation operator (we use concatenation along the channel mode). For  $Z$ -shot learning, this operator averages over  $Z$  second-order matrices representing the support image before concatenating with the query matrix.

A naive loss for the location- & scale-wise model is given as:

$$L = \sum_{k,q,p} w_p (\zeta_{kq}^{pp} - \delta(l_k - l_q))^2 + \lambda \sum_{k,q,s} w'_s (\zeta_{kq}^{ss} - \delta(l_k - l_q))^2, \quad (4)$$

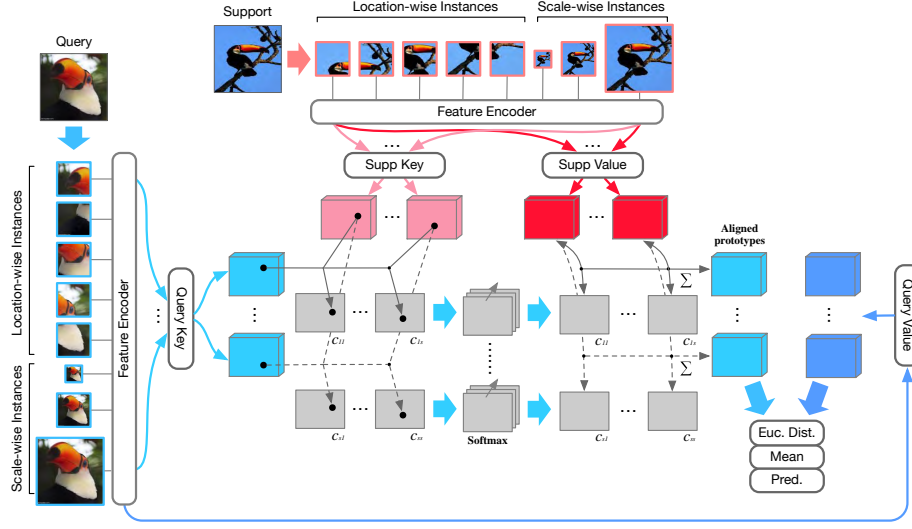
where  $l_k$  and  $l_q$  refer to labels for support and query samples,  $w_p$  and  $w'_s$  are some priors (weight preferences) w.r.t. locations  $p$  and scales  $s$ . For instance,  $w_p = 1$  if  $p = 5$  (center crop),  $w_p = 0.5$  otherwise, and  $w_s = 1/2^{s-1}$ .

A less naive formulation assumes a modified loss which performs matching between various regions and scales, defined as:

$$L = \sum_{k,q} \sum_{p,p'} (w_{pp'}^{kq})^\gamma (\zeta_{kq}^{pp'} - \delta(l_k - l_q))^2 + \lambda \sum_{k,q} \sum_{s,s'} (w_{ss'}^{kq})^\gamma (\zeta_{kq}^{ss'} - \delta(l_k - l_q))^2, \quad (5)$$

where  $w_{pp'}$  and  $w_{ss'}$  are some pair-wise priors (weight preferences) w.r.t. locations  $(p, p')$  and scales  $(s, s')$ . We favor this formulation and we strive to learn  $w_{pp'}^{kq}$  and  $w_{ss'}^{kq}$  rather than just specify rigid priors for all  $(k, q)$  support-query pairs. Finally, coefficient  $0 \leq \gamma \leq \infty$  balances the impact of re-weighting. If  $\gamma = 0$ , all weights are equal one. If  $\gamma = 0.5$ , lower weights contribute in a balanced way. If  $\gamma = 1$ , we obtain regular re-weighting. If  $\gamma = \infty$ , the largest weight wins.

**Spatially-aware Matching.** As our feature encoder processes images at different scales and locations, the model should have the ability to select the best matching locations and scales for each support-query pair. Thus, we propose a pair-wise attention mechanism to re-weight (activate/deactivate) different



**Fig. 3.** Our Spatially-aware Matching CrossTransformer (SMCT) is built upon the cross-transformer [57]. We introduce the spatially-aware (location- and scale-wise) image sequences as inputs to exploit cross-attention during feature matching.

matches withing support-query pairs when aggregating the final scores of comparator. Figure 2 shows this principle.

Specifically, as different visual concepts may be expressed by their constituent parts (mixture of objects, mixture of object parts, *etc.*), each appearing at a different location or scale, we perform a soft-attention which selects  $w_{pp'} \geq 0$  and  $w'_{ss'} \geq 0$ . Moreover, as co-occurrence representations are used as inputs to the attention network, the network selects a mixture of dominant scales and locations for co-occurring features (which may correspond to pairs of object parts).

The Spatially-aware Matching (SM) network (two convolutional blocks and an FC layer) performs the location- and scale-wise matching respectively using shared model parameters. We opt for a decoupled matching in order to reduce the training overhead. We perform  $5 \times 5 + 3 \times 3 = 34$  matches per support-query pair rather than  $5 \times 5 \times 3 \times 3 = 225$  matches but a full matching variant is plausible (and could perform better). We have:

$$w_{pp'}^{kq} = m\left(\vartheta(\Psi_k^p, \Psi_q^{p'}); \mathcal{M}\right) \quad \text{and} \quad w_{ss'}^{lkq} = m\left(\vartheta(\Psi_k'^s, \Psi_q'^{s'}); \mathcal{M}\right), \quad (6)$$

where  $m(\cdot)$  is the Spatially-aware Matching network,  $\mathcal{M}$  denotes its parameters,  $(k, q)$  are query-support sample indexes. We impose a penalty to control the sparsity of matching:

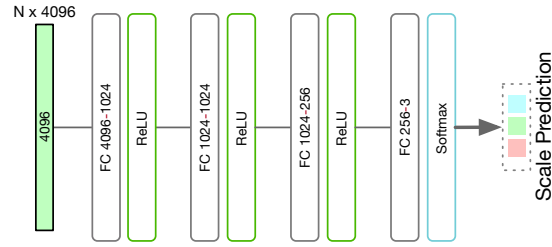
$$\Omega = \sum_{k,q} \sum_{p,p'} \left| w_{pp'}^{kq} \right| + \sum_{k,q} \sum_{s,s'} \left| w_{ss'}^{lkq} \right|. \quad (7)$$



Our spatially-aware matching network differs from the feature-based attention mechanism as we score the match between pairs of cropped/resized regions (not individual regions) to produce the attention map.

### 3.2 Self-supervised Scale and Scale Discrepancy

**Scale Discriminator.** To improve discriminative multi-scale representations, we employ self-supervision. We design a MLP-based Scale Discriminator (SD) as shown in Figure 4 which recognizes the scales of training images.



**Fig. 4.** Scale Discriminator with 3 fully-connected layers.

Specifically, we feed second-order representations to the SD module and assign labels 1, 2 or 3 for 256×256, 128×128 or 64×64 images, respectively. We apply cross-entropy loss to train the SD module and classify the scale corresponding to given second-order feature matrix.

Given  $\Psi_i^s$  which is the second-order representation of  $\mathbf{X}_i^s$ , we vectorize them via  $(:)$  and forward to the SD module to predict the scale index  $s$ . We have:

$$\mathbf{p}_i^s = \text{sd}(\Psi_{i(:)}^s; \mathcal{C}), \quad (8)$$

where  $\text{sd}(\cdot)$  refers to the scale discriminator,  $\mathcal{C}$  denotes parameters of  $\text{sd}(\cdot)$ , and  $\mathbf{p}$  are the scale prediction scores for  $\Psi$ . We go over all  $i$  corresponding to support and query images in the mini-batch and we use cross-entropy to learn the parameters of Scale Discriminator:

$$L_{sd} = - \sum_{i,s} \log \left( \frac{\exp(\mathbf{p}_i^s[s])}{\sum_{s'} \exp(\mathbf{p}_i^s[s'])} \right), \quad (9)$$

where  $s, s' \in \mathcal{I}_S$  enumerate over scale indexes.

**Discrepancy Discriminator.** As relation learning requires comparing pairs of images, we propose to model scale discrepancy between each support-query pair by assign a discrepancy label to each pair. Specifically, we assign label  $\Delta_{s,s^*} = s - s^* + 1$  where  $s$  and  $s^*$  denote the scales of a given support-query pair. Then we train so-called Discrepancy Discriminator (DD) to recognize the



discrepancy between scales. DD uses the same architecture as SD while the input dimension is doubled due to concatenated support-query pairs on input. Thus:

$$\mathbf{p}_{ij}^{s,s*} = \text{dd}(\vartheta(\boldsymbol{\Psi}_i^s, \boldsymbol{\Psi}_j^{s*}); \mathcal{D}), \quad (10)$$

where  $\text{dd}(\cdot)$  refers to scale discrepancy discriminator,  $\mathcal{D}$  are the parameters of  $dd$ ,  $\mathbf{p}_{ij}^{s,s*}$  are scale discrepancy prediction scores,  $\vartheta$  is concat. in mode 3. We go over all  $i, j$  support+query image indexes in the mini-batch and we apply the cross-entropy loss to learn the discrepancy labels:

$$L_{dd} = - \sum_{i,s} \sum_{j,s*} \log \left( \frac{\exp(\mathbf{p}_{ij}^{s,s*} [\Delta_{s,s*}])}{\sum_{s'} \sum_{s'^*} \exp(\mathbf{p}_{ij}^{s,s*} [\Delta_{s',s'^*}])} \right), \quad (11)$$

where where  $s, s', s^*, s'^* \in \mathcal{I}_S$  enumerate over scale indexes.

**Final Loss.** The total loss combines the proposed Scale Selector, Scale Discriminator and Discrepancy Discriminator:

$$\arg \min_{\mathcal{F}, \mathcal{R}, \mathcal{M}, \mathcal{C}, \mathcal{D}} \quad \alpha \Omega + L + \beta L_{sd} + \gamma L_{dd}, \quad (12)$$

where  $\alpha, \beta, \gamma$  are the hyper-parameters that control the impact of the regularization and each individual loss component.

### 3.3 Transformer-based Spatially-aware Pipeline

Our SM network can be viewed as an instance of attention, whose role is to re-weight numbers of spatial pairs to improve the discriminative relation learning between support and query samples. Recently, transformers have proven very effective in learning the discriminative representations in both natural language processing and computer vision tasks. Inspired by the self-attention [57], which can naturally be used to address the feature matching problem, we further develop a novel Spatially-aware Matching CrossTransformer (SmCT) to match location- and scale-wise support-query patches in few-shot learning. Figure 3 shows the architecture of SmCT, which consists of 4 heads, namely the support key and value heads, and the query key and value heads. In contrast to using SM in classic pipelines, where we decoupled the matching into location-wise and scale-wise steps, all possible matching combinations are considered in the SmCT pipeline. Thus, we do not use symbols  $\mathbf{I}^p$  and  $\mathbf{I}^s$  in this section. The image sequence is simply given by  $\{\mathbf{I}^s\}_{s \in \mathcal{I}_8}$ .

We pass the spatial sequences of support and query samples  $\mathbf{I}_k^s$  and  $\mathbf{I}_q^s$  into the backbone and obtain  $\boldsymbol{\Phi}_k^s, \boldsymbol{\Phi}_q^s \in \mathbb{R}^{D \times N}$ . Due to different scales of feature maps for  $s=6, 7, 8$ , we downsample larger feature maps and obtain  $N'=10 \times 10$  feature vectors. We feed them into the key/value heads to obtain keys  $\mathbf{K}_k^s, \mathbf{K}_q^s \in \mathbb{R}^{d_k \times N'}$  and values  $\mathbf{V}_k^s, \mathbf{V}_q^s \in \mathbb{R}^{d_v \times N'}$ . Then we multiply support-query spatial key pairs followed by SoftMax in order to obtain the normalized cross-attention scores  $\mathbf{C}_{s\bar{s}} = \text{SoftMax}(\mathbf{K}_k^{sT} \mathbf{K}_q^{\bar{s}}) \in \mathbb{R}^{N' \times N'}$ , which are used as correlations in aggregation

**Table 1.** Evaluations on *mini*-ImageNet and *tiered*-ImageNet (5-way acc. given).

Model	Backbone	<i>mini</i> -ImageNet		<i>tiered</i> -ImageNet	
		1-shot	5-shot	1-shot	5-shot
<i>MN</i>	[1]	-	43.56 ± 0.84	55.31 ± 0.73	-
<i>PN</i>	[2]	Conv-4-64	49.42 ± 0.78	68.20 ± 0.66	53.31 ± 0.89
<i>MAML</i>	[20]	Conv-4-64	48.70 ± 1.84	63.11 ± 0.92	51.67 ± 1.81
<i>RN</i>	[3]	Conv-4-64	50.44 ± 0.82	65.32 ± 0.70	54.48 ± 0.93
<i>GNN</i>	[21]	Conv-4-64	50.30	66.40	-
<i>MAML++</i>	[58]	Conv-4-64	52.15 ± 0.26	68.32 ± 0.44	-
<i>SalNet</i>	[24]	Conv-4-64	57.45 ± 0.86	72.01 ± 0.75	-
<i>SoSN</i>	[4]	Conv-4-64	52.96 ± 0.83	68.63 ± 0.68	-
<i>TADAM</i>	[59]	ResNet-12	58.50 ± 0.30	76.70 ± 0.30	-
<i>MetaOpt</i>	[60]	ResNet-12	61.41 ± 0.61	77.88 ± 0.46	65.99 ± 0.72
<i>DeepEMD</i>	[61]	ResNet-12	65.91 ± 0.82	82.41 ± 0.56	-
<i>PN+SM</i>		Conv-4-64	52.01 ± 0.80	69.92 ± 0.67	54.37 ± 0.82
<i>RN+SM</i>		Conv-4-64	54.99 ± 0.87	68.57 ± 0.63	57.01 ± 0.91
<i>SoSN+SM</i>		Conv-4-64	57.11 ± 0.84	71.98 ± 0.63	61.58 ± 0.90
<i>SoSN+SM</i>		ResNet-12	62.36 ± 0.85	78.86 ± 0.63	66.35 ± 0.91
<i>DeepEMD+SM</i>		ResNet-12	<b>66.93 ± 0.84</b>	<b>84.34 ± 0.61</b>	<b>70.19 ± 0.89</b>
					<b>86.98 ± 0.74</b>

**Table 2.** Evaluations on the Open MIC dataset (Protocol I) (1-shot learning accuracy). (<http://users.cecs.anu.edu.au/~koniusz/openmic-dataset>).

Model	way	p1→p2	p1→p3	p1→p4	p2→p1	p2→p3	p2→p4	p3→p1	p3→p2	p3→p4	p4→p1	p4→p2	p4→p3
<i>RN</i> [3]	5	71.1	53.6	63.5	47.2	50.6	68.5	48.5	49.7	68.4	45.5	70.3	50.8
<i>SoSN</i> <sub>(84)</sub> [4]		81.4	65.2	75.1	60.3	62.1	77.7	61.5	82.0	78.0	59.0	80.8	62.5
<i>SoSN</i> <sub>(256)</sub>		84.1	69.3	82.5	64.9	66.9	82.8	65.8	85.1	81.1	65.1	83.9	66.6
<i>SoSN+SM</i>		<b>85.6</b>	<b>73.6</b>	<b>85.0</b>	<b>67.7</b>	<b>69.6</b>	<b>83.1</b>	<b>68.2</b>	<b>86.9</b>	<b>82.9</b>	<b>67.4</b>	<b>84.7</b>	<b>68.4</b>
<i>RN</i> [3]	20	40.1	30.4	41.4	23.5	26.4	38.6	26.2	25.8	46.3	23.1	43.3	27.7
<i>SoSN</i> <sub>(84)</sub> [4]		61.5	42.5	61.0	36.1	38.3	56.3	38.7	59.9	59.4	37.4	59.0	38.6
<i>SoSN</i> <sub>(256)</sub>		63.9	49.2	65.9	43.1	44.6	62.6	44.2	63.9	64.1	43.8	63.1	44.3
<i>SoSN+SM</i>		<b>65.5</b>	<b>51.1</b>	<b>67.6</b>	<b>45.2</b>	<b>46.3</b>	<b>64.5</b>	<b>46.3</b>	<b>66.2</b>	<b>67.0</b>	<b>45.3</b>	<b>65.8</b>	<b>47.1</b>
<i>RN</i> [3]	30	37.8	27.3	39.8	22.1	24.3	36.7	24.5	23.7	44.2	21.4	41.5	25.5
<i>SoSN</i> <sub>(84)</sub> [4]		60.6	40.1	58.3	34.5	35.1	54.2	36.8	58.6	56.6	35.9	57.1	37.1
<i>SoSN</i> <sub>(256)</sub>		61.7	46.6	64.1	41.4	40.9	60.3	41.6	61.0	60.0	42.4	61.2	41.4
<i>SoSN+SM</i>		<b>62.6</b>	<b>47.3</b>	<b>65.2</b>	<b>41.9</b>	<b>41.7</b>	<b>61.5</b>	<b>43.1</b>	<b>61.8</b>	<b>61.0</b>	<b>43.1</b>	<b>62.1</b>	<b>42.3</b>

p1: shn+hon+clv, p2: clk+gls+scl, p3: sci+nat, p4: shx+rlc.

Notation  $x \rightarrow y$  means training on exhibition  $x$  and testing on  $y$ .

of support values w.r.t. each location and scale. We obtain the aligned spatially-aware prototypes  $\tilde{\mathbf{V}}_k^s \in \mathbb{R}^{d_v \times N'}$ :

$$\tilde{\mathbf{V}}_k^s = \sum_{\tilde{s}} \mathbf{V}_k^s \mathbf{C}_{s\tilde{s}}. \quad (13)$$

We measure Euclidean distances between the aligned prototypes and corresponding query values, which act as the final similarity between sample  $\mathbf{I}_k$  and  $\mathbf{I}_q$ :

$$L = \sum_{k,q} (\zeta_{kq} - \delta(l_k - l_q))^2 \quad \text{where} \quad \zeta_{kq} = \sum_s \|\tilde{\mathbf{V}}_k^s - \tilde{\mathbf{V}}_q^s\|_F^2. \quad (14)$$

## 4 Experiments

Below we demonstrate usefulness of our proposed Spatial- and Scale-matching Network by evaluations (one- and few-shot protocols) on *mini*-ImageNet [1], *tiered*-ImageNet [63], Meta-Dataset [62] and fine-grained datasets.

**Table 3.** The experiments on selected subsets of Meta-Dataset (train-on-ILSVRC setting). We compare our Spatially-aware Matching pipelines with recent baseline models. Following the training steps in [57], we also apply the SimCLR episodes to train our SmCT (which uses the ResNet-34 backbone).

		ImageNet	Aircraft	Bird	DTD	Flower	Avg
$k$ -NN	[62]	41.03	46.81	50.13	66.36	83.10	57.49
MN	[1]	45.00	48.79	62.21	64.15	80.13	60.06
PN	[2]	50.50	53.10	68.79	66.56	85.27	64.84
RN	[3]	34.69	40.73	49.51	52.97	68.76	49.33
SoSN	[4]	50.67	54.13	69.02	66.49	87.21	65.50
CTX	[57]	62.76	79.49	80.63	75.57	95.34	78.76
$PN+SM$		53.12	57.06	72.01	70.23	88.96	68.28
$RN+SM$		41.07	46.03	53.24	58.01	72.98	54.27
$SoSN+SM$		52.03	56.68	70.89	69.03	90.36	67.80
$SMCT$		<b>64.12</b>	<b>81.03</b>	<b>82.98</b>	<b>76.95</b>	<b>96.45</b>	<b>80.31</b>

**Table 4.** Evaluations on fine-grained recognition datasets, Flower-102, CUB-200-2011 and Food-101 (5-way acc. given). See [3,4] for details of baselines listed in this table.

	Flower-102		CUB-200-2011		Food-101	
Model	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
$PN$	62.81	82.11	37.42	51.57	36.71	53.43
$RN$	68.52	81.11	40.36	54.21	36.89	49.07
$SoSN$	76.27	88.55	47.45	63.53	43.12	58.13
$RN+SM$	71.69	84.45	45.79	58.67	45.31	55.67
$SoSN+SM$	<b>81.69</b>	<b>91.21</b>	<b>54.24</b>	<b>70.85</b>	<b>48.86</b>	<b>63.67</b>

**Setting.** We use the standard  $84 \times 84$  image resolution for *mini*-ImageNet and fine-grained datasets, and  $224 \times 224$  resolution for Meta-Dataset for fair comparisons. Hyper-parameter  $\alpha$  is set to 0.001 while  $\beta$  is set to 0.1 via cross-validation on *mini*-ImageNet. Note that SmCT uses ResNet-34 backbone on Meta-Dataset, and ResNet-12 on *mini*-ImageNet and *tiered*-ImageNet. The total number of training episode is 200000, and the number of testing episode is 1000.

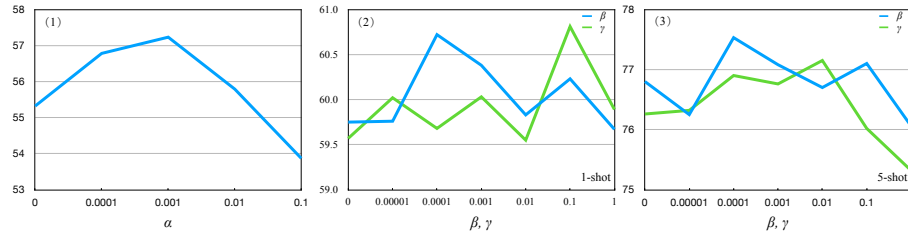
#### 4.1 Datasets

Below, we describe our experimental setup and datasets.

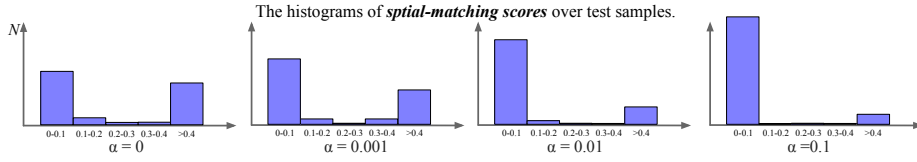
**mini-ImageNet** [1] consists of 60000 RGB images from 100 classes. We follow the standard protocol (64/16/20 classes for training/validation/testing).

**tiered-ImageNet** [63] consists of 608 classes from ImageNet. We follow the protocol that uses 351 base classes, 96 validation and 160 test classes.

**Open MIC** is the Open Museum Identification Challenge (Open MIC) [64], a dataset with photos of various museum exhibits, *e.g.*, paintings, timepieces, sculptures, glassware, relics, science exhibits, natural history pieces, ceramics, pottery, tools and indigenous crafts, captured from 10 museum spaces according to which this dataset is divided into 10 subproblems. In total, it has 866 diverse



**Fig. 5.** Accuracy w.r.t.  $\alpha$  (subfigure 1),  $\beta$  (the blue curve in subfigure 2&3) and  $\gamma$  (the red curve in subfigure 2&3), which control the impact of SD and DD.



**Fig. 6.** The histograms of spatial-matching scores with different  $\alpha$  values, which demonstrate how  $\Omega$  induces sparsity and improves the performance.

classes and 1–20 images per class. We combine  $(shn+hon+clv)$ ,  $(clk+gls+scl)$ ,  $(sci+nat)$  and  $(shx+rlc)$  into subproblems  $p1, \dots, p4$ . We form 12 possible pairs in which subproblem  $x$  is used for training and  $y$  for testing ( $x \rightarrow y$ ).

**Meta-Dataset** [62] is a recently proposed benchmark consisting of 10 publicly available datasets to measure the generalized performance of each model. The Train-on-ILSVRC setting means that the model is merely trained via ImageNet training data, and then evaluated on the test data of remaining 10 datasets. In this paper, we follow the Train-on-ILSVRC setting. We choose 5 datasets to measure the overall performance.

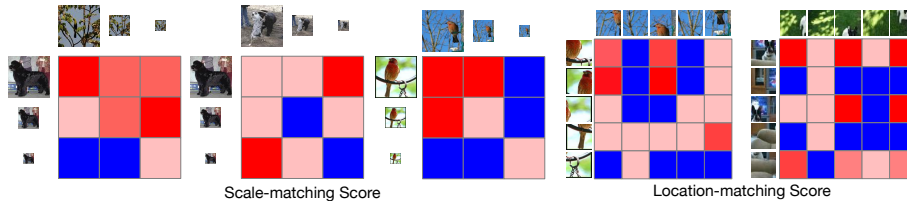
**Flower-102** [65] contains 102 fine-grained classes of flowers. Each class has 40–258 images. We randomly select 80/22 classes for training/testing.

**Caltech-UCSLD-Birds 200-2011 (CUB-200-2011)** [66] has 11788 images of 200 fine-grained bird species, 150 classes a for training and the rest for testing.

**Food-101** [67] has 101000 images of 101 fine-grained classes, and 1000 images per category. We choose 80/21 classes for training/testing.

## 4.2 Performance Analysis

Table 1 shows our evaluations on *mini*-ImageNet [1] and *tiered*-ImageNet. Our approach achieves the state-of-the-art performance among all methods based on both ‘Conv-4-64’ and ‘ResNet-12’ backbones at  $84 \times 84$  input resolution. By adding SM (Eq. (3)) and self-supervisory loss (Eq. (9)), we achieve 57.11 and 71.98 scores for 1- and 5-shot protocols with the SoSN baseline (Conv-4-64 backbone), and 66.93 and 85.34 with the DeepEMD baseline (ResNet-12 backbone)



**Fig. 7.** The histograms of spatial-matching scores with different  $\alpha$  values, which demonstrate how  $\Omega$  induces sparsity and improves the performance.

**Table 5.** Ablation studies w.r.t. the location- and scale-wise inputs (no matching used). Region crops and scale selection is done either in the image space (*img*) or on the feature maps (*feat.*), (5-way 1-shot accuracy, ‘SoSN+SM’ with ResNet-12 backbone).

Scale-wise ( <i>img</i> )	Scale-wise ( <i>feat.</i> )	Loc.-wise ( <i>img</i> )	Loc.-wise ( <i>feat.</i> )
59.51	58.54	60.14	58.16

on *mini*-ImageNet. The average improvements gained from SM are 4% and 3.5% with the Conv-4-64 backbone, 1% and 1.9% with the ResNet-12 backbone for 1- and 5-shot protocols. These results outperform results in previous works, which strongly supports the benefit of our spatially-aware matching. On *tiered*-ImageNet, our proposed model obtains 3% and 4% improvement for 1-shot and 5-shot protocols with both the Conv-4-64 and the ResNet-12 backbones. Our performance on *tiered*-ImageNet is also better than in previous works. However, we observed that SmCT with the ResNet-12 backbone does not perform as strongly on the above two datasets. A possible reason is that the complicated transformer architecture overfits when the scale of training dataset is small (in contrast to its performance on Meta-Dataset).

We also evaluate our proposed network on the Open MIC dataset [64]. Table 2 shows that our proposed method performs better than baseline models. By adding SM and self-supervised discriminators into baseline models, the accuracies are improved by 1.5%  $\sim$  3.0%.

Table 3 presents results on the Train-on-ISLVR setting on 5 of 10 datasets of Meta-Dataset. Applying SM on classic simple baseline few-shot learning methods brings impressive improvements on all datasets. Furthermore, our SmCT achieves the state-of-the-art results compared to previous methods. This observation is consistent with our analysis that transformer-based networks are likely to be powerful when being trained on a large-scale dataset.

Table 4 shows that applying SM on classic baseline models significantly outperforms others on fine-grained Flower-102, CUB-200-2011 and Food-101.

**Spatially-aware Matching.** The role of SM is to learn matching between regions and scales of support-query pairs. As shown in Table 5, using the Spatially-aware Matching Network can further improve results for both 1-shot and 5-shot learning on *mini*-ImageNet by 1.5% and 1.0%, respectively. The results on

Flower-102, Food-101 yield  $\sim 1.5\%$  gain for SmN. Figure 5 shows the impact of  $\alpha$ ,  $\beta$  and  $\gamma$  on the performance. Figure 6 verifies the usefulness of  $\Omega$  regularization term. To this end, we show histograms of spatial-matching scores to demonstrate how  $\alpha$  affects the results *vs.* sparsity of matching scores. For instance, one can see the results are the best for moderate  $\alpha = 0.001$ , and the bin containing null counts also appears larger compared to  $\alpha = 0$  (desired behavior). Figure 7 visualizes the matching scores. We randomly sample support-query pairs to show that visually related patches have higher match scores (red) than unrelated pairs (blue). Do note the location- and scale- discrimination is driven by matching both similar and dissimilar support-query pairs driven by the relation loss.

**Self-supervised Discriminators.** Pretext tasks are known for their ability to boost the performance of image classification due to additional regularization they provide. Applying Scale Discriminator (SD) and Discrepancy Discriminator (DD) is an easy and cheap way to boost the representational power of our network. Pretext tasks do not affect the network complexity or training times. According to our evaluations, the SD improves the 1-shot accuracy by 1.1% and 5-shot accuracy by 1.0%, while the DD improves the accuracy by 1.2% and 1.1% 0.9% for 1- and 5-shot respectively on the *mini*-ImageNet dataset.

In summary, without any pre-training, combining our Spatially-aware Matching strategy brings consistent improvements on the Conv-4-64 and ResNet-12 backbones and various few-shot learning methods, with the overall accuracy outperforming state-of-the-art methods on all few-shot learning datasets. Our novel transformer-based SmCT model also performs strongly on the recently proposed Meta-Dataset, which further supports the usefulness of spatial modeling.

## 5 Conclusions

We have proposed the Spatially-aware Matching strategy for few-shot learning, which is shown to be orthogonal to the choice of baseline models and/or backbones. Our novel feature matching mechanism helps models learn a more accurate similarity due to matching multiple locations and coarse-to-fine scales of support-query pairs. We show how to leverage a self-supervisory pretext task based on spatial labels. We have also proposed a novel Spatially-aware Matching CrossTransformer to perform matching via the recent popular self- and cross-attention strategies. Our experiments demonstrate the usefulness of the proposed SM strategy and SmCT in capturing accurate image relations. Combining our SM with various baselines outperforms previous works in the same testbed. SmCT achieves SOTA results on large-scale training data.

**Acknowledgements** This work is supported by National Natural Science Foundation of China (No. 62106282), and Young Elite Scientists Sponsorship Program by CAST (No. 2021-JCJQ-QT-038). Code: <https://github.com/HongguangZhang/smfs1-master>.

## References

1. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. In: NIPS. (2016) 3630–3638 [2](#), [3](#), [10](#), [11](#), [12](#)
2. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: NeurIPS. (2017) 4077–4087 [2](#), [3](#), [10](#), [11](#)
3. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. arXiv preprint arXiv:1711.06025 (2017) [2](#), [3](#), [4](#), [10](#), [11](#)
4. Zhang, H., Koniusz, P.: Power normalizing second-order similarity network for few-shot learning. In: WACV, IEEE (2019) 1185–1193 [2](#), [3](#), [4](#), [10](#), [11](#)
5. Porikli, F., Tuzel, O.: Covariance tracker. CVPR (2006) [2](#), [4](#)
6. Guo, K., Ishwar, P., Konrad, J.: Action recognition from video using feature covariance matrices. Trans. Img. Proc. **22** (2013) 2479–2494 [2](#)
7. Carreira, J., Caseiro, R., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. ECCV (2012) [2](#)
8. Koniusz, P., Yan, F., Gosselin, P.H., Mikolajczyk, K.: Higher-order occurrence pooling for bags-of-words: Visual concept detection. IEEE Trans. Pattern Anal. Mach. Intell. **39** (2017) 313–326 [2](#), [4](#)
9. Koniusz, P., Zhang, H., Porikli, F.: A deeper look at power normalizations. In: CVPR. (2018) 5774–5783 [2](#), [4](#)
10. Wertheimer, D., Hariharan, B.: Few-shot learning with localization in realistic settings. In: CVPR. (2019) 6558–6567 [2](#), [3](#), [4](#)
11. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. CVPR **2** (2006) 2169–2178 [3](#)
12. Yang, J., Yu, K., Gong, Y., Huang, T.S.: Linear spatial pyramid matching using sparse coding for image classification. CVPR (2009) 1794–1801 [3](#)
13. Miller, E.G., Matsakis, N.E., Viola, P.A.: Learning from one example through shared densities on transforms. CVPR **1** (2000) 464–471 [3](#)
14. Li, F.F., VanRullen, R., Koch, C., Perona, P.: Rapid natural scene categorization in the near absence of attention. Proceedings of the National Academy of Sciences **99** (2002) 9596–9601 [3](#)
15. Fink, M.: Object classification from a single example utilizing class relevance metrics. NIPS (2005) 449–456 [3](#)
16. Bart, E., Ullman, S.: Cross-generalization: Learning novel classes from a single example by feature replacement. CVPR (2005) 672–679 [3](#)
17. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Trans. Pattern Anal. Mach. Intell. **28** (2006) 594–611 [3](#)
18. Lake, B.M., Salakhutdinov, R., Gross, J., Tenenbaum, J.B.: One shot learning of simple visual concepts. CogSci (2011) [3](#)
19. Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: ICML Deep Learning Workshop. Volume 2. (2015) [3](#)
20. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML. (2017) 1126–1135 [3](#), [10](#)
21. Garcia, V., Bruna, J.: Few-shot learning with graph neural networks. arXiv preprint arXiv:1711.04043 (2017) [3](#), [4](#), [10](#)
22. Rusu, A.A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S., Hadsell, R.: Meta-learning with latent embedding optimization. arXiv preprint arXiv:1807.05960 (2018) [3](#)



23. Gidaris, S., Komodakis, N.: Dynamic few-shot visual learning without forgetting. In: CVPR. (2018) 4367–4375 [3](#)
24. Zhang, H., Zhang, J., Koniusz, P.: Few-shot learning via saliency-guided hallucination of samples. In: CVPR. (2019) 2770–2779 [3](#), [4](#), [10](#)
25. Kim, J., Kim, T., Kim, S., Yoo, C.D.: Edge-labeling graph neural network for few-shot learning. In: CVPR. (2019) [3](#), [4](#)
26. Gidaris, S., Komodakis, N.: Generating classification weights with gnn denoising autoencoders for few-shot learning. In: CVPR. (2019) [3](#), [4](#)
27. Hou, R., Chang, H., Ma, B., Shan, S., Chen, X.: Cross attention network for few-shot classification. NeurIPS **32** (2019) [3](#), [4](#)
28. Wu, Z., Li, Y., Guo, L., Jia, K.: Parn: Position-aware relation networks for few-shot learning. In: ICCV. (2019) 6659–6667 [3](#), [4](#)
29. Hao, F., He, F., Cheng, J., Wang, L., Cao, J., Tao, D.: Collect and select: Semantic alignment metric learning for few-shot learning. In: CVPR. (2019) 8460–8469 [3](#), [4](#)
30. Li, W., Wang, L., Xu, J., Huo, J., Gao, Y., Luo, J.: Revisiting local descriptor based image-to-class measure for few-shot learning. In: CVPR. (2019) 7260–7268 [3](#), [4](#)
31. Zhang, H., Li, H., Koniusz, P.: Multi-level second-order few-shot learning. IEEE Transactions on Multimedia (2022) [3](#)
32. Ni, G., Zhang, H., Zhao, J., Xu, L., Yang, W., Lan, L.: Anf: Attention-based noise filtering strategy for unsupervised few-shot classification. In: Pacific Rim International Conference on Artificial Intelligence, Springer (2021) 109–123 [3](#)
33. Simon, C., Koniusz, P., Nock, R., Harandi, M.: On modulating the gradient for meta-learning. In: ECCV. (2020) [3](#)
34. Sun, K., Koniusz, P., Wang, Z.: Fisher-Bures adversary graph convolutional networks. UAI **115** (2019) 465–475 [4](#)
35. Zhu, H., Koniusz, P.: Simple spectral graph convolution. In: ICLR. (2021) [4](#)
36. Zhu, H., Sun, K., Koniusz, P.: Contrastive laplacian eigenmaps. In: NeurIPS. (2021) 5682–5695 [4](#)
37. Zhang, Y., Zhu, H., Meng, Z., Koniusz, P., King, I.: Graph-adaptive rectified linear unit for graph neural networks. In: TheWebConf (WWW), ACM (2022) 1331–1339 [4](#)
38. Zhang, Y., Zhu, H., Song, Z., Koniusz, P., King, I.: COSTA: covariance-preserving feature augmentation for graph contrastive learning. In: KDD, ACM (2022) 2524–2534 [4](#)
39. Wang, L., Liu, J., Koniusz, P.: 3d skeleton-based few-shot action recognition with JEANIE is not so naïve. arXiv preprint arXiv: 2112.12668 (2021) [4](#)
40. Wang, L., Koniusz, P.: Temporal-viewpoint transportation plan for skeletal few-shot action recognition. ACCV (2022) [4](#)
41. Wang, L., Koniusz, P.: Uncertainty-DTW for time series and sequences. ECCV (2022) [4](#)
42. Kang, D., Kwon, H., Min, J., Cho, M.: Relational embedding for few-shot classification. In: ICCV. (2021) 8822–8833 [4](#)
43. Zhu, H., Koniusz, P.: EASE: Unsupervised discriminant subspace learning for transductive few-shot learning. CVPR (2022) [4](#)
44. Lu, C., Koniusz, P.: Few-shot keypoint detection with uncertainty learning for unseen species. CVPR (2022) [4](#)
45. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. ECCV (2006) [4](#)
46. Koniusz, P., Cherian, A., Porikli, F.: Tensor representations via kernel linearization for action recognition from 3d skeletons. In: ECCV, Springer (2016) 37–53 [4](#)

47. Koniusz, P., Wang, L., Cherian, A.: Tensor representations for action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **44** (2022) 648–665 [4](#)
48. Koniusz, P., Tas, Y., Porikli, F.: Domain adaptation by mixture of alignments of second-or higher-order scatter tensors. In: *CVPR*. Volume 2. (2017) [4](#)
49. Tas, Y., Koniusz, P.: CNN-based action recognition and supervised domain adaptation on 3d body skeletons via kernel feature maps. In: *BMVC*, BMVA Press (2018) 158 [4](#)
50. Zhang, H., Koniusz, P., Jian, S., Li, H., Torr, P.H.S.: Rethinking class relations: Absolute-relative supervised and unsupervised few-shot learning. In: *CVPR*. (2021) 9432–9441 [4](#)
51. Koniusz, P., Zhang, H.: Power normalizations in fine-grained image, few-shot image and graph classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **44** (2022) 591–609 [4](#)
52. Zhang, S., Luo, D., Wang, L., Koniusz, P.: Few-shot object detection by second-order pooling. In: *ACCV*. Volume 12625 of *Lecture Notes in Computer Science*., Springer (2020) 369–387 [4](#)
53. Yu, X., Zhuang, Z., Koniusz, P., Li, H.: 6DoF object pose estimation via differentiable proxy voting regularizer. In: *BMVC*, BMVA Press (2020) [4](#)
54. Zhang, S., Wang, L., Murray, N., Koniusz, P.: Kernelized few-shot object detection with efficient integral aggregation. In: *CVPR*. (2022) 19207–19216 [4](#)
55. Zhang, S., Murray, N., Wang, L., Koniusz, P.: Time-rEversed diffuSiON tEnSor Transformer: A new TENET of Few-Shot Object Detection. In: *ECCV*. (2022) [4](#)
56. Simon, C., Koniusz, P., Harandi, M.: On learning the geodesic path for incremental learning. In: *CVPR*. (2021) 1591–1600 [4](#)
57. Doersch, C., Gupta, A., Zisserman, A.: Crosstransformers: spatially-aware few-shot transfer. *arXiv preprint arXiv:2007.11498* (2020) [7](#), [9](#), [11](#)
58. Antoniou, A., Edwards, H., Storkey, A.: How to train your maml. *arXiv preprint arXiv:1810.09502* (2018) [10](#)
59. Oreshkin, B., Lopez, P.R., Lacoste, A.: Tadaml: Task dependent adaptive metric for improved few-shot learning. In: *Advances in Neural Information Processing Systems*. (2018) 721–731 [10](#)
60. Lee, K., Maji, S., Ravichandran, A., Soatto, S.: Meta-learning with differentiable convex optimization. In: *CVPR*. (2019) 10657–10665 [10](#)
61. Zhang, C., Cai, Y., Lin, G., Shen, C.: Deepemd: Few-shot image classification with differentiable earth mover’s distance and structured classifiers. In: *CVPR*. (2020) 12203–12213 [10](#)
62. Triantafillou, E., Zhu, T., Dumoulin, V., Lamblin, P., Evci, U., Xu, K., Goroshin, R., Gelada, C., Swersky, K., Manzagol, P.A., et al.: Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096* (2019) [10](#), [11](#), [12](#)
63. Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J.B., Larochelle, H., Zemel, R.S.: Meta-learning for semi-supervised few-shot classification. In: *ICLR*. (2018) [10](#), [11](#)
64. Koniusz, P., Tas, Y., Zhang, H., Harandi, M., Porikli, F., Zhang, R.: Museum exhibit identification challenge for the supervised domain adaptation and beyond. *ECCV* (2018) 788–804 [11](#), [13](#)
65. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*. (2008) [12](#)

- 66. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology (2011) [12](#)
- 67. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: ECCV. (2014) [12](#)