DeepEMD: Few-Shot Image Classification with Differentiable Earth Mover's Distance and Structured Classifiers

Chi Zhang[1], Yujun Cai[1], Guosheng Lin[1]*, Chunhua Shen[2]
[1] Nanyang Technological University, Singapore [2] The University of Adelaide, Australia
E-mail: chi007@e.ntu.edu.sg, gslin@ntu.edu.sg

Abstract

In this paper, we address the few-shot classification task from a new perspective of optimal matching between image regions. We adopt the Earth Mover's Distance (EMD) as a metric to compute a structural distance between dense image representations to determine image relevance. The EMD generates the optimal matching flows between structural elements that have the minimum matching cost, which is used to represent the image distance for classification. To generate the important weights of elements in the EMD formulation, we design a cross-reference mechanism, which can effectively minimize the impact caused by the cluttered background and large intra-class appearance variations. To handle k-shot classification, we propose to learn a structured fully connected layer that can directly classify dense image representations with the EMD. Based on the implicit function theorem, the EMD can be inserted as a layer into the network for end-to-end training. We conduct comprehensive experiments to validate our algorithm and we set new state-of-the-art performance on four popular few-shot classification benchmarks, namely miniImageNet, tieredImageNet, Fewshot-CIFAR100 (FC100) and Caltech-UCSD Birds-200-2011 (CUB).

1. Introduction

Deep neural networks have achieved breakthroughs in a wide range of visual understanding tasks in the past few years. However, its data-driven nature often makes it struggle when no sufficiently large labeled training data is available. Meta learning, on the other hand, is proposed to learn a model that can quickly generalize to new tasks with minor adaption steps. One of the most well-studied test-bed for meta-learning algorithms is few-shot image classification, which aims to perform classification on new image categories with only a small amount of labeled training data.

To address this problem, a line of the previous literature adopts metric-based methods [30, 43, 63, 66, 67, 73] that learn to represent image data in an appropriate feature space and use a distance function to predict image labels. Following the formulation of the standard image classification networks [16, 24, 62], metric-based methods often employ a convolution neural network to learn image feature representations, but replace the fully connected layer with a distance function, *e.g.*, *cosine* distance and Euclidean distance. Such distance functions directly compute the distance between the embeddings of the test images and training images for classification, which bypasses the difficult optimization problem in learning a classifier for the few-shot setting. The network is trained by sampling from a distribution of tasks, in the hopes of acquiring generalization ability to unseen but similar tasks.

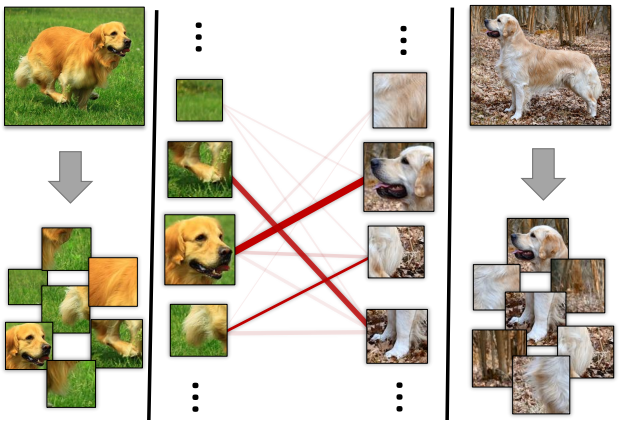

Figure 1: Illustration of using Earth Mover's Distance for one-shot image classification. Our algorithm uses optimal matching cost between image regions to represent the image distance.

Despite their promising results, we observe that the cluttered background and large intra-class appearance variations may drive the image-level embeddings from the same category far apart in a given metric space. Although the problem can be alleviated by the neural network under the fully supervised training, thanks to the activation functions and abundant training images, it is almost inevitably am-

*Corresponding author: G. Lin (e-mail: gslin@ntu.edu.sg)

plified in low-data regimes and thus negatively impacts the image classification. Moreover, a mixed global representation destroys image structures and loses local features. Local features can provide discriminative and transferable information across categories, which can be important for image classification in the few-shot scenario. Therefore, a desirable metric-based algorithm should have the ability to utilize the local discriminative representations for metric learning and minimize the impact caused by the irrelevant regions.

A natural way to compare two complex structured representations is to compare their building blocks. The difficulty lies in that we do not have their correspondence supervision for training and not all building elements can always find their counterparts in the other structures. To solve the problems above, in this paper, we formalize the few-shot classification as an instance of optimal matching, and we propose to use the optimal matching cost between two structures to represent their similarity. Given the feature representations generated by two images, we adopt the Earth Mover's Distance (EMD) [50] to compute their structural similarity. EMD is the metric for computing distance between structural representations, which was originally proposed for image retrieval. Given the distance between all element pairs, EMD can acquire the optimal matching flows between two structures that have the minimum cost. It can also be interpreted as the minimum cost to reconstruct a structure representation with the other one. An illustration of our motivation is shown in Fig. 1. EMD has the formulation of the transportation problem [17] and the global minimum can be achieved by solving a Linear Programming problem. To embed the optimization problem into the model for end-to-end training, we can apply the implicit function theorem [3, 8, 22] to form the Jacobian matrix of the optimal optimization variable with respect to the problem parameters [3].

An important problem parameter in the EMD formulation is the weight of each element. Elements with large weights generate more matching flows and thus contribute more to the overall distance. Ideally, the algorithm should have the flexibility to assign less weight on irrelevant regions such that they contribute less to the overall distance no matter which elements they match with. To achieve this goal, we propose a cross-reference mechanism to determine the importance of the elements. In our cross-reference mechanism, each node is determined by comparing it with the global statistics of the other structure. This aims to give less weight to the high-variance background regions and the object parts that are not co-occurrent in two images.

In the k-shot setting where multiple support images are presented, we propose to learn a structured fully connected (FC) layer as the classifier for classification to make use of the increasing number of training images. The structured FC layer includes a group of learnable vectors for each class. At inference time, we use the EMD to compute the distance between the image embeddings and the learnable vector set in each class for classification. The structured FC is an extension of the standard fully connected layer in that it replaces dot product operations between vectors with EMD function between vector sets such that the structured FC layer could directly classify feature maps. The structured FC layer can also be interpreted as learning a prototype embedding from a dummy image for each category such that the test images can be matched with them for classification.

To validate our algorithm, we conduct extensive experiments on multiple datasets to demonstrate the effectiveness of our algorithm. Our main contributions are summarized as follows:

- We propose to formalize the few-shot image classification as an optimal matching problem and adopt the Earth Mover's Distance as the distance metric between structured representations. The EMD layer can be embedded into the network for end-to-end training.
- We propose a cross-reference mechanism to generate the weights of elements in the EMD formulation, which can effectively reduce the noise introduced by the background regions in the images.
- We propose to learn a structured fully connected layer in the k-shot settings, which could directly classify the structural representations of an image using the Earth Mover's Distance.
- Experiments on four popular few-shot classification benchmark datasets—miniImagenet, tieredImagenet, FC100 and CUB, show that our algorithm on both 1-shot and 5-shot tasks significantly outperforms the baseline methods and achieves new state-of-the-art performance on all of them.

2. Related Work

Few-Shot Learning. There are two main streams in the few-shot classification literature, metric-based approaches and optimization-based approaches. Optimization-based methods [2, 5, 9–11, 18, 26, 31–33, 36, 38, 40–42, 44, 46, 47, 54, 55, 58, 64, 79] target at effectively adapting model parameters to new tasks in the low-shot regime. Our method is more related to the metric-based methods [30, 43, 63, 66, 67, 73], which aim to represent samples in an appropriate feature space where data from different categories can be distinguished with distance metrics.

Apart from the two popular branches, many other promising methods are also proposed to handle the few-shot classification problem, such as works based on graph theories [13, 15, 21], reinforcement learning [6], differentiable

SVM [25], temporal convolutions [39], *etc*. [4, 12, 14, 20, 27, 28, 37, 45, 49, 57, 60, 61, 65, 69–71, 76, 77].

Other related topics. Besides image classification, few-shot learning is also investigated in image segmentation [34, 74, 75] and object detection tasks [72]. There are also some previous works related to the techniques adopted in this paper. For example, Schulter *et al*. [56] solve the multi-object tracking problem with a network flow formulation. Zhao *et al*. [78] propose to use the differential EMD to handle visual tracking problem based on the sensitivity analysis of the simplex method. Li [29] uses a tensor-SIFT based EMD to tackle the contour tracking problem.

3. Method

In this section, we first present a brief review of the Earth Mover's Distance and describe how we formulate the one-shot classification as an optimal matching problem that can be trained end-to-end. Then, we describe our cross-reference mechanism to generate the weight of each node, which is an important parameter in the EMD formulation. Finally, we demonstrate how to use the EMD distance function to handle k-shot learning with our proposed structured fully connected layer. The overview of our framework for one-shot classification is shown in Fig. 2.

3.1. Revisiting Earth Mover's Distance

The Earth Mover's Distance is a distance measure between two sets of weighted objects or distributions, which is built upon the basic distance between individual objects. It has the form of the well-studied transportation problem (TP) from Linear Programming. Specially, suppose that a set of sources or suppliers $\mathcal{S} = \{s_i | i = 1, 2, ...m\}$ are required to transport goods to a set of destinations or demanders $\mathcal{D} = \{d_j | j = 1, 2, ...k\}$, where s_i denotes the supply units of supplier i and d_j represents the demand of j-th demander. The cost per unit transported from supplier i to demander j is denoted by c_{ij} , and the number of units transported is denoted by x_{ij}. The goal of the transportation problem is then to find a least-expensive flow of goods $\tilde{\mathcal{X}} = \{\tilde{x}_{ij} | i = 1, ...m, j = 1, ...k\}$ from the suppliers to the demanders:

$$\begin{aligned} \underset{x_{ij}}{\text{minimize}} \quad & \sum_{i=1}^{m} \sum_{j=1}^{k} c_{ij} x_{ij} \\ \text{subject to} \quad & x_{ij} \geqslant 0, i = 1, ..., m, j = 1, ..., k \\ & \sum_{j=1}^{k} x_{ij} = s_i, \;\; i = 1, ...m \\ & \sum_{i=1}^{m} x_{ij} = d_j, \;\; j = 1, ...k \end{aligned} \tag{1}$$

Note that roles of suppliers and demanders can be switched without affecting the total transportation cost. s_i and d_j are also called the weights of the nodes, which controls the total matching flows generated by each node. EMD seeks an optimal matching $\tilde{\mathcal{X}}$ between suppliers and demanders such that overall matching cost can be minimized. The global optimal matching flows $\tilde{\mathcal{X}}$ be achieved by solving the Linear Programming problem.

3.2. EMD for Few-Shot Classification

In the few-shot classification task, metric-based methods aim to find a good distance metric and data representations to compute the similarity between images, which are used for classification. Different from the previous methods that perform distance computation between the image-level embeddings, our approach advocates the use of discriminative local information. We decompose an image into a set of local representations and use the optimal matching cost between two images to represent their similarity. Concretely, we first deploy a fully convolutional network (FCN) [59] to generate the image embedding $\mathbf{U} \in \mathbb{R}^{H \times W \times C}$, where H and W denote the spatial size of the feature map and C is the feature dimension. Each image representation contains a collection of local feature vectors $[\mathbf{u}_1, \mathbf{u}_2, ...\mathbf{u}_{HW}]$, and each vector $\mathbf{u}_i$ can be seen as a node in the set. Thus, the similarity of two images can be represented as the optimal matching cost between two sets of vectors. Following the original EMD formulation in Equation 1, the cost per unit is obtained by computing the pairwise distance between embedding nodes $\mathbf{u}_i$, $\mathbf{v}_j$ from two image features:

$$c_{ij} = 1 - \frac{\mathbf{u}_i{}^T \mathbf{v}_j}{\|\mathbf{u}_i\| \|\mathbf{v}_j\|}, \tag{2}$$

where nodes with similar representations tend to generate fewer matching cost between each other. As to the generation of weights s_i and d_j, we leave the detailed elaborations in Section 3.4. Once acquiring the optimal matching flows $\tilde{\mathcal{X}}$, we can compute the similarity score s between image representations with:

$$s(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^{HW} \sum_{j=1}^{HW} (1 - c_{ij}) \tilde{x}_{ij}. \tag{3}$$

3.3. End-to-End Training

In order to embed the optimal matching problem into a neural network for end-to-end training, it is helpful to make the solution of the optimal matching $\tilde{\mathcal{X}}$ be differentiable with respect to the problem parameters θ. As is indicated by [3], we can apply implicit function theorem [3, 8, 22] on the optimality (KKT) conditions to obtain the Jacobian. For the sake of completeness, we start with Equation 1 which is assigned compactly in matrix form:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & c(\theta)^T x \\ \text{subject to} \quad & G(\theta) x \leqslant h(\theta), \\ & A(\theta) x = b(\theta). \end{aligned} \tag{4}$$

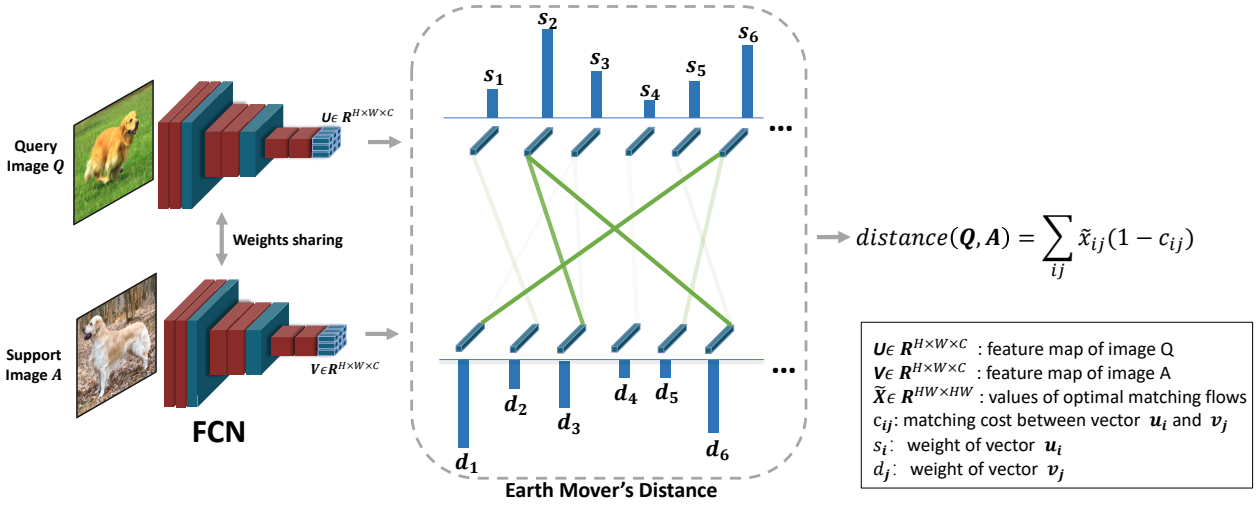

Figure 2: Our framework for 1-shot image classification. Given a pair of images, we first use a Fully Convolutional Network to generate dense representations of them, which contain two sets of feature vectors. The model generates the weights of all vectors with our proposed cross-reference mechanism (not indicated in the figure). Then we use the Earth Mover's Distance to generate the optimal matching flows between two sets that have the minimum overall matching cost. Finally, based on the optimal matching flows and matching costs, we can compute the distance between two images, which are used for classification.

Here $x \in \mathbb{R}^n$ is our optimization variable, with $n = HW \times HW$ representing the total number of matching flows in $\mathcal{X}$. θ is the problem parameter that relates to the earlier layers in a differentiable way. $Ax = b$ represents the equality constraints and $Gx \leqslant h$ denotes the inequality constraint in Equation 1. Accordingly, the Lagrangian of the LP problem in equation 4 is given by:

$$L(\theta, x, \nu, \lambda) = c^T x + \lambda^T (Gx - h) + \nu^T (Ax - b), \quad (5)$$

where ν are the dual variables on the equality constraints and $\lambda \geqslant 0$ are the dual variables on the inequality constraints.

Following the KKT conditions with notational convenience, we can obtain the optimum $(\tilde{x}, \tilde{\nu}, \tilde{\lambda})$ of the objective function by solving $g(\theta, \tilde{x}, \tilde{\nu}, \tilde{\lambda}) = 0$ with primal-dual interior point methods, where

$$g(\theta, x, \nu, \lambda) = \begin{bmatrix} \nabla_\theta L(\theta, x, \nu, \lambda) \\ \mathbf{diag}(\lambda)(G(\theta)x - h(\theta)) \\ A(\theta)x - b(\theta) \end{bmatrix}. \quad (6)$$

Then, the following theorem holds to help us derive the gradients of the LP parameters.

Theorem 1 (From Barratt [3]) Suppose $g(\theta, \tilde{\lambda}, \tilde{\nu}, \tilde{x}) = 0$. Then, when all derivatives exist, the partial Jacobian of x with respect to θ at the optimal solution $(\tilde{\lambda}, \tilde{\nu}, \tilde{x})$, namely $J_\theta \tilde{x}$, can be obtained by satisfying:

$$J_\theta \tilde{x} = -J_x g(\theta, \tilde{\lambda}, \tilde{\nu}, \tilde{x})^{-1} J_\theta g(\theta, \tilde{x}, \tilde{\nu}, \tilde{\lambda}). \quad (7)$$

Here the formula for the Jacobian of the solution mapping is obtained by applying the implicit function theorem to the KKT conditions. For instance, the (partial) Jacobian with respect to θ can be defined as

$$J_\theta g(\theta, \tilde{\lambda}, \tilde{\nu}, \tilde{x}) = \begin{bmatrix} J_\theta \nabla_x L(\theta, \tilde{x}, \tilde{\nu}, \tilde{\lambda}) \\ \mathbf{diag}(\tilde{\lambda}) J_\theta (G(\theta)x - h(\theta)) \\ J_\theta (A(\theta)\tilde{x} - b(\theta)) \end{bmatrix}. \quad (8)$$

Therefore, once getting the optimal solution $\tilde{x}$ for the LP problem, we can obtain a closed-form expression for the gradient of $\tilde{x}$ with respect to the input LP parameters θ. This helps us achieve an efficient backpropagation through the entire optimization process without perturbation of the initialization and optimization trajectory.

3.4. Weight Generation

As can be observed in the EMD formulation, an important problem parameter is the weight of each node, *e.g.*, s_i, which controls the total matching flows $\sum_{j=1}^{n} x_{ij}$ from it. Intuitively, the node with a larger weight plays a more important role in the comparison of two sets, while a node with a very small weight can hardly influence the overall distance no matter which nodes it matches with. In the pioneering work that adopts EMD for color-based image retrieval [50], they use the histogram as the elementary feature and perform feature clustering over all pixels to generate the nodes. The weight of each node is set as the size of the corresponding cluster. It makes sense because for color-based image retrieval, large weight should be given to the dominant colors with more pixels, such that the retrieved images can be visually close to the query images.

However, for few-shot image classification tasks where features for classification often have high-level semantic meanings, the number of pixels does not necessarily reflect the importance. It is common to find image data with

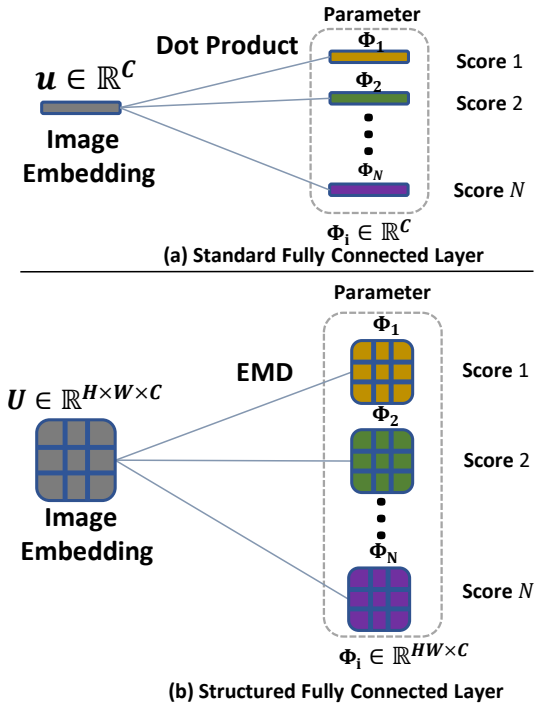

Figure 3: Comparison of standard fully connected layer **(a)** and our proposed structured fully connected layer (SFC) **(b)**. The SFC learns a group of vectors as the prototype for each class such that we can use the EMD to generate category scores.

greater background regions than the target objects in classification datasets, *e.g.*, ImageNet. Thus, the importance of a local feature representation can hardly be determined only by inspecting individual images alone. Instead, we argue that for the few-shot classification task, the weights of node features should be generated by comparing the nodes on both sides. To achieve this goal, we propose a cross-reference mechanism that uses dot product between a node feature and the average node feature in the other structure to generate a relevance score as the weight value:

$$s_i = \max\{\mathbf{u}_i^T \cdot \frac{\sum_{j=1}^{HW} \mathbf{v}_j}{HW}, 0\}, \tag{9}$$

where $\mathbf{u}_i$ and $\mathbf{v}_j$ denotes the vectors from two feature maps, and function $max(\cdot)$ ensures the weights are always non-negative. For clarity, here we simply take s_i as an example and d_i can be obtained in the same manner. The cross-reference mechanism aims to give less weight to the high-variance background regions and more to the co-occurrent object regions in two images. This can also put less weight on the object parts that do not occur in two images and thus allows partial matching to some extent. Finally, we normalize all the weights in the structure to make both sides have the same total weights for matching:

$$\hat{s}_i = s_i \frac{HW}{\sum_{j=1}^{HW} s_j}. \tag{10}$$

Model	Embedding	Metric	5-way	10-way
ProtoNet [63]	global	Euclidean	60.37	44.34
MatchingNet [67]	global	*cosine*	63.08	47.09
FC [5]	global	*dot*	59.41	44.08
FC [5]	global	*cosine*	55.43	40.42
KNN [30]	local	*cosine*	62.52	47.08
Prediction Fusion [33]	local	*cosine*	62.38	47.04
DeepEMD (our)	local	EMD	**65.91**	**49.66**

Table 1: Comparison of different metric-based methods for 1-shot classification. Our model with EMD as the distance metric significantly outperforms baseline models based on image-level representations and local representations.

3.5. Structured Fully Connected Layer

Thus far we have discussed using the Earth Mover's Distance as the metric to generate the distance value between paired images. A question is then raised—how do we handle the k-shot setting where multiple support images are available? Before presenting our design in detail, let us have a review of how the standard fully layer classify an image embedding extracted by CNNs. A FC layer, parameterized by $[\mathbf{\Phi}_1, ..., \mathbf{\Phi}_N] \in \mathbb{R}^{C \times N}$ contains a set of learnable vectors $\mathbf{\Phi}_i \in \mathbb{R}^C$ corresponding to each category. At inference time, given an image embedding $\mathbf{u} \in \mathbb{R}^C$ generated by the convolutional layer, the FC layer generates the score of a class i by computing the dot product between the image vector $\mathbf{u}$ and the parameter vector $\mathbf{\Phi}_i$, and this process is applied to all the categories in parallel by matrix multiplication. There are also some previous works replacing the dot product operation in the FC layer with the *cosine* function for computing the category scores [5, 53]. The learning of the FC layer can be seen as finding a prototype vector for each class such that we can use distance metrics to classify an image. An illustration of the standard FC layer is shown in Fig. 3 (a).

With the same formulation, we can learn a structured fully connected layer that adopts EMD as the distance function to directly classify a structured feature representation. The learnable embedding for each class becomes a group of vectors, rather than one vector, such that we can use the structured distance function EMD to undertake image classification. This can also be interpreted as learning a prototype feature map generated by a dummy image for each class. The comparison of the structured FC and the standard FC can be found in Fig. 3. At inference time, we fix the trained 1-shot FCN model as the feature extractor and use SGD to learn the parameters in the structured fully connected layer by sampling data from the support set.

4. Experiments

To evaluate the performance of our proposed algorithm for few-shot classification, we conduct extensive experiments on multiple datasets. In this section, we first present

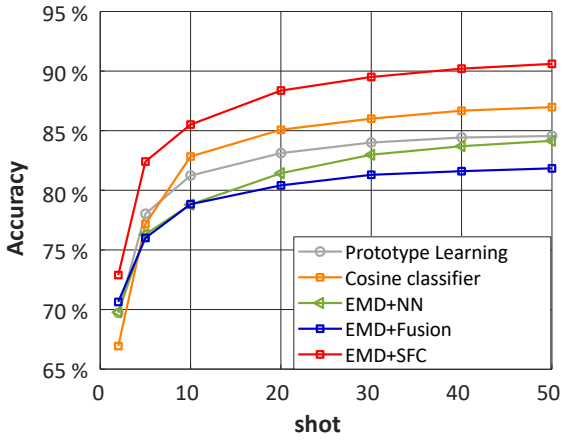

Figure 4: Experiment on 5-way k-shot classification. The proposed structured FC layer significantly outperforms previous k-shot solutions.

Method	Operation	5-way	10-way
Full Connections	Average	55.16	40.88
Full Connections	**CR**	55.41	41.60
EMD	Equal	56.95	42.89
EMD	K-means [19]	56.25	41.85
EMD	mean-shift [7]	53.56	39.70
EMD	**CR**	**61.13**	**46.92**

Table 2: Different methods for setting the weights in the EMD. We report the 1-shot performance with only the feature pre-training step. EMD with our cross-reference (**CR**) mechanism yields the best result. The model variant that is solely based the on the cross-reference mechanism as attention and removes the EMD can cause a significant performance drop. The combination of the EMD and the cross-reference generates the best result.

dataset information and some important implementation details in our network design. Then, we conduct various ablative experiments to validate each component in our network. Finally, we compare our model with the state-of-the-art methods on popular benchmark datasets.

4.1. Implementation Details

For a fair comparison with previous works, we employ a 10-layer ResNet (ResNet10) as our model backbone, which is widely used in the few-shot classification literature. We transform it into a fully convolutional manner by removing the classifier at the end. Given an image of size 84×84, the model generates a feature map of size $5 \times 5 \times 512$, *i.e.* 25 512-dimensional vectors. We adopt the GPU accelerated Convex solver QPTH [1] to solve the Linear Programming problem in our network and compute gradients for back-propagation. As is commonly implemented in the state-of-the-art literature, we adopt a feature pre-training step followed by the episodic meta-training [67] to learn our network. During the network pre-training, we find that a two-layer FC layer is better than a single layer classifier. For the k-shot classification task, we initialize the structured FC layer with the average feature map of all support data in each class, and sample a batch of 5 images from the support set to finetune the structured FC layer for 100 iterations.

4.2. Dataset Description

We conduct few-shot classification experiments on four popular benchmark datasets: *mini*ImageNet [67], *tiered*ImageNet [49], Fewshot-CIFAR100 (FC100) [43] and Caltech-UCSD Birds-200-2011 (CUB) [68].

***mini*ImageNet.** *mini*ImageNet was first proposed in [67], and becomes the most popular benchmark in the few-shot classification literature. It contains 100 classes with 600 images in each class, which are built upon the ImageNet dataset [51]. The 100 classes are divided into 64, 16, 20 for meta-training, meta-validation and meta-testing, respectively.

***tiered*ImageNet.** *tiered*ImageNet is also a subset of ImageNet, which includes 608 classes from 34 super-classes. Compared with *mini*ImageNet, the splits of meta-training(20), meta-validation(6) and meta-testing(8) are set according to the super-classes to enlarge the domain difference between training and testing phase. The dataset also include more images for training and evaluation (779,165 images in total).

Fewshot-CIFAR100. FC100 is a few-shot classification dataset build on CIFAR100 [23]. We follow the split division proposed in [43], where 36 super-classes were divided into 12 (including 60 classes), 4 (including 20 classes), 4 (including 20 classes), for meta-training, meta-validation and meta-testing, respectively, and each class contains 100 images.

Caltech-UCSD Birds-200-2011. CUB was originally proposed for fine-grained bird classification, which contains 11,788 images from 200 classes. We follow the splits in [73] that 200 classes are divided into 100, 50 and 50 for meta-training, meta-validation and meta-testing, respectively. The challenge in this dataset is the minor difference between classes.

4.3. Ablative Analysis

In our ablation study, we implement various experiments to evaluate the effectiveness of our algorithm. All experiments are conducted on the *mini*ImageNet dataset.

Comparison with methods based on image-level representations. In the beginning, we first compare our method with a set of metric based methods that utilize image-level vector representations on the 1-shot task. These methods adopt global average pooling to generate vector representations for images and use various distance metrics for classification. We select the representative metric-based methods in the literature for comparison: 1) Prototypical Network [63] with Euclidean distance. 2) Matching Network [67] with *cosine* distance. 3) Finetuning a FC clas-

Method	Backbone	*mini*Imagenet		*tiered*Imagenet	
		1-shot	5-shot	1-shot	5-shot
***cosine* classifier [5]**	*ResNet12*	55.43 ± 0.81	77.18 ± 0.61	61.49 ± 0.91	82.37. ± 0.67
TADAM [43]	*ResNet12*	58.50 ± 0.30	76.70 ± 0.30	-	-
ECM [48]	*ResNet12*	59.00 ± –	77.46 ± –	63.99 ± –	81.97 ± –
TPN [35]	*ResNet12*	59.46 ± –	75.65 ± –	59.91 ± 0.94	73.30 ± 0.75
PPA [46]	*WRN-28-10*†	59.60 ± 0.41	73.74 ± 0.19	65.65 ± 0.92	83.40. ± 0.65
ProtoNet [63]	*ResNet12*	60.37 ± 0.83	78.02 ± 0.57	65.65 ± 0.92	83.40. ± 0.65
wDAE-GNN [15]	*WRN-28-10*†	61.07 ± 0.15	76.75 ± 0.11	68.18 ± 0.16	83.09 ± 0.12
MTL [64]	*ResNet12*	61.20 ± 1.80	75.50 ± 0.80	-	-
LEO [52]	*WRN-28-10*†	61.76 ± 0.08	77.59 ± 0.12	66.33 ± 0.05	81.44 ± 0.09
DC [33]	*ResNet12*	62.53 ± 0.19	79.77 ± 0.19	-	-
MetaOptNet [25]	*ResNet12*	62.64 ± 0.82	78.63 ± 0.46	65.99 ± 0.72	81.56 ± 0.53
FEAT [73]	*ResNet24*†	62.96 ± 0.20	78.49 ± 0.15	-	-
MatchNet [67]	*ResNet12*	63.08 ± 0.80	75.99 ± 0.60	68.50 ± 0.92	80.60 ± 0.71
CTM [28]	*ResNet18*†	64.12 ± 0.82	80.51 ± 0.13	68.41 ± 0.39	84.28 ± 1.73
DeepEMD (our)	*ResNet12*	**65.91 ± 0.82**	**82.41 ± 0.56**	**71.16 ± 0.87**	**86.03 ± 0.58**

(a) Results on ***mini*ImageNet** and ***tiered*ImageNet** datasets

Method	Backbone	1-shot	5-shot
***cosine* classifier [5]**	*ResNet12*	38.47 ± 0.70	57.67 ± 0.77
TADAM [43]	*ResNet12*	40.10 ± 0.40	56.10 ± 0.40
MetaOptNet [25]	*ResNet12*	41.10± 0.60	55.5 ± 0.60
ProtoNet [63]	*ResNet12*	41.54 ± 0.76	57.08 ± 0.76
DC [33]	*ResNet12*	42.04 ± 0.17	57.05 ± 0.16
MatchNet [67]	*ResNet12*	43.88 ± 0.75	57.05 ± 0.71
MTL [64]	*ResNet12*	45.10± 1.8	57.6 ± 0.9
DeepEMD (our)	*ResNet12*	**46.47 ± 0.78**	**63.22 ± 0.71**

(b) Results on **Fewshot-CIFAR100** dataset.

Method	Backbone	1-shot	5-shot
ProtoNet [63]	*ResNet12*	66.09 ± 0.92	82.50 ± 0.58
RelationNet [5, 66]	*ResNet34*†	66.20 ± 0.99	82.30 ± 0.58
DEML [79]	*ResNet50*†	66.95 ± 1.06	77.11 ± 0.78
MAML [5, 9]	*ResNet34*†	67.28 ± 1.08	83.47 ± 0.59
***cosine* classifier [5]**	*ResNet12*	67.30 ± 0.86	84.75 ± 0.60
MatchNet [67]	*ResNet12*	71.87 ± 0.85	85.08 ± 0.57
DeepEMD (our)	*ResNet12*	**75.65 ± 0.83**	**88.69 ± 0.50**

(c) Results on **Caltech-UCSD Birds-200-2011** dataset.

Table 3: Comparison with the state-of-art 1-shot 5-way and 5-shot 5-way performance (%) with 95% confidence intervals on ***mini*ImageNet (a)**, ***tiered*ImageNet (a)**, **Fewshot-CIFAR100 (b)** and **Caltech-UCSD Birds-200-2011 (c)** datasets. Our model achieves new state-of-the-art performance on all datasets, and even outperforms methods with deeper backbones†.

sifier. In [5], Chen *et al*. propose to fix the pretrained feature extractor and finetune the FC layer with the support images. For fair comparisons, we adopt the same backbones and training schemes for all these baseline methods and report the experiment results in Table. 1. As we can see, our algorithm significantly outperforms baseline methods that relies on image-level vector representations under both 1-shot 5-way and 1-shot 10-way settings, which indicates the effectiveness of the optimal matching based method that relies on local features.

Comparison with methods based on local representations. There are also a few methods in the literature focusing on local representations to solve few-shot classification. They all remove the global average pooling in the CNN to achieve dense representations of images. In [30], Li *et al*. use the top k nearest vectors (**KNN**) between two feature maps to represent image-level distance. Lifchitz *et al*. [33] propose to make predictions with each local representation and average their output probabilities. We replace our EMD head with their methods for comparison. The result is shown in Table. 1. Our optimal matching based algorithm outperforms all other model variants. The possible reason is that although the basic ground distance in the EMD is based on local features, our algorithm compares the two structures in a global way. Solely based on nearest local features in two images may not extract sufficient information to differentiate images. For example, eyes can be the nearest feature between animal images, but such feature can hardly be used to differentiate animal species.

Weights in the EMD. We next investigate the weights in the EMD formulations. In the earlier work using the EMD for image retrieval, they use the pixel color as the feature and cluster pixels to generate nodes. The weight of the node is set as the portion of pixels in this cluster. We experiment two clustering algorithms to generate weights as the baseline models: K-means [19] and mean-shift [7]. As the clustering process of the mentioned algorithms are non-differentiable, for a fair comparison, we use the features after pre-training to evaluate all methods. We also incorporate a baseline model with equal weights into comparison. To test whether our performance is solely brought by the cross-reference mechanism, we also compare our network with a model variant that is solely based on the cross-reference mechanism without EMD. We compute the *cosine* distance between all vector pairs, and compute a weighted sum of these distances with the node weights generated by the cross-reference mechanism. As we can see from the results in Table. 2, our cross-reference mechanism can bring an improvement of 4.2% over the baseline with equal weights, while clustering-based methods do not help improve the performance, which demonstrates that the number of pixels dose not necessarily indicate the importance in the few-shot scenario. For the model variant solely based on the cross-reference mechanism as an attention, it can only slightly improve the result of simple average operation, while a combination of both the cross-reference mechanism and the EMD can yield a significant performance improvement, which again validates the advantages of using the EMD as the metric and the effectiveness of the cross-reference mechanism.

Comparison with other *k*-shot methods. As the EMD distance metric is a paired function for two structures, the first baseline model for k-shot experiment is the nearest neighbour (**NN**) method that classifies the query images as the category of the nearest support sample. We also test making prediction with each support sample and fuse their logits. We then compare our network with a few k-shot solutions in previous works: 1) Prototype Learning. In [63],

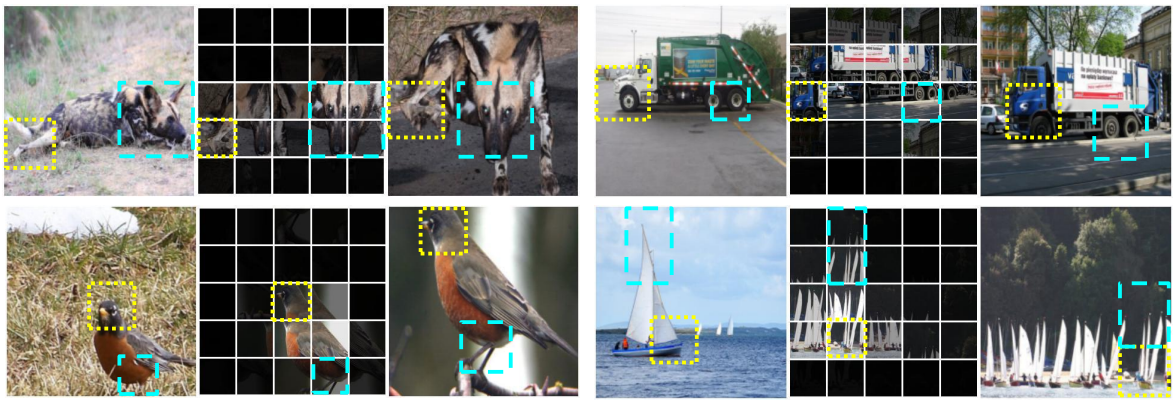

Figure 5: Visualization of the optimal matching flows. Given two images (left and right), we plot the best matched patch of each local region in the left image (middle). The weight controls the brightness of the corresponding region. The middle image can also be seen as the reconstruction of the left image using patches from the right one. Our algorithm can effectively establish semantic correspondence between local regions and gives less weights to the background regions.

they average the feature embeddings of support images in each class as the prototype and apply the nearest neighbour method for classification. 2) Finetuning a *cosine* classifier [5]. We test the model performance on the k-shot 5-way tasks under multiple k values , and the results are shown in Fig. 4. Our structured FC layer consistently outperforms baseline models and with the number of support sets increasing, our network shows even more advantages.

Visualization of matching flows and weights. It is interesting to visualize the optimal matching flows and node weights in the network inference process. In Fig. 5, we provide some visualization examples. The middle image plots the best matched patch of each local region in the left image, and the weight controls the brightness of the corresponding region. The middle image can also be seen as a reconstructed version of the left image, using the local patches from the right image. As we can see, our algorithm can establish semantic correspondence between local regions, and the background regions in the left image are given small weight, thus contributing less to the overall distance. The visualization of full matching flows and more examples can be found in our supplementary material.

4.4. Time Complexity.

Compared with the baseline models, the training and inference of DeepEMD come with more computation cost, as an LP problem must be solved for each forward process. As is discussed in [1], the main computation lies in the factorization of the KKT matrix as well as back-substitution when using the interior point method to solve the LP problem, which have cubic and quadratic time complexity, respectively. We also tried the OpenCV library for solving the LP problem via a modified simplex algorithm, which is much faster than the QPTH [1] solver which uses the interior point method. Therefore, we can use QPTH for training the network and use OpenCV for validation and the final test.

4.5. Comparison with the State-of-the-Art

Finally, we compare our algorithm with the state-of-the-art methods. We report 1-shot 5way and 5-shot 5-way performance on 4 popular benchmarks: *mini*ImageNet, *tiered*ImageNet, FC100 and CUB. We reproduce the models in some earlier works [5, 63, 67] with our network backbone and training strategy, and report the highest performance between our results and their reported ones. The results are shown in Table. 3. Our algorithm achieves new state-of-the-art performance on all datasets. In particular, our results outperform the state-of-the-art performance by a significant margin on multiple tasks, *e.g.*, 1-shot (**%3.78**) and 5-shot (**3.61%**) on the CUB dataset; 5-shot (**%5.55**) on the FC100 dataset.

5. Conclusion

We have proposed a few-shot classification framework that employs the Earth Mover's Distance as the distance metric. The implicit theorem allows our network end-to-end trainable. Our proposed cross-reference mechanism for setting the weights of nodes turns out crucial in the EMD formulation and can effectively minimize the negative impact caused by irrelevant regions. The learnable structured fully connected layer can directly classify dense representations of images in the k-shot settings. Our algorithm achieves new state-of-the-art performance on multiple dataset.

Acknowledgements

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme (Award Number: AISG-RP-2018-003) and the MOE Tier-1 research grants: RG126/17 (S) and RG28/18 (S).

References

[1] Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 136–145. PMLR, 2017. 6, 8

[2] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your maml. In *ICLR*, 2019. 2

[3] Shane Barratt. On the differentiability of the solution to convex optimization problems. *arXiv preprint arXiv:1804.05098*, 2018. 2, 3, 4

[4] Sergey Bartunov and Dmitry P. Vetrov. Few-shot generative modelling with generative matching networks. In *AISTATS*, 2018. 3

[5] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019. 2, 5, 7, 8

[6] Wen-Hsuan Chu, Yu-Jhe Li, Jing-Cheng Chang, and Yu-Chiang Frank Wang. Spot and learn: A maximum-entropy patch sampler for few-shot image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[7] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, May 2002. 6, 7

[8] Asen L Dontchev and R Tyrrell Rockafellar. Implicit functions and solution mappings. *Springer Monographs in Mathematics. Springer*, 208, 2009. 2, 3

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

[10] Sebastian Flennerhag, Andrei A Rusu, Razvan Pascanu, Hujun Yin, and Raia Hadsell. Meta-learning with warped gradient descent. *arXiv preprint arXiv:1909.00025*, 2019. 2

[11] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimilano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. *arXiv preprint arXiv:1806.04910*, 2018. 2

[12] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *ICML*, 2018. 3

[13] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017. 2

[14] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 3

[15] 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. 2, 7

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[17] Frank L Hitchcock. The distribution of a product from several sources to numerous localities. *Journal of mathematics and physics*, 20(1-4):224–230, 1941. 2

[18] Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[19] Xin Jin and Jiawei Han. *K-Means Clustering*, pages 563–564. Springer US, Boston, MA, 2010. 6, 7

[20] Rohit Keshari, Mayank Vatsa, Richa Singh, and Afzel Noore. Learning structure and strength of CNN filters for small sample size training. In *CVPR*, 2018. 3

[21] 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. 2

[22] Steven G Krantz and Harold R Parks. *The implicit function theorem: history, theory, and applications*. Springer Science & Business Media, 2012. 2, 3

[23] Alex Krizhevsky. Learning multiple layers of features from tiny images. *University of Toronto*, 2009. 6

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 1

[25] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, 2019. 3, 7

[26] Yoonho Lee and Seungjin Choi. Gradient-based meta-learning with learned layerwise metric and subspace. In *ICML*, 2018. 2

[27] Fei-Fei Li, Robert Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):594–611, 2006. 3

[28] Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. Finding task-relevant features for few-shot learning by category traversal. In *CVPR*, 2019. 3, 7

[29] Peihua Li. Tensor-sift based earth mover's distance for contour tracking. *Journal of mathematical imaging and vision*, 46(1):44–65, 2013. 3

[30] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 5, 7

[31] Xinzhe Li, Qianru Sun, Yaoyao Liu, Qin Zhou, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. In *Advances in Neural Information Processing Systems*, pages 10276–10286, 2019. 2

[32] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few shot learning. In *ICML*, 2018. 2

[33] Yann Lifchitz, Yannis Avrithis, Sylvaine Picard, and Andrei Bursuc. Dense classification and implanting for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2, 5, 7

[34] Weide Liu, Chi Zhang, Guosheng Lin, and Fayao Liu. Crnet: Cross-reference networks for few-shot segmentation, 2020. 3

[35] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. *arXiv preprint arXiv:1805.10002*, 2018. 7

[36] Jelena Luketina, Tapani Raiko, Mathias Berglund, and Klaus Greff. Scalable gradient-based tuning of continuous regularization hyperparameters. In *ICML*, 2016. 2

[37] Akshay Mehrotra and Ambedkar Dukkipati. Generative adversarial residual pairwise networks for one shot learning. *arXiv*, 1703.08033, 2017. 3

[38] Luke Metz, Niru Maheswaranathan, Brian Cheung, and Jascha Sohl-Dickstein. Meta-learning update rules for unsupervised representation learning. In *ICLR*, 2019. 2

[39] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. Snail: A simple neural attentive meta-learner. In *ICLR*, 2018. 3

[40] Tsendsuren Munkhdalai and Hong Yu. Meta networks. In *ICML*, 2017. 2

[41] Tsendsuren Munkhdalai, Xingdi Yuan, Soroush Mehri, and Adam Trischler. Rapid adaptation with conditionally shifted neurons. In *ICML*, 2018. 2

[42] Devang K Naik and RJ Mammone. Meta-neural networks that learn by learning. In *IJCNN*, 1992. 2

[43] Boris N. Oreshkin, Pau Rodríguez, and Alexandre Lacoste. TADAM: task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, 2018. 1, 2, 6, 7

[44] Eunbyung Park and Junier B Oliva. Meta-curvature. In *Advances in Neural Information Processing Systems*, pages 3309–3319, 2019. 2

[45] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5822–5830, 2018. 3

[46] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L. Yuille. Few-shot image recognition by predicting parameters from activations. In *CVPR*, 2018. 2, 7

[47] Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with implicit gradients. In *Advances in Neural Information Processing Systems*, pages 113–124, 2019. 2

[48] Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Few-shot learning with embedded class models and shot-free meta training. *arXiv preprint arXiv:1905.04398*, 2019. 7

[49] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018. 3, 6

[50] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000. 2, 4

[51] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 6

[52] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *ICLR*, 2019. 7

[53] Tim Salimans and Durk P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 901–909, 2016. 5

[54] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap. Meta-learning with memory-augmented neural networks. In *ICML*, 2016. 2

[55] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero- and few-shot learning via aligned variational autoencoders. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[56] Samuel Schulter, Paul Vernaza, Wongun Choi, and Manmohan Chandraker. Deep network flow for multi-object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6951–6960, 2017. 3

[57] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Rogério Schmidt Feris, Abhishek Kumar, Raja Giryes, and Alexander M. Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. In *NeurIPS*, 2018. 3

[58] Tyler R. Scott, Karl Ridgeway, and Michael C. Mozer. Adapted deep embeddings: A synthesis of methods for k-shot inductive transfer learning. In *NeurIPS*, 2018. 2

[59] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017. 3

[60] Wei Shen, Ziqiang Shi, and Jun Sun. Learning from adversarial features for few-shot classification. *arXiv preprint arXiv:1903.10225*, 2019. 3

[61] Pranav Shyam, Shubham Gupta, and Ambedkar Dukkipati. Attentive recurrent comparators. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3173–3181. JMLR. org, 2017. 3

[62] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[63] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017. 1, 2, 5, 6, 7, 8

[64] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *CVPR*, 2019. 2, 7

[65] Xin Sun, Zhenning Yang, Chi Zhang, Guohao Peng, and Keck-Voon Ling. Conditional gaussian distribution learning for open set recognition. *arXiv preprint arXiv:2003.08823*, 2020. 3

[66] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018. 1, 2, 7

[67] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NIPS*, 2016. 1, 2, 5, 6, 7, 8

[68] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 6

[69] Yu-Xiong Wang, Ross B. Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *CVPR*, 2018. 3

[70] Davis Wertheimer and Bharath Hariharan. Few-shot learning with localization in realistic settings. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[71] Shipeng Yan, Songyang Zhang, and Xuming He. A dual attention network with semantic embedding for few-shot learning. In *AAAI*, 2019. 3

[72] Ze Yang, Yali Wang, Xianyu Chen, Jianzhuang Liu, and Yu Qiao. Context-transformer: Tackling object confusion for few-shot detection, 2020. 3

[73] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Learning embedding adaptation for few-shot learning. *arXiv*, 1812.03664, 2018. 1, 2, 6, 7

[74] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9587–9595, 2019. 3

[75] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5217–5226, 2019. 3

[76] 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. 3

[77] Ruixiang Zhang, Tong Che, Zoubin Grahahramani, Yoshua Bengio, and Yangqiu Song. Metagan: An adversarial approach to few-shot learning. In *NeurIPS*, 2018. 3

[78] Qi Zhao, Zhi Yang, and Hai Tao. Differential earth mover's distance with its applications to visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):274–287, 2008. 3

[79] Fengwei Zhou, Bin Wu, and Zhenguo Li. Deep meta-learning: Learning to learn in the concept space. *arXiv*, 1802.03596, 2018. 2, 7