

Transductive Few-Shot Classification on the Oblique Manifold

Guodong Qi^{1,2}, Huimin Yu^{1,2,3}, Zhaohui Lu², Shuzhao Li^{1,2}

¹College of Information Science and Electronic Engineering, Zhejiang University

²ZJU-League Research & Development Center, ³State Key Lab of CAD&CG, Zhejiang University

{guodong-qi, yhm2005, leeshuz}@zju.edu.cn, lzh8910210@163.com

Abstract

Few-shot learning (FSL) attempts to learn with limited data. In this work, we perform the feature extraction in the Euclidean space and the geodesic distance metric on the Oblique Manifold (OM). Specially, for better feature extraction, we propose a non-parametric Region Self-attention with Spatial Pyramid Pooling (RSSPP), which realizes a trade-off between the generalization and the discriminative ability of the single image feature. Then, we embed the feature to OM as a point. Furthermore, we design an Oblique Distance-based Classifier (ODC) that achieves classification in the tangent spaces which better approximate OM locally by learnable tangency points. Finally, we introduce a new method for parameters initialization and a novel loss function in the transductive settings. Extensive experiments demonstrate the effectiveness of our algorithm and it outperforms state-of-the-art methods on the popular benchmarks: mini-ImageNet, tiered-ImageNet, and Caltech-UCSD Birds-200-2011 (CUB).

1. Introduction

Convolutional neural networks (CNN) trained with large-scale labeled data have achieved the competitive performance as humans in recent years. However, different from these models struggling with a few labeled instances per class, humans learn rapidly by leveraging content and prior knowledge. To address this, few-shot learning (FSL) has drawn increasing attention. FSL aims to learn a prior knowledge that can rapidly generalize to new tasks with limited samples.

The *transductive* learning methods [66, 38, 33, 4] and *metric* learning methods [48, 27, 62] have been promising in a recent line of works. Transductive learning has shown superior performance over *inductive* learning, because the unlabeled test examples are classified at once, instead of the one sample at a time as in inductive settings. Metric learning methods represent the image in an appropriate feature space and replace the fully connected layer in standard

image classification [20] with the distance function, *e.g.* Euclidean distance or *cosine* distance. However, these works may lose the geometric inherent properties in the Euclidean space. Though it can be alleviated by data dimensionality reduction [22, 50, 21], these methods are not generalized well to new tasks or easy to over-fit in the few-shot settings.

Another solution is to make use of Riemannian geometry [11]. Riemannian geometry studies real smooth differential manifolds and it defines several geometric notions, *e.g.* the length of a curve, with a Riemannian metric. Admitting the Riemannian geometry, the Grassmannian manifold [26, 55] and the SPD manifold [36] are highly prevalent in modeling characters of image sets and videos, where *intra-class* variance, *e.g.*, illumination conditions or other scenarios, are comprised. They are capable of “filling-in” missing images. However, the advantages are based on modeling sufficient number of images for each class. It is infeasible to apply the aforementioned manifolds to FSL.

Another manifold that admits Riemannian geometry, is the *oblique manifold* (OM) [52], which is an embedded sub-manifold with all normalized columns and it is used for independent component analysis [1]. We argue that OM is superior in FSL. The reasons are in two-fold: 1) *Whitening* is not required in OM. The whitening step is to estimate the covariance matrix and remove the scaling indeterminacy of data in Grassmannian manifold or SPD manifold, which is infeasible to perform whitening step for a single image classification. Without whitening, OM is free of extrinsic constraints. 2) All the columns of OM have unit Euclidean norm. OM offers an intrinsic property similar to the *L2-normalization* in Euclidean space. In this way, the CNN features in Euclidean space can be embedded to OM more easily.

However, the absence of whitening may lead to weak generalization of OM, since the whitening step is to remove the impact of intra-variance of data. To address this, we resort to the powerful CNN to enhance the generalization, as generalization emerged from the pretrained CNN [20, 64, 34], ensemble learning [9] or SPP [19]. However, more generalization means less discriminative

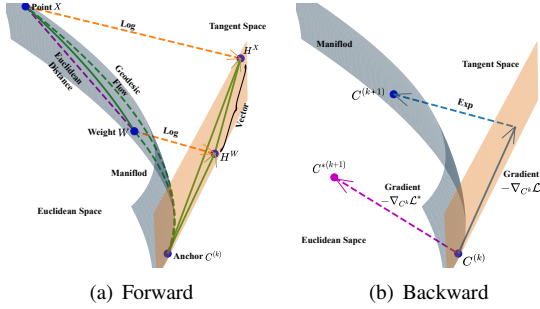


Figure 1: Illustration of Oblique Distance-based Classifier (ODC), which is parameterized with *Weights* and *Anchors*. (a) Forward: more precious than Euclidean distance, the geodesic flow is transformed to a vector in the tangent space by Log. (b) Backward: parameters in Euclidean space are updated in the directions of negative gradient, while manifold-valued parameters are updated along the manifold by Exp. Best viewed in color.

CNN [39]. Recent works [6, 59] show that discriminative local regions can be enriched when training with self-attention networks. Inspired by these works, to improve the generalization of OM without losing the discriminative representation, we propose a non-parametric *region self-attention with spatial pyramid pooling* (RSSPP). Specially, given a CNN feature map from a single image, RSSPP applies multi-kernel max-pooling similar to SPP [19] to enhance the generalization of features. Then, RSSPP employs the self-attention mechanism to improve the discriminative ability. Note that our RSSPP is non-parametric and avoids over-fitting in FSL.

After applying RSSPP, to take advantage of the aforementioned properties of OM, we map the Euclidean features to embedded features¹ on OM. Since distance-based classifier is widely used in FSL [4, 51], we present a novel *oblique distance-based classifier* (ODC) to classify the manifold-valued features. As illustrated in Figure 1, ODC is parameterized with *weights* and *anchors*. Both are members of OM, where the weights put emphasis on classifying points by the precise geodesic distance, and the anchors are points of tangency, which determine the tangent spaces. The tangent spaces offer Euclidean properties, so we can employ modern machine learning tools to perform classification with an iterative way. In the forward pass, the geodesic distance is transformed to the vector in tangent spaces at the anchors with the logarithmic map (Log). During the backward pass, the gradient is computed, mapped back to the geodesic flow by exponential map (Exp), and used to update the parameters without leaving manifold.

As the representation ability of tangent spaces decreases

¹We also refer to the embedded features as *manifold-valued features* or *points* to be distinguished from features in Euclidean.

a lot when points are too far from anchors [47], the anchors need to be initialized appropriately. We argue that the anchors should be in the neighborhood of the Karcher mean (KM) [17] among associated points. Considering the potential ill-distribution gap between the train samples and the test samples, we initialize the anchors by selecting samples in the transductive settings, and design a weighted loss function to integrate the tangent spaces. Furthermore, considering the calculation of KM is an NP-hard problem, we propose a pseudo-KM method to utilize the weighted mean (*i.e.* inner product) operation on the features in Euclidean space, and then embed the mean feature to OM. Empirically we observe the initialization of weights is also critically important. Similar to the anchors, we calculate prototypes [46] and embed the prototypes to OM as the initial value of weights.

Finally, we acquire the classification scores by performing weighted sum over softmax on the Euclidean distance in tangent space. The effectiveness of our method is demonstrated by extensive experiments on multiple datasets. To conclude, our main contributions are summarized as follows:

- To our best knowledge, it is the first time to model FSL on OM, which intrinsically satisfies the normality constraint and without the need for whitening.
- We propose a non-parametric RSSPP. It applies the multi-kernel max-pooling similar to SPP to enhance generalization and the self-attention mechanism to improve discriminative ability.
- To perform classification in the oblique manifold, we propose the ODC parameterized with weights and anchors. ODC is initialized with a carefully designed strategy. We also design a weighted sum loss function over the anchors, and utilize the exponential map to update the weights and anchors.
- The experiments on popular datasets demonstrate that our algorithm on FSL significantly outperforms the baseline and achieves new state-of-the-art performance on all of them.

2. Related Work

Few-Shot Learning. Metric learning approaches [48, 30, 27, 62, 63] and optimization-based approaches [12, 29, 37, 41, 28] are two common line algorithms in few-shot learning. Optimization-based methods adapt the model parameters to new tasks rapidly with the inner loop. Metric learning methods, which are more related to ours, target at learning a good embedding in appropriate spaces and distance function to metric the relationship of feature. These methods refer to the whole image [53, 46] or local

regions [63, 10] as embedding. However, they all compute distance in the Euclidean space. Instead, we utilize the geodesic distance in the oblique manifold.

Recently, fine-tuning methods [8, 51] have shown that a pretrained in base datasets also provide a good embedding for news tasks, and they only fine-tune the last classifier layer. Transductive learning [32, 40, 60, 66] is another technique that classifies unlabeled data at once, instead of the one-by-one sample in inductive methods. TIM [4] maximizes the mutual information between the unlabeled data and the label probability together with the supervised loss during fine-tuning. Based on Tim, we propose a novel weighted sum loss function.

Manifold Learning. Manifold learning, including dimensionality reduction [22, 50, 21] and Riemannian manifolds learning [18, 16, 15, 26, 35, 58] has attracted significant attention in the past few years. Popular manifolds, *e.g.* Grassmannian manifolds [5, 55] and SPD manifolds [25, 36], leverage modern Euclidean machine learning tools and maintain the structure of the manifolds meanwhile. Recently, Souza *et al.* [47] propose to transform the manifold points to tangent vectors at random tangency points by logarithmic map. However, they all focus on image sets and video data, while we solve the single image classification in the oblique manifold. Furthermore, different from Souza *et al.* [47], the tangency points in our method are defined carefully.

Other Related Work. Descriptors, *e.g.*, modern CNN [20], have been utilized to encode an image to capture as much generalization as possible in recent years. Self-attention [6, 13], *e.g.*, spatial attention [56] and channel attention [49], are applied in computer vision as a complement to the whole features [59, 23, 65] to enrich discriminative information. In this work, we employ SPP [19] and spatial attention to make a trade-off between generalization and discriminative ability. Different from them, no extra parameters in our method ensures to avoid over-fitting.

3. Preliminaries

Before presenting the methods, we give a brief overview of the oblique manifold geometry and the few-shot problem.

3.1. Oblique Manifold

The *oblique manifold* (OM) $\mathcal{O}(n, p)$ is defined as a set of matrix in $\mathbb{R}^{n \times p}$ with unit Euclidean norm columns [52, 1]. Formally, OM is defined by:

$$\mathcal{O}(n, p) = \{ \mathbf{Y} \in \mathbb{R}^{n \times p} : \text{diag}(\mathbf{Y}^T \mathbf{Y}) = \mathbf{I}_p \} \quad (1)$$

where $\text{diag}(\cdot)$ denotes the diagonal matrix.

3.2. Tangent Spaces

A tangent space $T_{\mathbf{K}}\mathcal{O}$ is the set of all tangent vectors to \mathcal{O} at point \mathbf{K} . It provides a local vector space approxima-

tion of the manifold. Any tangent vector $\dot{\mathbf{K}}$ must meet this constraint: $\dot{\mathbf{K}}^T \mathbf{K} + \mathbf{K}^T \dot{\mathbf{K}} = \mathbf{0}$. To be exact, the tangent space at \mathbf{K} on OM is calculated by:

$$T_{\mathbf{K}}\mathcal{O}(n, p) = \{ \dot{\mathbf{K}} \in \mathbb{R}^{n \times p} : \text{diag}(\mathbf{K}^T \dot{\mathbf{K}}) = \mathbf{0} \} \quad (2)$$

3.3. Exponential and Logarithmic Maps

Given a length t on a geodesic $\gamma(t)$, where $\gamma(0)$ is a start point, $\dot{\gamma}(t)$ is a direction, the specific points are obtained by the exponential map $\text{Exp} : \mathcal{O} \times T\mathcal{O} \times \mathbb{R} \rightarrow \mathcal{O}$. Specially, we denote $\gamma(t) = \text{Exp}_{\mathbf{K}} \mathbf{H}$, which shows point $\gamma(t)$ in geodesic emanating by:

$$\text{Exp}_{\mathbf{K}}(\mathbf{H}) = \mathbf{K} \cos(t\|\dot{\mathbf{K}}\|) + \frac{\dot{\mathbf{K}}}{\|\dot{\mathbf{K}}\|} \sin(t\|\dot{\mathbf{K}}\|) \quad (3)$$

$\|\cdot\|$ refers to the Frobenius norm in the ambient space, $\mathbf{H} = t\dot{\gamma}(0)$, $\gamma(0) = \mathbf{K}$ and $\dot{\gamma}(0) = \dot{\mathbf{K}}$

Logarithmic map is the inverse of exponential map. Given two points \mathbf{K} and \mathbf{X} on OM, we denote $\mathbf{H} = \text{Log}_{\mathbf{K}} \mathbf{X}$ where $\text{Log} : \mathcal{O} \times \mathcal{O} \rightarrow T\mathcal{O}$. It outputs the tangent vector \mathbf{H} at \mathbf{K} pointing towards \mathbf{X} , describing the shortest path curve from \mathbf{K} to \mathbf{X} . Formally, Log is written as:

$$\begin{aligned} \text{dist}(\mathbf{K}, \mathbf{X}) &= \sqrt{\sum_{i=1}^p \arccos^2(\text{diag}(\mathbf{K}^T \mathbf{X}))_i} \\ P_{\mathbf{K}}(\mathbf{X} - \mathbf{K}) &= (\mathbf{X} - \mathbf{K}) - \mathbf{K} \text{diag}(\mathbf{K}^T (\mathbf{X} - \mathbf{K})) \\ \text{Log}_{\mathbf{K}}(\mathbf{X}) &= \frac{\text{dist}(\mathbf{K}, \mathbf{X})}{\|P_{\mathbf{K}}(\mathbf{X} - \mathbf{K})\|} P_{\mathbf{K}}(\mathbf{X} - \mathbf{K}) \end{aligned} \quad (4)$$

3.4. Few-Shot Learning

Given a *base* training set $\mathbb{D}_{base} := \{ \mathbf{I}_i, y_i : \mathbf{I}_i \in \mathbb{I}_{base}, y_i \in \mathbb{C}_{base} \}_{i=1}^{N_{base}}$, \mathbf{I}_i denotes raw image sample i from base image set \mathbb{I}_{base} and y_i is its associated label in the base training set \mathbb{C}_{base} . In the few-shot scenario, given a *novel* test set $\mathbb{D}_{novel} := \{ \mathbf{I}_i, y_i : \mathbf{I}_i \in \mathbb{I}_{novel}, y_i \in \mathbb{C}_{novel} \}_{i=1}^{N_{novel}}$, where $\mathbb{D}_{base} \cap \mathbb{D}_{novel} = \phi$, we create the c -way k_S -shot *tasks* randomly sampled with a few number of examples. Specially, we sample c classes from \mathbb{C}_{novel} and for each sampled class we choose k_S samples randomly. These selected samples form *support* set S with size $|S| := k_S \times c$. Similarity, we obtain *query* set Q with size $|Q| := k_Q \times c$ by sampling k_Q unlabeled (unseen) examples for each classes randomly.

4. Method

In this section, we firstly present the detail of RSSPP. Then, we describe the embedded method on OM. Next, we design ODC and show the strategy for initializing parameters. Finally, we reveal the loss function and optimization on OM. The overview of our method is illustrated in Figure 2.

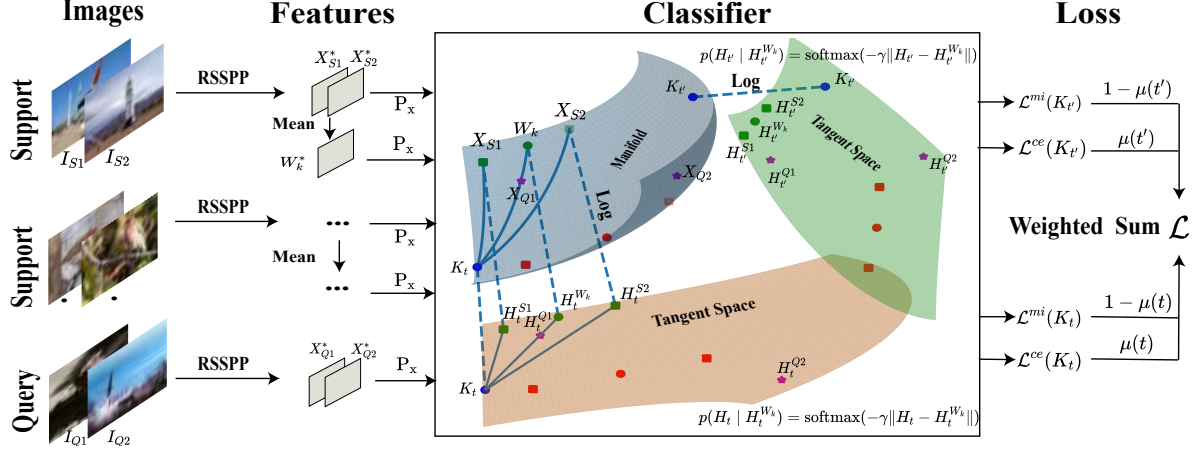


Figure 2: The overall pipeline of our method. For clarity and visibility, we take 2-way 2-shot classification as an example, plot partially, and push the tangent spaces away from the manifold in the classifier. Given images $\{I_S, I_Q\}$, we obtain features $\{X_S^*, X_Q^*\}$ by RSSPP and the prototypes W_k^* . Then, they are embedded as points $\{X_S, X_Q, W_k\}$ by P_x . Next, the geodesic flows between points and anchors $\{K_t, K_{t'}\}$ are transformed to tangent vectors $\{H_t^S, H_t^Q, H_t^{W_k}, \dots\}$ in the tangent spaces by Log. Finally, we calculate the probability distribution by softmax and total loss \mathcal{L} for fine-tuning.

4.1. Region Self-Attention with Spatial Pyramid Pooling

To make a trade-off [39] for features between generalization and discrimination from a single image, we offer the non-parametric region self-attention with spatial pyramid pooling (RSSPP). As illustrated in Figure 3, given an image I and a pre-trained CNN φ from the base datasets, we obtain the image features $F \in \mathbb{R}^{n \times h \times w}$ by $F := \varphi(I)$, where n , h , and w denote the feature dimension, the spatial height and width of the feature map respectively. To get the generalization feature set \mathbb{F}_∂ , the RSSPP utilizes multi-kernel max-pooling similar to SPP:

$$\begin{aligned} s_i &:= (\lfloor h/i \rfloor, \lfloor w/i \rfloor) \\ k_i &:= (h - (i+1) \cdot \lfloor h/i \rfloor, w - (i+1) \cdot \lfloor w/i \rfloor) \\ \mathbb{F}_\partial &:= \{F_i = \text{MP}(F, k_i, s_i) \in \mathbb{R}^{n \times i \times i}\}_{i=1}^p \end{aligned} \quad (5)$$

where $\text{MP}(\cdot)$ denotes the max-pooling operator, p is the number of kernels satisfying $p < \min(h, w)$, k_i is the max-pooling kernel size, s_i is the stride, and $\lfloor \cdot \rfloor$ is the largest integer that is less than or equal to \cdot . Next, to improve discriminative ability, we utilize the self-attention mechanism and denote the key encoder, value encoder and query as:

$$\begin{aligned} k_i &:= \frac{\text{GAP}(F_i) \cdot \text{GAP}(F_p)}{\text{GAP}(F_1)} + \text{GAP}(F_i) \\ v_i &:= \text{GAP}(F_i) \\ q &:= \text{GAP}(F_p) \end{aligned} \quad (6)$$

where $k_i, v_i, q \in \mathbb{R}^n$, $F_i \in \mathbb{F}_\partial$ and GAP is the global average-pooling. To the end, the key encoder enhances the local discriminative region furthermore, the value en-

coder encodes the current feature map and the query encoder stands for applying the global average pooling on the whole feature map. Finally, following the self-attention and the residual shortcut (SA), we obtain the features x_i^* :

$$x_i^* := \text{SA}(q, k_i, v_i) := \text{softmax}\left(\frac{qk_i^T}{\sqrt{n}}\right)v_i + v_i \quad (7)$$

Finally, we collect the feature matrix $X^* \in \mathbb{R}^{n \times p}$ by $X^* := \text{Cat}(\{x_i^*\}_{i=1}^p)$, where $\text{Cat}(\cdot)$ denotes concatenating the given sequence of features. For simplicity, we formulate the series connected functions in this section as RSSPP:

$$\text{RSSPP} := \text{Cat} \circ \text{SA} \circ \text{MP} \circ \varphi \quad (8)$$

In FSL, we obtain the support features set $S^* := \{X_S^* = \text{RSSPP}(I) \in \mathbb{R}^{n \times p}\}_{I \in S}$ and the query features set $Q^* := \{X_Q^* = \text{RSSPP}(I) \in \mathbb{R}^{n \times p}\}_{I \in Q}$.

4.2. Embedded on the Oblique Manifold

Since $\text{diag}(X^{*T} X^*) \neq I_p$, the feature matrix X^* is not a member of OM. We apply a projector $P_x(\cdot)$ to get the manifold-valued feature matrix X :

$$X := P_x(X^*) = \text{Cat}\left(\left\{\frac{x_i^*}{\|x_i^*\|}\right\}_{i=1}^p\right) \quad (9)$$

In this way, the support embedding set and query embedding set are collected as $S := \{X_S = P_x(X_S^*) \in \mathbb{R}^{n \times p}\}$ and $Q := \{X_Q = P_x(X_Q^*) \in \mathbb{R}^{n \times p}\}$ respectively.

4.3. Classification on the Oblique Manifold

In this work, we define the oblique distance-based classifier (ODC) with weights $W := \{W_k \in \mathbb{R}^{n \times p}\}_{k=1}^c$ and

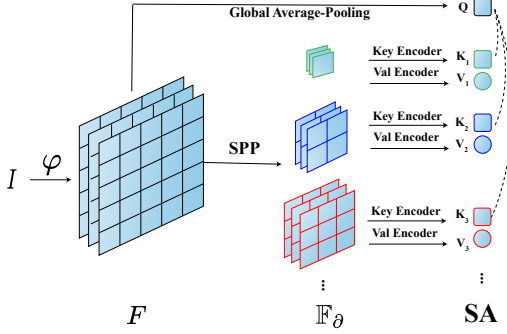


Figure 3: The overview of Region Self-attention with Spatial Pyramid Pooling. Both key and value encoders are non-parametric.

anchors $\mathbb{K} := \{\mathbf{K}_t\}_{t=0}^\tau$, where $\tau + 1$ is the length of the anchors set. Both weights and anchors are points on OM.

Initialize Anchors. ODC seeks the anchor \mathbf{K}_t so that the tangent vector $\mathbf{H}_t = \text{Log}_{\mathbf{K}_t}(\mathbf{X})$ is as representative as possible. Commonly, Log is a helpful approximation of manifold in a local neighborhood of the anchor \mathbf{K}_t , however, its representation capability decreases to points too far from \mathbf{K}_t . Thus, it is critical to initialize proper anchors.

Intuitively, \mathbf{K}_t are required to be close to the Karcher mean (KM) [17] of associated points. However, the computation of exact KM is an NP hard problem. Instead, we propose the pseudo-KM as initial value of \mathbf{K}_t . In particular, considering \mathbf{X}^* in Euclidean space, it is reasonable to calculate Euclidean anchors \mathbf{K}_0^* by weighted mean over the support features: $\mathbf{K}_0^* := \frac{1}{|S|} \sum_{\mathbf{X}_S^* \in S^*} \mathbf{X}_S^*$. Then, we get manifold anchors \mathbf{K}_0 : $\mathbf{K}_0 = \text{P}_x(\mathbf{K}_0^*)$.

Furthermore, considering the potential ill-distribution gap between the support samples and the query samples in FSL, we initialize anchors with a strategy that the associated samples are selected carefully in the transductive settings. Formally, the general initial \mathbf{K}_t is computed by:

$$\mathbf{K}_t := \begin{cases} \text{P}_x\left(\frac{\sum_{\mathbf{X}_S^* \in S^*} \mathbf{X}_S^*}{|S|}\right) & \tau = 0 \\ \text{P}_x\left(\frac{(\tau-t) \cdot \sum_{\mathbf{X}_S^* \in S^*} \mathbf{X}_S^* + t \cdot \sum_{\mathbf{X}_Q^* \in Q^*} \mathbf{X}_Q^*}{(\tau-t) \cdot |S| + t \cdot |Q|}\right) & \tau > 0 \end{cases} \quad (10)$$

Initialize Weights. Empirically, we observe that it gains from initializing weights manually. Similar to the anchors, we calculate the prototypes of support features with weighted mean, and embed the class prototypes to OM:

$$\mathbf{W}_k := \text{P}_x\left(\frac{1}{k_S} \sum_{\mathbf{X}_S^* \in S^*} \mathbf{X}_S^* \delta(y(\mathbf{X}_S^*) = k)\right) \quad (11)$$

where $y(\mathbf{X}_S^*)$ is the label of \mathbf{X}_S^* , and $\delta(y(\mathbf{X}_S^*) = k) = 1$ if and only if $y(\mathbf{X}_S^*)$ equals k .

Classification. The geodesic distance is difficult to compute and modern Euclidean transformation can not be applied directly. Since the tangent vector offers Euclidean

Algorithm 1: Pseudo-code for Classification

Paras. : Number of Anchors τ , Number of Kernels p , Iterations $iter$, Learning rate λ_r , Temperature γ , Weights factor $\{\lambda, \alpha, \mu(t)\}$

Input: Pre-trained CNN φ . Task $\{S, Q\}$

Compute $\{S^*, Q^*\}$, $\{S, Q\}$ by Equation 8, 9;

Initialize \mathbb{K}, \mathbb{W} by Equation 10, 11;

for $i \leftarrow 0$ **to** $iter$ **do**

for $t \leftarrow 0$ **to** τ **do**

 Compute $\mathbf{H}_t, \mathbf{H}_{t_k}^W$ by Equation 4;

 Compute $p_{k,t}^X$ by Equation 12;

 Compute $\mathcal{L}^{ce}, \mathcal{L}^{mi}$ by Equation 13;

end

 Compute \mathcal{L} by Equation 16;

 Update \mathbb{K}, \mathbb{W} by Equation 17 and 3;

end

Output: Query predictions are computed by Equation 15;

properties, we transform the geodesic distance as the tangent vector \mathbf{H} , which is the shortest path from anchor \mathbf{K}_t pointing towards \mathbf{X} on OM by Log. Then, we use softmax to calculate the distribution.

Specially, following Equation 4, we acquire the support and query tangent vectors \mathbf{H}_t by $\mathbf{H}_t := \text{Log}_{\mathbf{K}_t}(\mathbf{X})$ where $\mathbf{X} \in S$ or Q and weights tangent vectors $\mathbf{H}_t^{W_k}$ by $\mathbf{H}_t^{W_k} := \text{Log}_{\mathbf{K}_t}(\mathbf{W}_k)$ at each anchor \mathbf{K}_t respectively. The distribution over classes $\mathbb{P}(y = k | \mathbf{X}, \mathbf{K}_t, \mathbb{W})$ in the tangent spaces is denoted as $p_{k,t}^X$:

$$\begin{aligned} p_{k,t}^X &:= \frac{\exp(-\gamma \|\mathbf{H}_t - \mathbf{H}_t^{W_k}\|^2)}{\sum_{k'} \exp(-\gamma \|\mathbf{H}_t - \mathbf{H}_t^{W_{k'}}\|^2)} \\ &= \frac{\exp(-\gamma \|\text{Log}_{\mathbf{K}_t}(\mathbf{X}) - \text{Log}_{\mathbf{K}_t}(\mathbf{W}_k)\|^2)}{\sum_{k'} \exp(-\gamma \|\text{Log}_{\mathbf{K}_t}(\mathbf{X}) - \text{Log}_{\mathbf{K}_t}(\mathbf{W}_{k'})\|^2)} \end{aligned} \quad (12)$$

where γ is a temperature parameter, and $\mathbf{X} \in S$ or Q .

4.4. Optimization on the Oblique Manifold

Classification Loss . We follow TIM [4] and define the cross-entropy loss $\mathcal{L}^{ce}(\mathbb{W}, \mathbf{K}_t)$ with regard to the supervised support set, and the weighted mutual information $\mathcal{L}^{mi}(\mathbb{W}, \mathbf{K}_t)$ for the unlabeled query samples in the tangent spaces as:

$$\begin{aligned} \mathcal{L}^{ce}(\mathbb{W}, \mathbf{K}_t) &:= -\lambda \frac{1}{|S|} \sum_{\mathbf{X} \in S} \sum_{k=1}^c \delta(y(\mathbf{X}) = k) \log(p_{k,t}^X) \\ \mathcal{L}^{mi}(\mathbb{W}, \mathbf{K}_t) &:= -\alpha \frac{1}{|Q|} \sum_{\mathbf{X} \in Q} \sum_{k=1}^c p_{k,t}^X \log(p_{k,t}^X) \\ &\quad + \sum_{k=1}^c \frac{1}{|Q|} \sum_{\mathbf{X} \in Q} p_{k,t}^X \log\left(\frac{1}{|Q|} \sum_{\mathbf{X} \in Q} p_{k,t}^X\right) \end{aligned} \quad (13)$$

with non-negative hyper-parameter $\alpha = \lambda = 0.1$.

Method	Type	Backbone	<i>mini-ImageNet</i>		<i>tiered-ImageNet</i>	
			1-shot	5-shot	1-shot	5-shot
TPN [32]	Trans.	ResNet-12	59.46	75.64	-	-
TEAM [40]	Trans.	ResNet-18	60.07	75.90	-	-
Transductive tuning [8]	Trans.	ResNet-12	62.35 ± 0.66	74.53 ± 0.54	-	-
MetaOpt [28]	Induc.	ResNet-12	62.65 ± 0.61	78.63 ± 0.46	65.99 ± 0.72	81.56 ± 0.53
DSN [45]	Induc.	ResNet-12	64.60 ± 0.73	79.51 ± 0.50	67.39 ± 0.82	82.85 ± 0.56
SimpleShot [57]	Induc.	ResNet-18	63.10 ± 0.20	79.92 ± 0.14	69.68 ± 0.22	84.56 ± 0.16
S2M2 _R [34]	Induc.	ResNet-18	64.06 ± 0.18	80.58 ± 0.12	-	-
Distill [51]	Induc.	ResNet-12	64.82 ± 0.60	82.14 ± 0.43	71.52 ± 0.69	86.03 ± 0.49
DeepEMD [63]	Induc.	ResNet-12	65.91 ± 0.82	82.41 ± 0.56	71.16 ± 0.87	86.03 ± 0.58
LaplacianShot [66]	Trans.	ResNet-18	72.11 ± 0.19	82.31 ± 0.14	78.98 ± 0.21	86.39 ± 0.16
TIM [4]	Trans.	ResNet-18	73.9	85.0	79.9	88.5
Ours	Induc.	ResNet-18	63.98 ± 0.29	82.47 ± 0.44	70.50 ± 0.31	86.71 ± 0.49
Ours	Trans.	ResNet-18	77.20 ± 0.36	87.11 ± 0.42	83.73 ± 0.36	90.46 ± 0.46
LEO [44]	Induc.	WRN28-10	61.76 ± 0.08	77.59 ± 0.12	66.33 ± 0.05	81.44 ± 0.09
CC+rot [14]	Induc.	WRN28-10	62.93 ± 0.45	79.87 ± 0.33	70.53 ± 0.51	84.98 ± 0.36
MatchingNet [53]	Induc.	WRN28-10	64.03 ± 0.20	76.32 ± 0.16	-	-
FEAT [61]	Induc.	WRN28-10	65.10 ± 0.20	81.11 ± 0.14	70.41 ± 0.23	84.38 ± 0.16
SimpleShot [57]	Induc.	WRN28-10	65.87 ± 0.20	82.09 ± 0.14	70.90 ± 0.22	85.76 ± 0.15
S2M2 _R [34]	Induc.	WRN28-10	64.93 ± 0.18	83.18 ± 0.11	73.71 ± 0.22	88.59 ± 0.14
Transductive tuning [8]	Trans.	WRN28-10	65.73 ± 0.68	78.40 ± 0.52	73.34 ± 0.71	85.50 ± 0.50
SIB [24]	Trans.	WRN28-10	70.0 ± 0.6	79.2 ± 0.4	-	-
BD-CSPN [31]	Trans.	WRN28-10	70.31 ± 0.93	81.89 ± 0.60	78.74 ± 0.95	86.92 ± 0.63
LaplacianShot [4]	Trans.	WRN28-10	74.86 ± 0.19	84.13 ± 0.14	80.18 ± 0.21	87.56 ± 0.15
IFSL [62]	Trans.	WRN28-10	73.51 ± 0.56	83.21 ± 0.33	83.07 ± 0.52	88.69 ± 0.33
TIM [4]	Trans.	WRN28-10	77.8	87.4	82.1	89.8
Ours	Induc.	WRN28-10	66.78 ± 0.30	85.29 ± 0.41	71.54 ± 0.29	87.79 ± 0.46
Ours	Trans.	WRN28-10	80.64 ± 0.34	89.39 ± 0.39	85.22 ± 0.34	91.35 ± 0.42

Table 1: **Compared with the state-of-the-art on *mini-ImageNet* and *tiered-ImageNet*.** The results are 5-way few-shot classification scores averaged on 900 episodes for 5-shot and 4000 episodes for 1-shot, with 95% confidence interval. “-” signifies the result is available. “Trans.” and “Indus.” stand for the transductive and instructive settings.

As aforementioned, the representation ability of Log decreases to points too far from anchors. It is not advisable to apply the same weight factors of $\mathcal{L}^{ce}(\mathbb{W}, \mathbf{K}_t)$ and $\mathcal{L}^{mi}(\mathbb{W}, \mathbf{K}_t)$ over all the anchors. Intuitively, if t is smaller, \mathbf{K}_t will be closer to the support points. Then the support tangent vectors based \mathbf{K}_t will be more credible, the large weight factor is suitable and vice versa. However, when $t = \frac{\tau}{2}$, \mathbf{K}_t will be the global mean of all samples from both the support set and the query set. It conveys that the support tangent vectors at \mathbf{K}_t are also reliable due to robust to the outliers, and then it requires the large weight factor. Based on the above observations, we propose the weight factor $\mu(t)$ over t on $\mathcal{L}^{ce}(\mathbb{W}, \mathbf{K}_t)$ empirically:

$$\mu(t) := \begin{cases} 1 & \tau = 0 \\ \frac{-t(2t-\tau)^2 + \tau^3}{\tau^3} & \tau > 0 \end{cases} \quad (14)$$

Finally, the predicts of query $s^{\mathbf{X}}$ are:

$$s^{\mathbf{X}} := \arg \max \begin{cases} p_{k,t}^{\mathbf{X}} & \tau = 0 \\ \sum_t (1 - \mu(t)) p_{k,t}^{\mathbf{X}} & \tau > 0 \end{cases} \quad (15)$$

where $\mathbf{X} \in \mathbb{Q}$ and the classification loss $\mathcal{L} := \mathcal{L}(X, W)$ is:

$$\begin{aligned} \mathcal{L}(\mathbb{W}, \mathbb{K}) \\ := \frac{\sum_{t=0}^{\tau} (\mu(t) \mathcal{L}^{ce}(\mathbb{W}, \mathbf{K}_t) + (1 - \mu(t)) \mathcal{L}^{mi}(\mathbb{W}, \mathbf{K}_t))}{\sum_{t=0}^{\tau} \mu(t)} \end{aligned} \quad (16)$$

Optimization. Both \mathbb{W} and \mathbb{K} have a similar optimization process. As an example, \mathbb{W} is updated by Riemannian stochastic gradient descent (RSGD) [2, 3] which updates parameters along the shortest path in manifold and avoiding leaving the manifold.

Specially, we firstly compute Euclidean gradient $\nabla_{\mathbb{W}} \mathcal{L}$ with respect to \mathbb{W} by $\nabla_{\mathbb{W}} \mathcal{L} := \frac{d}{d\mathbb{W}} \mathcal{L}(\text{Log}_{\mathbf{K}} \mathbf{X}, \text{Log}_{\mathbf{K}} \mathbf{W})$. Then, we project the Euclidean gradient $\nabla_{\mathbb{W}} \mathcal{L}$ to Riemannian gradient $G_{\mathbb{W}} \mathcal{L}$, which means the tangent vectors $\nabla_{\mathbb{W}} \mathcal{L}$ at \mathbb{W} pointing to the updated parameters $\mathbb{W}^{(k+1)}$. Referring to Equation 3, $\mathbb{W}^{(k+1)}$ is computed by:

$$\mathbb{W}^{(k+1)} := \text{Exp}_{\mathbb{W}^{(k)}} - \lambda_r \nabla_{\mathbb{W}^{(k)}} \mathcal{L} \quad (17)$$

where λ_r is the learning rate of RSGD. The pseudo-code for classification on OM is shown in Algorithm 1.

Method	Type	CUB		M \rightarrow C
		1-shot	5-shot	5-shot
MAML [12]	Induc.	68.42	83.47	51.34
MatchingNet [53]	Induc	73.49	84.45	53.07
ProtoNet [46]	Induc	72.99	86.64	62.02
RelationNet [48]	Induc	68.58	84.05	57.71
IFSL [62] [†]	Induc	-	-	60.05
Chen [7]	Induc	67.02	83.58	65.57
SimpleShot [57]	Induc	70.28	86.37	65.63
S2M2 _R [34]	Induc.	71.81	86.22	70.44
DeepEMD [63] [†]	Induc	75.65	88.69	-
IFSL [62] [‡]	Trans.	-	-	62.07
LaplacianShot [66]	Trans.	80.96	88.68	66.33
TIM [4]	Trans.	82.2	90.8	71.0
Ours	Induc.	78.24	92.15	72.47
Ours	Trans.	85.87	94.97	74.11

Table 2: Results for 5-shot and 1-shot CUB, and 5-shot for cross-domain on *mini-ImageNet* \rightarrow CUB with 5-way and backbone ResNet18. [†] is based on ResNet12 and [‡] utilizes ResNet10.

5. Experiments

5.1. Implementation Details

Backbone. In all of our experiments, we take ResNet-18 and WRN28-10 as backbone.

Training. Follow [66, 4], the backbone is trained with standard cross-entropy on the base classes in Euclidean space with L2-normalization activations. In this way, the features will fall on the oblique manifold naturally. The training epochs are 90, and the learning rate of SGD optimizer is 0.1 initially, which is divided by 10 at epochs 45 and epochs 66. The images are resized to 84×84 . The batch size is 256 for ResNet-18 and 128 for WRN28-10. Same as [66], we apply data augmentation, *e.g.* color jitter, random cropping, and random horizontal flipping and utilize label-smoothing with the parameter 0.1.

Fine-tuning and Hyper-parameters. The backbones are fixed during fine-tuning phase. We evaluate 4000 tasks for 5-way 1-shot and 900 tasks for 5-way 5-shot sampled from test datasets randomly and report the average accuracy along with 95% confidence interval. To stay in line with TIM [4], we set $\alpha = 0.1$, $\lambda = 0.1$ in Equation 13 and $\gamma = 7.5$ in Equation 12. Empirically, we set $q = 11$ in Equation 5, $\tau = 14$ for transductive inference, and $\tau = 0$ in Equation 10 for inductive inference. We use the RSGD [2] optimizer with the learning rate $\lambda_r = 0.1$ and the iterations for each task is 100.

5.2. Datasets

Three popular benchmark datasets are conducted: *mini-ImageNet* [53], *tiered-ImageNet* [42], and Caltech-UCSD Birds-200-2011 (CUB) [54].

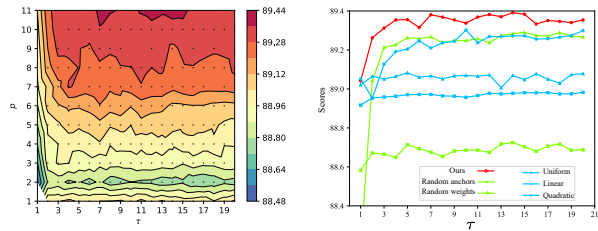


Figure 4: Ablation study. (a) Contour plot: visualization of classification scores over τ and p . Each point stands for a test with associated parameters. (b) Score lines graph: the red line denotes our method, the chartreuse green lines denote studying on the initialized methods, and the sky blue lines mean researching the weight factor of loss.

mini-ImageNet. *mini-ImageNet* is the subset of *ImageNet* datasets [43]. It contains 60,000 images totally with 84×84 size and 100 classes, which are divided into 64, 16, and 20 for training, validation and testing respectively.

tiered-ImageNet. *tiered-ImageNet* is also another one of *ImageNet* with 779,165 images and 84×84 size in total. It includes 34 classes and 608 sub-classes. The splits of training, validation, and testing are 20, 6, and 8 respectively, based on the high-level classes to enlarge the semantic gap between training and testing.

Caltech-UCSD Birds-200-2011 (CUB). CUB is used for fine-grained bird classification with a minor difference. It contains 200 classes and 11,788 images in all and the classes are divided into 100, 50, and 50 for training, validation, and testing respectively.

5.3. Comparison with State-of-the-Art Methods

Few-shot image classification. From Table 1, we can observe that our method outperforms the state-of-the-art in the transductive settings, and the gains are consistent with different network models and datasets. Note that our inductive model beats other inductive methods in 5-shot scenario. It shows that OM is more discriminative than Euclidean space and with more data the generalization ability of OM are enhanced rapidly.

Fine-grained image classification. Table 2 reports the results of fine-grained 5-way 5-shot classification on CUB. Our transductive method outperforms the best state-of-the-art by about 3.6% and 4.2% margins in 1-shot and 5-shot respectively. This shows that precise geodesic distance is more suitable than Euclidean distance to distinguish close features.

Cross-domain (*mini-ImageNet* \rightarrow CUB). Cross-domain few-shot classification is a challenging scenario. Following [7], we train Resnet-18 with training set from *mini-ImageNet* and validation set from CUB. The results

Riemannian	ODC	RSSPP	Trans.	<i>mini-ImageNet</i>		<i>tiered-ImageNet</i>		CUB	
				1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Euclidean	✗	✗	✗	65.25	82.17	70.19	85.51	76.80	90.31
Euclidean	✗	✓	✓	66.39	83.33	64.26	84.19	81.25	91.86
L2-Euclidean	✗	✗	✗	67.61	82.24	66.50	84.00	77.30	90.10
L2-Euclidean	✗	✓	✓	68.33	83.83	65.78	84.25	81.22	92.48
Oblique	✓	✗	✗	66.00	84.34	71.29	87.27	77.98	91.51
Oblique	✓	✓	✗	66.78	85.29	71.54	87.79	78.24	92.15
Oblique	✓	✗	✓	80.22	88.71	84.70	91.20	85.01	94.37
Ours	✓	✓	✓	80.64	89.39	85.22	91.35	85.87	94.97

Table 3: The effectiveness of RSSPP, ODC and transductive on *mini-ImageNet*, *tiered-ImageNet*, and CUB, with backbone WRN28-10, WRN28-10 and Resnet18 respectively.

in Table 2 (rightmost column) is evaluated on 5-way 5-shot from test set of CUB. We can observe that both our methods outperform others by about 3% margins.

5.4. Ablation Study

Effects of RSSPP, ODC, and transductive. To check the effectiveness of components of our method, we made some comparison experiences. Results are reported in Table 3. In the first row, we replace the oblique manifold with Euclidean space, remove RSSPP and transductive settings, *i.e.*, $\tau = p = 0$, and utilize the global average-polling directly. In this way, the model acts as transforming features by centering [57]. The third and fourth rows show results of Euclidean with L2-normalization (noted “L2-Euclidean”) to compare the geodesic distance and Euclidean distance. We also carry out experiments to illustrate the efficacy of modules individually. The deference between the first four rows and the last four rows is: 1) The former are based on Euclidean distance while the latter are based on geodesic distance. 2) Parameters of the former are updated in Euclidean space while parameters of the latter are updated along the surface of manifold. The results show that the scores gain about 0.5% by RSSPP and 1%-3% by ODC in the transductive settings, and the ODC is better than Euclidean classifier.

Effects of τ and p . We conduct an ablation study on the effect of p and τ . Figure 4(a) is the contour plot that depicts how classification scores change with p and τ in 5-shot settings on WRN28-10, where p ranges from 1 to 11, τ ranges from 1 to 20, and both are integers. It can be observed that: 1) The larger p , the better scores may be due to a better generalization and discriminative ability. 2) As τ increases, the higher scores. The reason is that more local approximate representations are utilized. However, the scores do not increase steadily. The explanation may be that the weight factor of single well-approximated representation decreases in the proportion of total representations according to Equation 16.

The initialization of anchors and weights. We also study the effect of initialization of anchors and weights. The

results are reported in Figure 4(b) with chartreuse green. We observe that anchors with random initialization perform the worst. It hardly catches the local approximate representation without considering the structure of manifold-features, and the parameters may be updated in the wrong direction. Random initialization of weights also leads to worse performance than our method by 0.15% margin, which shows that it is necessary to initialize weights carefully.

The Weight factor of Loss. For multi-tangent spaces, suitable weight factors are required to determine the importance of scores according to the anchors, which engendered by the ratio of support points and query points. We propose another three functions of weight factors: 1) Uniform distribution, the factors of all terms are the same, *i.e.*, $\mu(t) = \frac{1}{2}$. 2) Linear function, the factors are reduced linearly, *i.e.*, $\mu(t) = 1 - \frac{t}{\tau}$. 3) Quadratic function when t moves far, the factor is reduced sharply, *i.e.*, $\mu(t) = 1 - (\frac{t}{\tau})^2$. We conduct experiments to compare these factor functions, and the results are shown in Figure 4(b) with deep sky blue. Our method outperforms best over all τ .

6. Conclusion

In this paper, we leverage the geodesic distance on the oblique manifold to solve the few-shot classification. The geodesic distance is transformed to vector in the tangent space and the modern machine learning tools can be utilized. We introduce a novel RSSPP for balancing generalization and discriminative ability, oblique distance-based classifier with new initial methods and a novel loss function for accurate classification. Finally, We validate the effectiveness on popular datasets. In the future, we will focus on how to integrate the local tangent space effectively or optimize the oblique manifold directly without resorting to the tangent space.

Acknowledgement. This work was supported in part by the Research Project of ZJU-League Research & Development Center, Zhejiang Lab under Grant 2019KD0AB01.

References

- [1] P. . Absil and K. A. Gallivan. Joint diagonalization on the oblique manifold for independent component analysis. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 5, pages V–V, 2006. 1, 3
- [2] Gary Bécigneul and Octavian-Eugen Ganea. Riemannian adaptive optimization methods. In *Proc. of ICLR*. OpenReview.net, 2019. 6, 7
- [3] Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, 2013. 6
- [4] Malik Boudiaf, Imtiaz Masud Ziko, Jérôme Rony, Jose Dolz, Pablo Piantanida, and Ismail Ben Ayed. Information maximization for few-shot learning. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 1, 2, 3, 5, 6, 7
- [5] R. Chakraborty, J. Bouza, J. Manton, and B. C. Vemuri. Manifoldnet: A deep neural network for manifold-valued data with applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. 3
- [6] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. SCA-CNN: spatial and channel-wise attention in convolutional networks for image captioning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6298–6306. IEEE Computer Society, 2017. 2, 3
- [7] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *Proc. of ICLR*. OpenReview.net, 2019. 7
- [8] Guneet Singh Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. In *Proc. of ICLR*. OpenReview.net, 2020. 3, 6
- [9] Nikita Dvornik, Julien Mairal, and Cordelia Schmid. Diversity with cooperation: Ensemble methods for few-shot classification. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3722–3730. IEEE, 2019. 1
- [10] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Selecting relevant features from a multi-domain representation for few-shot classification. In *European Conference on Computer Vision*, pages 769–786. Springer, 2020. 2
- [11] Luther Pfahler Eisenhart. *Riemannian geometry*, volume 51. Princeton university press, 1997. 1
- [12] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017. 2, 7
- [13] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3146–3154. Computer Vision Foundation / IEEE, 2019. 3
- [14] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8058–8067. IEEE, 2019. 6
- [15] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 2066–2073. IEEE Computer Society, 2012. 3
- [16] Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In Dimitris N. Metaxas, Long Quan, Alberto Sanfeliu, and Luc Van Gool, editors, *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, pages 999–1006. IEEE Computer Society, 2011. 3
- [17] Karsten Grove and Hermann Karcher. How to conjugate 1-close group actions. *Mathematische Zeitschrift*, 132(1):11–20, 1973. 2, 5
- [18] Mehrtash Tafazzoli Harandi, Conrad Sanderson, Sareh Abolahrari Shirazi, and Brian C. Lovell. Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 2705–2712. IEEE Computer Society, 2011. 3
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9):1904–1916, 2015. 1, 2, 3
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 1, 3
- [21] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006. 1, 3
- [22] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933. 1, 3
- [23] Han Hu, Zheng Zhang, Zhenda Xie, and Stephen Lin. Local relation networks for image recognition. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3463–3472. IEEE, 2019. 3
- [24] Shell Xu Hu, Pablo Garcia Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil D. Lawrence, and Andreas C. Damianou. Empirical bayes transductive meta-learning with synthetic gradients. In *Proc. of ICLR*. OpenReview.net, 2020. 6

- [25] Zhiwu Huang and Luc Van Gool. A riemannian network for SPD matrix learning. In Satinder P. Singh and Shaul Markovitch, editors, *Proc. of AAAI*, pages 2036–2042. AAAI Press, 2017. 3
- [26] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Projection metric learning on grassmann manifold with application to video based face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 140–149. IEEE Computer Society, 2015. 1, 3
- [27] Youngsung Kim, Jinwoo Shin, Eunho Yang, and Sung Ju Hwang. Few-shot visual reasoning with meta-analogical contrastive learning. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 1, 2
- [28] Kwonjoon Lee, Subhansu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 10657–10665. Computer Vision Foundation / IEEE, 2019. 2, 6
- [29] 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 Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 10276–10286, 2019. 2
- [30] Yann Lifchitz, Yannis Avrithis, Sylvaine Picard, and Andrei Bursuc. Dense classification and implanting for few-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 9258–9267. Computer Vision Foundation / IEEE, 2019. 2
- [31] Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. In *ECCV*, 2020. 6
- [32] 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. In *Proc. of ICLR*. OpenReview.net, 2019. 3, 6
- [33] Yaoyao Liu, Bernt Schiele, and Qianru Sun. An ensemble of epoch-wise empirical bayes for few-shot learning. In *European Conference on Computer Vision*, pages 404–421. Springer, 2020. 1
- [34] Puneet Mangla, Mayank Singh, Abhishek Sinha, Nupur Kumari, Vineeth N Balasubramanian, and Balaji Krishnamurthy. Charting the right manifold: Manifold mixup for few-shot learning. In *WACV*, pages 2207–2216, Snowmass Village, CO, USA, 2020. IEEE. 1, 6, 7
- [35] Federico Monti, Davide Boscaïni, Jonathan Masci, Emanuele Rodolà, Jan Svoboda, and Michael M. Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5425–5434. IEEE Computer Society, 2017. 3
- [36] Xuan Son Nguyen, Luc Brun, Olivier Lezoray, and Sébastien Bougleux. A neural network based on SPD manifold learning for skeleton-based hand gesture recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 12036–12045. Computer Vision Foundation / IEEE, 2019. 1, 3
- [37] Eunbyung Park and Junier B. Oliva. Meta-curvature. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 3309–3319, 2019. 2
- [38] Massimiliano Patacchiola, Jack Turner, Elliot J. Crowley, Michael O Boyle, and Amos J Storkey. Bayesian meta-learning for the few-shot setting via deep kernels. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16108–16118. Curran Associates, Inc., 2020. 1
- [39] Rémi Pautrat, Viktor Larsson, Martin R. Oswald, and Marc Pollefeys. Online invariance selection for local feature descriptors. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *ECCV 2020*, volume 12347, pages 707–724, Cham, 2020. Springer International Publishing. 1, 4
- [40] Limeng Qiao, Yemin Shi, Jia Li, Yonghong Tian, Tiejun Huang, and Yaowei Wang. Transductive episodic-wise adaptive metric for few-shot learning. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3602–3611. IEEE, 2019. 3, 6
- [41] Aravind Rajeswaran, Chelsea Finn, Sham M. Kakade, and Sergey Levine. Meta-learning with implicit gradients. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 113–124, 2019. 2
- [42] 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 *Proc. of ICLR*. OpenReview.net, 2018. 7
- [43] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 7
- [44] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Had-

- sell. Meta-learning with latent embedding optimization. In *Proc. of ICLR*. OpenReview.net, 2019. 6
- [45] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtaash Harandi. Adaptive subspaces for few-shot learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 4135–4144. IEEE, 2020. 6
- [46] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4077–4087, 2017. 2, 7
- [47] L. S. Souza, N. Sogi, B. B. Gatto, T. Kobayashi, and K. Fukui. An interface between grassmann manifolds and vector spaces. In *CVPR Workshop*, pages 3695–3704, 2020. 2, 3
- [48] 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 *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1199–1208. IEEE Computer Society, 2018. 1, 2, 7
- [49] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, Mark Sandler, Andrew Howard, and Quoc V. Le. Mnasnet: Platform-aware neural architecture search for mobile. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2820–2828. Computer Vision Foundation / IEEE, 2019. 3
- [50] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. 1, 3
- [51] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: A good embedding is all you need? In *ECCV*, 2020. 2, 3, 6
- [52] Nickolay T. Trendafilov and Ross A. Lippert. The multimode procrustes problem. *Linear Algebra and its Applications*, 349(1):245–264, 2002. 1, 3
- [53] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 3630–3638, 2016. 2, 6, 7
- [54] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 7
- [55] R. Wang, X.-J. Wu, and J. Kittler. Graph embedding multi-kernel metric learning for image set classification with grassmannian manifold-valued features. *IEEE Transactions on Multimedia*, 23:228–242, 2021. 1, 3
- [56] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7794–7803. IEEE Computer Society, 2018. 3
- [57] Yan Wang, Wei-Lun Chao, Kilian Q. Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. In *arXiv:1911.04623*, 2019. 6, 7, 8
- [58] Jianre Wei, Jian Liang, Ran He, and Jinfeng Yang. Learning discriminative geodesic flow kernel for unsupervised domain adaptation. In *ICME*, pages 1–6, San Diego, CA, 2018. IEEE. 3
- [59] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, pages 3–19, 2018. 2, 3
- [60] Ling Yang, Liangliang Li, Zilun Zhang, Xinyu Zhou, Erjin Zhou, and Yu Liu. DPGN: distribution propagation graph network for few-shot learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 13387–13396. IEEE, 2020. 3
- [61] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8805–8814. IEEE, 2020. 6
- [62] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020. 1, 2, 6, 7
- [63] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover’s distance and structured classifiers. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12200–12210. IEEE, 2020. 2, 6, 7
- [64] Hongguang Zhang, Jing Zhang, and Piotr Koniusz. Few-shot learning via saliency-guided hallucination of samples. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2770–2779. Computer Vision Foundation / IEEE, 2019. 1
- [65] Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring self-attention for image recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10073–10082. IEEE, 2020. 3
- [66] Imtiaz Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning. In *International Conference on Machine Learning*, pages 11660–11670. PMLR, 2020. 1, 3, 6, 7