Efficient Few-Shot Classification using Self-Supervised Learning and Class Factor Analysis

Youngjae Lee
School of Computer Science and Engineering
Kyungpook National University
Daegu, Republic of korea
leeyj2711@gmail.com

Hyeyoung Park
School of Computer Science and Engineering
Kyungpook National University
Daegu, Republic of korea
hypark@knu.ac.kr

***Abstract*—Recently, significant advancements have been made in the few-shot classification task by integrating pre-trained self-supervised learning models. Although the self-supervised learning models have demonstrated their effectiveness, their application in few-shot scenarios, specifically in meta-training or fine-tuning, is computationally intensive and complicated. This paper introduces an efficient approach to address these challenges. We propose to use feature analysis methods instead of network model training. This method uses two factors that define the data generation model, resulting in easily classifiable features. The two factors are estimated from a set of different vectors from the train dataset and the test dataset. Although this method is simple compared to network learning, it provides good performance in experiments using few-shot classification benchmark datasets.**

Keywords—Few-shot classification, Feature analysis, Class factor, Environment factor Self-supervised learning,

I. INTRODUCTION

Few-shot classification is a challenging task in the field of machine learning, where notable progress has been made using pre-trained self-supervised learning models [1, 2, 3, 4]. Such models have the capability of capturing complex patterns from extensive datasets and have dramatically enhanced the performance of classifiers across diverse tasks. However, using these models in a few-shot task requires a complex meta-training or fine-tuning process.

The application of large-scale pre-trained models for few-shot classification often demands substantial computational resources, hindering their practical applicability in real-world problems. To mitigate this problem, our research takes a distinct approach by proposing a method that utilizes class factors and environment factors for feature extraction. Solves the limited train data problem by estimating environment factors from a set of difference vectors in the meta-train dataset. On the other hand, by estimating class factors from each test subset of episodes, we can obtain the good features for classification. Instead of the complicated task of updating parameters within a large model, Our approach aims to solve the problem of few-shot classification as a relatively simple task by extracting discriminative features directly from the data.

In this paper, we introduce a detailed examination of our proposed approach, provide the rationale behind the use of class factors [5, 6, 7], and elaborate on the steps involved in our methodology. In order to evaluate the effectiveness of our method, we conducted comprehensive experiments on four widely used benchmark datasets for few-shot classification. The results indicate the superior performance of our approach, which achieves the best results on two datasets and comes close to state-of-the-art performance on the remaining two datasets. These results emphasize the practical applicability and efficiency of the proposed method, positioning it as a competitive solution in the landscape of few-shot classification methods.

II. RELTED WORKS

A. Few-Shot Classification

In the domain of few-shot classification, conventional research is often based on small networks such as CNN-4-64 [8, 9, 10] and ResNet-12 [11, 12]. The conventional paradigm for general classification tasks is that larger models tend to perform better. However, this paradigm is subject to challenges in few-shot learning scenarios [13, 14]. This is because these models tend to overfit the classes in the train dataset and are not well suited to the few-shot task of classifying unseen classes.

On the other hand, the use of pre-trained self-supervised learning models has become a notable and potentially promising approach [15]. These models demonstrate superior feature extraction capabilities, learned from massive amounts of unlabeled data. This makes them suitable candidates for improving few-shot classification performance. Several studies have shown positive results when such models are applied to relatively large architectures for few-shot classification tasks [1, 2, 3, 4]. Despite these advances, a significant problem remained. The training of large-scale models for few-shot classification using conventional techniques proved to be a difficult challenge. The sophisticated properties of these models required complex procedures, including the need to fine-tune or meta-train these models. This made their application to few-shot tasks complicated and resource-intensive. To solve these problems, we propose a more efficient classification method by analyzing the given features in detail instead of additional network learning.

B. Class and Environment Factors for feature analysis

Finding efficient low-dimensional features for classification has been widely studied in classical pattern recognition research. One of them is the low-dimensional feature extraction method [7] based on the data generation model [16] defined as the combination of environment factors and class factors. Each factor that defines the generation model has its own unique characteristics. All classes share the same environment factor that represents the distortions caused by various environmental conditions.

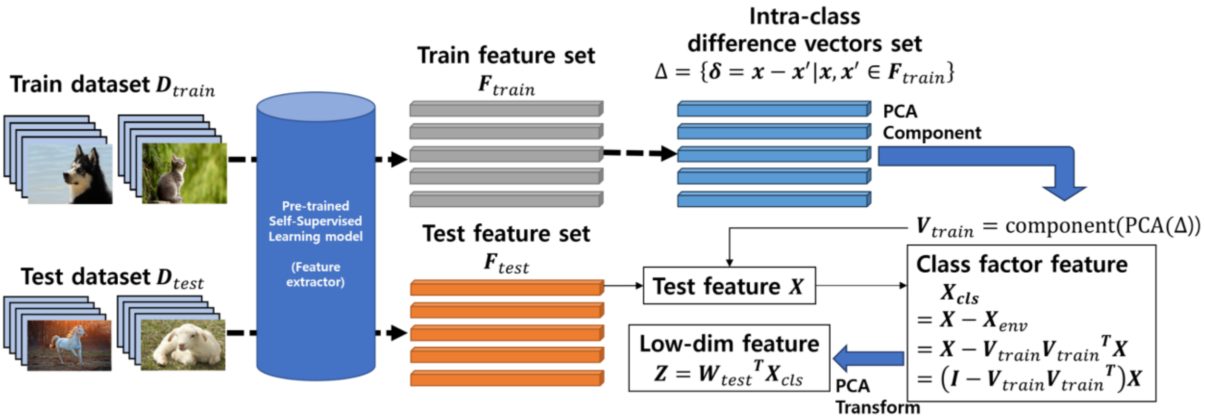

Fig. 1. The overall structure of the proposed method. Each dataset is converted into features by a pre-trained self-supervised learning model, of which intra-class difference vectors set Δ are extracted from the meta-train dataset to estimate $\boldsymbol{V}_{train}$ via PCA. This can be used to extract class factor features $\boldsymbol{X}_{cls}$ from the test feature set $\boldsymbol{F}_{test}$, which can then be subjected to PCA to obtain the final low-dimensional features $\boldsymbol{Z}$.

On the other hand, class factors represent the unique characteristics of each class. For estimating the environmental factors, we use the difference vector of two data belonging to the same class. The estimated probability distribution of the environment factors can be used or defining a probabilistic similarity measure like the method proposed by Moghaddam et al [17].

On the other hand, the class factors are emphasized to facilitate classification by excluding environmental factors. To achieve this, the residual space, which is the orthogonal complement of the principal subspace of environmental components, is first estimated and utilized for classification [18]. Methods that utilize this residual space have often shown promise in classical classification problems [19, 20, 21]. In this paper, we attempt to apply this classical research that has not worked well on complex problems to self-supervised model features.

III. Proposed Method

We propose a method that combines a pre-trained self-supervised learning model with feature analysis using class factors and environmental factors for the few-shot task. First, to apply the classic feature analysis method to a few-shot task, we need to make some modifications to the estimation methods. In the proposed method, the data generation model is defined as the sum of class factors and environment factors according to [16]. This means that data $\boldsymbol{x}$ belonging to an arbitrary class can be represented as

$$\boldsymbol{x} = \boldsymbol{x}_{cls} + \boldsymbol{x}_{env} \tag{1}$$

where $\boldsymbol{x}_{cls}$ is class components and $\boldsymbol{x}_{env}$ is environment components. We want to find low-dimensional factors that are easy to classify. If we call each of these low-dimensional factors $\boldsymbol{z}$ and $\boldsymbol{y}$, the data $\boldsymbol{x}$ can be represented as

$$\boldsymbol{x} = \boldsymbol{x}_{cls} + \boldsymbol{x}_{env} = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{V}\boldsymbol{y} \tag{2}$$

where $\boldsymbol{W}_{d\times p}$ and $\boldsymbol{V}_{d\times q}$ are transformation matrices of p-dimensional vector $\boldsymbol{z}$ and q-dimensional vector $\boldsymbol{y}$.

Of these, the environmental factor loading matrix is estimated using the difference vector $\boldsymbol{\delta}$ of the two data belonging to the same class, which can be written as

$$\boldsymbol{\delta} = \boldsymbol{x} - \boldsymbol{x}' \tag{3}$$

where $\boldsymbol{x}$ and $\boldsymbol{x}'$ belong to the same class. By using the decomposition of $\boldsymbol{x}$ given in (2), this can be rewritten as

$$\boldsymbol{\delta} = (\boldsymbol{Wz} - \boldsymbol{Wz}') + (\boldsymbol{Vy} - \boldsymbol{Vy}') \approx \boldsymbol{V}(\boldsymbol{y} - \boldsymbol{y}') + \boldsymbol{\epsilon}, \tag{4}$$

where $\boldsymbol{\epsilon}$ is noise with a small variance. Because of the class-dependent property of the low-dimensional class factor $\boldsymbol{z}$, two data samples belonging to the same class will have almost the same values of $\boldsymbol{z}$.

In the classical feature analysis for the conventional classification tasks, the set of difference vectors, $\Delta = \{\boldsymbol{\delta} = \boldsymbol{x} - \boldsymbol{x}' | \boldsymbol{x}\ and\ \boldsymbol{x}'\ has\ belongs\ to\ the\ same\ class\}$, is composed from the training set and used for estimating the environmental factors. Once the environmental factors and their subspace are obtained, its residual space can be used for finding class factors. However, in the case of the few-shot task, we assume the situation where the number of samples in each class is very limited and the classes in test set is unseen in the training set. Considering this specific situation, we separate the two estimation processes: one for the environment factor and the other for the class factor.

First, based on the assumption that all classes have the same environmental factors, we estimate the environmental factor loading matrix $\boldsymbol{V}_{train}$ by using the meta-train dataset $\boldsymbol{D}_{train}$, which has a sufficient number of samples. The estimated loading matrix can then be applied to the given test set with a limited number of samples from unseen classes. When the test dataset with n samples is given as a $n \times d$ matrix $\boldsymbol{X}$, the environmental factor loading matrix $\boldsymbol{V}_{train}$ can be applied to obtain a matrix $\boldsymbol{X}_{env}$ containing only the environmental components of the entire data $\boldsymbol{X}$, such as

$$\boldsymbol{X}_{env} = \boldsymbol{V}_{train}\boldsymbol{V}_{train}{}^{\boldsymbol{T}}\boldsymbol{X}. \tag{5}$$

Also, by definition of the data generation model, we can get a matrix with only class component, $\boldsymbol{X}_{cls}$, by simply subtracting the environment components $\boldsymbol{X}_{env}$ from the entire data $\boldsymbol{X}$, such as

$$X_{cls} = X - X_{env} = \left(I - V_{train}V_{train}{}^{T}\right)X. \quad (6)$$

Finally, we apply PCA to X_{cls} to obtain low-dimensional features that preserve class components as much as possible. This is the process of estimating the class factor transform matrix W. Unlike environmental factors, class factors are class-dependent, so it makes sense to estimate them only for the class to which the subject belongs. Therefore, we will estimate the class factor transform matrix W from a subset of each classified episode, which gives us the set of low-dimensional features Z, such as

$$Z = W^{T}X_{cls}. \quad (7)$$

The final classification stage included the prototype classification procedure. The porotype classification procedure is often used in few-shot tasks, where the mean vector of the training data for each class is defined as a prototype and test data are classified by comparing the similarity to each prototype. To evaluate the similarity between features and prototype, we adopted the Euclidean distance metric, which is widely used in other few-shot tasks.

TABLE I. CLASSIFICATION ALGORITHM OF THE PROPOSED METHOD

Algorithm 1 Algorithm for estimation and classification
Input: Train dataset D_{train}, Test dataset D_{test}
Output: Class classification results from query
Load pre-trained self-supervised model $f(\cdot)$
Construct train dataset feature $F_{train} = f(D_{train})$
Construct a randomized intra-class different vector set Δ_{train} from F_{train}.
$\Delta_{train} = \{\delta_{train} = x - x' \mid x$ and x' has belongs to the same class$\}$
Estimate environmental factor loading matrix
$V_{train} = \text{component}(PCA(\Delta_{train}))$
Construct train dataset feature $F_{test} = f(D_{test})$
Construct test matrix X from F_{test}
$X_{cls} = \left(I - V_{train}V_{train}{}^{T}\right)X$
Class factor loading matrix $W = \text{component}(PCA(X_{cls}))$
Feature extraction $Z = W^{T}\left(I - V_{train}V_{train}{}^{T}\right)X$
Support and query $S, Q = split(Z)$
Prototypes $P = mean(S)$
$d = L^{2}\text{-norm}(Q, P)$
Return $\text{argmin}(d)$

The detailed steps of the proposed method can be seen in Table I. The average classification performance across these episodes serves as a comprehensive metric, offering insights into the method's adaptability and effectiveness across diverse few-shot scenarios.

IV. EXPERIMENTAL RESULTS

We used four benchmark datasets to evaluate our proposed method: miniImageNet [32], tieredImageNet [33], CIFAR-FS [34], and CUB [35].

1) miniImageNet: Originating from the ILSVRC-12 dataset, miniImageNet comprises 100 classes and a total of 60,000 images. The dataset's images are characterized by a resolution of $84 \times 84 \times 3$, providing a testing ground for the capabilities of our proposed method.

2) tieredImageNet: Another subset of ILSVRC-12, tieredImageNet expands the scope with 608 classes and an approximate image count of 780,000. Similar to miniImageNet, the images in tieredImageNet adhere to a resolution of $84 \times 84 \times 3$.

3) CIFAR-FS: It is a variant of CIFAR-100 that is reconstructed for the few-shot task with a class-wise division of train/validation/test data rather than a sample-wise division within classes. Notably, the images in this dataset have a lower resolution of $32 \times 32 \times 3$, introducing a distinctive challenge compared to datasets with higher resolutions.

4) CUB: With a focus on categorizing birds by species, the CUB dataset encompasses 200 classes and around 12,000 images. The images in this dataset adhere to a resolution of $84 \times 84 \times 3$, aligning with the characteristics of miniImageNet.

To measure universal performance on the few-shot classification task, datasets with different features were used. The class divisions for every dataset presented in Table III are determined according to the previous works on. The performance evaluation comprised 1,000 episodes of few-shot classification.

In our research for an effective few-shot classification framework, we adopted the DINOv2 model as a feature extractor. Through a ViT-based architecture, DINOv2 showed remarkable self-supervised learning on a large dataset of 142M unlabeled images. The availability of pre-trained models with various sizes, conveniently accessible through the PyTorch Hub, supports its versatility.

Due to practical constraints in the difficult task of estimating the environmental factor loading matrix, a conservative approach was taken. Randomly selecting 120 vector pairs per class addressed physical limitations and ensured computational efficiency while maintaining the fidelity of the representation of the environmental factors.

TABLE II. NUMBER OF TRAIN, VALIDATION, AND TEST CLASSES IN EACH DATASET

dataset	*Train*	*Validation*	*Test*
miniImageNet	64	16	20
tieredImageNet	351	97	160
CIFAR-FS	64	16	20
CUB	100	50	50

In presenting our experimental results for two datasets based on ILSVRC-12, as detailed in Table II, we conducted a comprehensive comparison with past milestone papers and recent works. Remarkably, our focus was on works that employ pre-trained self-supervised learning models, which is similar to one of our approaches. From our analysis, we found that our accuracy significantly outperforms previous. Although we fell just short of achieving state-of-the-art performance, the significant accuracy achieved is encouraging. This is especially significant given the considerable effort that state-of-the-art models put into meta-training and fine-tuning their large models.

The efficiency of the proposed method is confirmed by the large difference in performance compared to other models, which excludes the exceptionally high-performance state-of-the-art model. This outstanding achievement is emphasized by the findings in Table III, which details the results of our experiments on CIFAR-FS

TABLE III. FEW-SHOT CLASSIFICATION ACCURACIES FOR THE 5-WAY 5-SHOT SETTINGS ON MINI-IMAGENET, TIERED-IMAGENET, CIFAR-FS AND CUB 200. THE BEST AND THE SECOND BEST SCORES ARE MARKED IN BOLD AND RED, RESPECTIVELY.

Method	***mini-ImageNet***	***Tiered-ImageNet***	***CIFAR-FS***	***CUB 200***
ProtoNet (NIPS 2017) [8]	81.10	84.10	86.70	88.80
Baseline (ICLR 2019) [9]	81.90	84.90	86.70	86.90
Baseline++ (ICLR 2019) [9]	82.10	85.40	87.40	87.10
PT+MAP (ICANN 2021) [22]	90.44	-	-	93.99
P>M>F (CVPR 2022) [4]	**98.40**	-	92.20	-
TRIDENT (TMLR 2022) [23]	95.95	**96.57**	-	-
HOT (NIPS 2022) [24]	84.40	87.30	87.50	91.50
FewTURE (NIPS 2022) [25]	84.50	86.43	88.98	-
PEMnE-BMS (algorithms 2022) [26]	91.53	91.09	91.86	96.43
SOT (arXiv 2022) [27]	-	-	**92.83**	**97.12**
BAVAR (AISTATS 2023) [28]	87.40	88.30	88.90	91.40
CPEA (ICCV 2023) [29]	87.06	90.12	88.98	-
SMKD (CVPR 2023) [30]	89.57	91.68	90.91	-
PUTM (ICCV 2023) [31]	88.10	92.20	90.40	93.70
Proposed	**97.31**	**95.93**	**94.01**	**99.13**

and CUB 200. Notably, our approach achieves a state-of-the-art status on both datasets, showing a remarkable performance gap compared to the previous state-of-the-art model.

We attribute this divergence to the intrinsic nature of the datasets. The first two datasets, rooted in ILSVRC-12, exhibit substantial and intricate variations, posing challenges in facile comprehension. In contrast, CIFAR-FS and CUB 200 manifest relatively modest variations, rendering our proposed method notably robust. The observation that our method performs comparably well, and in some instances even surpasses, methodologies that update the weights of extensive models through relatively straightforward computations indicates the remarkable success of our proposed approach.

In summary, the outstanding results achieved across various datasets demonstrate the adaptability and effectiveness of our proposed method. The observed robustness, specifically on CIFAR-FS and CUB 200, establishes our approach as a promising solution in the field of few-shot classification, providing valuable insights for future research in this domain.

V. CONCLUSION

In this paper, we propose a novel approach to few-shot classification using pre-trained self-supervised learning models and feature analysis with class factors and environment factors. The factors transform matrix is estimated using a set of difference vectors and the PCA method, and the resulting transformation is used to extract low-dimensional features that are easy to classify from the pre-trained self-supervised model features. Despite the simplicity of this process compared to training a network model, our method improves on previous methods in terms of accuracy, demonstrating better performance even when compared to state-of-the-art models. The robustness observed, particularly on the CIFAR-FS and CUB 200 datasets, highlights the adaptability of our approach to different levels of dataset complexity. Although it did not quite achieve the highest performance on certain datasets, the results suggest that our method successfully addresses the challenges of few-shot classification without the need for complex meta-training or fine-tuning of large models. We are also considering using simple neural network training to find the optimal transform matrix as a substitute for PCA, and extensions to many-class classification problems are also a possibility to maximize the benefits of our proposed method.

ACKNOWLEDGMENT

This work was supported by the Human Resources Program in Energy Technology of the Korea Institute of Energy Technology Evaluation and Planning(KETEP) granted financial resource from the Ministry of Trade, Industry & Energy, Republic of Korea (No. 20204010600060). This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No.NRF-2020R1A2C1010020).

REFERENCES

[1] S. Gidaris, A. Bursuc, N. Komodakis, P. Pérez and M. Cord, "Boosting few-shot visual learning with self-supervision," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 8059-8068.

[2] Y. Tian, Y. Wang, D. Krishnam, J. Tenenbau and P. Isola, "Rethinking few-shot image classification: a good embedding is all you need?," in Computer Vision–ECCV 2020: 16th European Conference, 2020, pp. 266-282.

[3] D. Chen, Y. Chen, Y. Li, F. Mao, Y. He and H. Xue, "Self-supervised learning for few-shot image classification," in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 1745-1749.

[4] S. Hu, D. Li, J. Stühmer, M. Kim and T. Hospedales, "Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9068-9077.

[5] K. Lee and H. Park, "A New Similarity Measure Based on Intraclass Statistics for Biometrie Systems," ETRI journal, vol. 25(5), 2003, pp. 401-406.

[6] K. Lee and H. Park, "Probabilistic learning of similarity measures for tensor PCA," Pattern Recognition Letters, vol. 33(10), 2012, pp. 1364-1372.

[7] M. Cho and H. Park, "A feature analysis for dimension reduction based on a data generation model with class factors and environment factors," Computer Vision and Image Understanding, vol. 113(9), 2009, pp. 1005-1016.

[8] J. Snell, K. Swersky and R. Zemel, "Prototypical networks for few-shot learning," Advances in neural information processing systems 30, 2017

[9] W. Chen, Y. Liu, Z. Kira, Y. Wang and J. Huang, "A Closer Look at Few-shot Classification," in International Conference on Learning Representations, 2019

[10] Y. Lee and H. Park, "Effect of Probabilistic Similarity Measure on Metric-Based Few-Shot Classification," Applied Sciences, vol.11(22), 2021, 10977.

[11] Y. Chen, Z. Liu, H. Xu, T. Darrell and X. Wang, "Meta-baseline: Exploring simple meta-learning for few-shot learning," in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 9062-9071.

[12] K. Lee, S. Maji, A. Ravichandran, S. Soatto, "Meta-learning with differentiable convex optimization," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 10657-10665.

[13] Z. Chen, Y. Fu, K. Chen, Y. Jiang, "Image block augmentation for one-shot learning," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33(01), 2019, pp. 3379-3386.

[14] B. Hariharan and R. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," in Proceedings of the IEEE international conference on computer vision, 2017, pp. 3018-3027.

[15] A. Kolesnikov, X. Zhai and L. Beyer, "Revisiting self-supervised visual representation learning," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 1920-1929.

[16] H. Park and M. Cho, "Classification of bio-data with small data set using additive factor model and SVM," in AI 2006: Advances in Artificial Intelligence: 19th Australian Joint Conference on Artificial Intelligence, 2006, pp. 770-779.

[17] B. Moghaddam, T. Jebara and A. Pentland, "Bayesian modeling of facial similarity," Advances in neural information processing systems, 11, 1998.

[18] B. Moghaddam, A. Pentland, "Probabilistic visual learning for object representation," IEEE Transactions on pattern analysis and machine intelligence, vol.19(7), 1997, pp. 696-710.

[19] L. Wang, T.K. Tan, "A New Proposal for Face Feature Description," ISO/MPEG m5750, Noordwijkerhout, 2000.

[20] H. Kim, D. Kim, S. Bang and S. Lee, "Face recognition using the second-order mixture-of-eigenfaces method," Pattern Recognition, vol.37(2), 2004, pp. 337-349.

[21] T. Kim, H. Kim, W. Hwang and J. Kittler, "Independent component analysis in a local facial residue space for face recognition," Pattern Recognition, vol.37(9), 2004, pp. 1873-1885.

[22] Y. Hu, V. Gripon and S. Pateux, "Leveraging the feature distribution in transfer-based few-shot learning," in International Conference on Artificial Neural Networks, 2021, pp. 487-499.

[23] A. Singh and H. Jamali, "Transductive Decoupled Variational Inference for Few-Shot Classification," Transactions on Machine Learning Research, 2022.

[24] D. Guo, L. Tian, H. Zhao, M. Zhao and H. Zha, "Adaptive distribution calibration for few-shot learning with hierarchical optimal transport," Advances in Neural Information Processing Systems, 35, 2022, 6996-7010.

[25] M. Hiller, R. Ma, M. Harandi and T. Drummond, "Rethinking generalization in few-shot classification," Advances in Neural Information Processing Systems, 35, 2022, 3582-3595.

[26] Y, Hu, S. Pateux, V. Gripon, "Squeezing backbone feature distributions to the max for efficient few-shot learning," Algorithms, vol.15(5), 147, 2022.

[27] D. Shalama and S. Korman, "The self-optimal-transport feature transform," unpublished.

[28] Y. Hu, S. Pateux and V. Gripon, "Adaptive Dimension Reduction and Variational Inference for Transductive Few-Shot Classification," in International Conference on Artificial Intelligence and Statistics. PMLR, 2023, pp. 5899-5917.

[29] F. Hao, F. He, L. Liu, F. Wu, D. Tao and J. Cheng, "Class-Aware Patch Embedding Adaptation for Few-Shot Image Classification," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 18905-18915.

[30] H. Lin, G. Han, J. Ma, S. Huang, X. Lin and S. Chang, "Supervised masked knowledge distillation for few-shot transformers," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 19649-19659.

[31] L. Tian, J. Feng, X. Chai, W. Chen, L. Wang, X. Liu and B. Chen, "Prototypes-oriented Transductive Few-shot Learning with Conditional Transport," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 16317-16326.

[32] O. Vinyals, C. Blundell, T. Lillicrap and D. Wierstra, "Matching networks for one shot learning," Advances in neural information processing systems, 29, 2016.

[33] L. Bertinetto, J. Henriques, P. Torr and A. Vedaldi, "Meta-learning with differentiable closed-form solvers," in International Conference on Learning Representations (ICLR), 2019.

[34] M. Ren, E. Triantafillou, S. Ravi, J. Snell, K. Swersky, J. Tenenbaum and R. Zemel, "Meta-Learning for Semi-Supervised Few-Shot Classification," in International Conference on Learning Representations, 2018.

[35] X. He and Y. Peng, "Fine-grained visual-textual representation learning," IEEE Transactions on Circuits and Systems for Video Technology, vol.30(2), pp.520-531.