Self-Supervised Class-Cognizant Few-Shot Classification

2022 IEEE International Conference on Image Processing (ICIP)

Shirekar, Ojas Kishore; Jamali-Rad, Hadi

DOI
10.1109/ICIP46576.2022.9897431

Publication date
2022

Document Version
Final published version

Published in
Proceedings of the 2022 IEEE International Conference on Image Processing (ICIP)

2022 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-9620-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICIP46576.2022.9897431

SELF-SUPERVISED CLASS-COGNIZANT FEW-SHOT CLASSIFICATION

Ojas Kishore Shirekar[†], *Hadi Jamali-Rad*[†⋆]

[†] Faculty of EEMCS, Delft University of Technology (TU Delft), Delft, The Netherlands
[⋆] Shell Global Solutions International B.V., Amsterdam, The Netherlands

ABSTRACT

Unsupervised learning is argued to be the dark matter of human intelligence[1]. To build in this direction, this paper focuses on unsupervised learning from an abundance of unlabeled data followed by few-shot fine-tuning on a downstream classification task. To this aim, we extend a recent study on adopting contrastive learning for self-supervised pre-training by incorporating class-level cognizance through iterative clustering and re-ranking and by expanding the contrastive optimization loss to account for it. To our knowledge, our experimentation both in standard and cross-domain scenarios demonstrate that we set a new state-of-the-art (SoTA) in (5-way, 1 and 5-shot) settings of standard mini-ImageNet benchmark as well as the (5-way, 5 and 20-shot) settings of cross-domain CDFSL benchmark. Our code and experimentation can be found in our GitHub repository: https://github.com/ojss/c3lr.

***Index Terms*—** Few-shot classification, self-supervised learning, contrastive learning.

1. INTRODUCTION

Few-shot learning has received an upsurge of attention recently because it highlights a fundamental gap between human intelligence and data-hungry supervised deep learning methods. We humans can learn in a self-supervised fashion and/or with very little supervision. To tackle this challenge, few-shot classification is cast as the task of predicting class labels for a set of unlabeled data points (*query set*) given only a small set of labeled ones (*support set*). The query and support samples are typically drawn from the same distribution. Few-shot classification approaches are typically comprised of two sequential phases [1–4]: (i) *pre-training* on an abundant dataset (sometimes called "base"), followed by (ii) *fine-tuning* on an unseen dataset containing "novel" classes. Typically, the target classes in pre-training and fine-tuning phases are mutually exclusive. In this paper, we focus on self-supervised (also sometimes interchangeably called "unsupervised" in the literature) setting where we have no access to the class labels of the base dataset in the pre-training phase or their distribution.

The authors thank Delft University of Technology and Shell Global Solutions International B.V. for permission to publish this work.

[1] Yann LeCun's note; Meta AI blog post on self-supervised learning.

The art here is to devise a synthetic class label assignment technique and corresponding loss function in the pre-training phase to efficiently transfer the learning to the fine-tuning phase. To this aim, studies have proposed two different approaches. The first approach follows a *meta-learning* strategy to create (synthetic) "tasks" similar to the the downstream episodic training in the fine-tuning phase [5–7]. The second one follows some sort of *transfer learning* approach, where a representation learning step in the pre-training phase is followed by episodic fine-tuning [1, 8, 9]. In the latter case, typically a feature extractor (encoder) is trained using metric learning to capture the global structure of the unlabeled data. Next, a simple predictor (typically a linear layer) is adopted in conjunction with the extractor for quick adaptation to the novel classes in the fine-tuning phase. The better the feature extractor captures the global structure of the unlabeled data, the less the predictor requires training samples and the faster it adapts itself to the unseen classes in the fine-tuning phase.

Recent studies [1, 9, 10] demonstrate that the second approach based on transfer learning outperforms meta-learning based methods in cross-domain settings, where the training and novel classes come from totally different distributions. Their results also show that a properly-devised transfer learning based unsupervised approach comes pretty close to the performance of a fully supervised counterpart [1, 3], something that we will also confirm through experimentation. Most recently, a new state-of-the-art (SoTA) in self-supervised few shot classification has been set by extending the prototypical networks (ProtoNets) [11] using a *contrastive* loss [2]. This approach (called ProtoTransfer [1]) constructs a contrastive metric embedding that clusters unlabeled prototypical samples and their augmentations. Inspired by this idea, we propose class-cognizant contrastive learning (C^3LR, Algorithm 1) to further extend it to incorporate class-level insights from the global structure of data. This is done via an unsupervised iterative re-ranking and clustering step resulting in clusters of unlabeled embeddings followed by a modified contrastive loss now containing a term that specifically promotes this class-level global structure. Our experimentation demonstrates that C^3LR outperforms its predecessor ProtoTransfer in (5-way, 1 and 5-shot) settings of Ominglot [12] and mini-Imagenet [13] benchmarks by about 1% and $2\%+$, respectively. The performance improvement goes up to 4.5% in the cross-domain

setting of the CDFSL benchmark [14]. As a result, to our best knowledge, C^{3}LR sets a new SoTA for most challenging settings of mini-ImageNet and CDFSL benchmarks.

2. CLASS-COGNIZANT CONTRASTIVE LEARNING (C^{3}LR)

In this section, we first describe our problem formulation. We then discuss the two phases of the proposed approach: self-supervised pre-training and few-shot supervised fine-tuning. The mechanics of the proposed approach and a sketch of the training procedure is shown in Figure 1.

2.1. Preliminaries

Let us denote the training data of size M as $\mathcal{D}_{\text{tr}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{M}$ with $(\boldsymbol{x}_i, y_i)$ representing an image $\boldsymbol{x}_i$ and its class label y_i. In the pre-training phase, we take L random samples from $\mathcal{D}_{\text{tr}}$ and augment each sample Q times by drawing augmentation functions $\psi^q(.), \forall q \in [Q]$ from the set $\mathcal{A}$. This results in a batch of size $B = (Q+1)L$ total samples. Note that the data labels are unknown in the pre-training phase. In the fine-tuning phase, we deal with the so-called episodic training on a set of tasks $\mathcal{T}$ containing N classes each with K samples per task drawn from the test dataset $\mathcal{D}_{\text{tst}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{M'}$ of size M'. From now on, we refer to this task construct as (N-way, K-shot) denoted by (N, K). An episode consists of a labeled support set, $\mathcal{S}$, from which the model learns and an unlabeled query set, $\mathcal{Q}$, on which the model predicts. Note that both $\mathcal{S}$ and $\mathcal{Q}$ contain a set of tasks of the form (N, K).

2.2. Self-Supervised Pre-Training

The fact that we do not have access to class labels calls for a self-supervised pre-training stage. As discussed earlier, we build upon the idea of employing contrastive learning for prototypical transfer learning following the footsteps of [1]. The **high-level idea** here is to not only enforce the latent embeddings of augmented images come close to that of the source image in the embedding space (the classical contrastive setting), but also enforce embeddings of the images belonging to each cluster (and their augmentations) come closer to each other, for which a preceding unsupervised cluster formation step is required. This can help enforce similar classes into separate clusters, which will in turn be used as additional information in a modified two-term contrastive loss in Algorithm 1. Let us walk you through the process in further details.

Algorithm 1 starts with **batch generation** (lines 2 to 7): each mini-batch consists of L random samples $\{\boldsymbol{x}_i\}_{i=1}^{L}$ from $\mathcal{D}_{\text{tr}}$, where $\boldsymbol{x}_i$ is treated as a 1-shot support sample for which we create Q randomly augmented versions $\tilde{\boldsymbol{x}}_{i,q}$ as query samples (line 5). This leads to a batch size of $B = (Q+1)L$. Then embeddings are generated by passing the samples through an encoder f_ϕ network. This is where the first major modification to ProtoTransfer [1] comes into play. Before the contrastive loss comes into action, we apply **re-ranking and clustering**

Algorithm 1: Class-Cognizant Contrastive Learning (C^{3}LR)

Require: L, Q, f_ϕ, $\mathcal{A}$, α, $d[\cdot,\cdot]$

1 **while** *not done* **do**
2 Sample minibatch $\{\boldsymbol{x}_i\}_{i=1}^{L}$
3 **forall** $i \in \{1, \ldots, L\}$ **do**
4 **forall** $q \in \{1, \ldots, Q\}$ **do**
5 $\tilde{\boldsymbol{x}}_{i,q} = \psi^q(\boldsymbol{x}_i)$; $\psi^q \sim \mathcal{A}$.
6 **end**
7 **end**
8 $\mathbf{R} = \texttt{ReRank}\Big(\Big[f_\phi\left(\{\boldsymbol{x}_i\}_{i=1}^{L}\right), f_\phi\left(\{\tilde{\boldsymbol{x}}_{i,\,q}\}_{i=1,q=1}^{L,Q}\right)\Big]\Big)$
9 $\mathcal{C} = \{\boldsymbol{C}_1, \boldsymbol{C}_2, \ldots, \boldsymbol{C}_P\} \leftarrow \texttt{HDBSCAN}(\mathbf{R})$
10 $\mathcal{M} = \{\mathbf{m}_p\}_{p=1}^{P}; \;\; \mathbf{m}_p = \frac{\sum_{x_j \in \boldsymbol{C}_p} x_j}{|\boldsymbol{C}_p|}$
11 **let** $r(i,q,p) = -\log \frac{\exp(-d[f_\phi(\tilde{\boldsymbol{x}}_{i,q}), \mathbf{m}_\mathbf{p}])}{\sum_{p=1}^{P} \exp(-d[f_\phi(\tilde{\boldsymbol{x}}_{i,q}), \mathbf{m}_\mathbf{p}])}$
12 **let** $\ell(i,q) = -\log \frac{\exp(-d[f_\phi(\tilde{\boldsymbol{x}}_{i,q}), f_\phi(\boldsymbol{x}_i)])}{\sum_{k=1}^{L} \exp(-d[f_\phi(\tilde{\boldsymbol{x}}_{i,q}), f_\phi(\boldsymbol{x}_k)])}$
13 $\mathcal{L}_1 = \frac{1}{LQ} \sum_{p=1}^{P} \sum_{i=1}^{L} \sum_{q=1}^{Q} r(i,q,p)$
14 $\mathcal{L}_2 = \frac{1}{LQ} \sum_{i=1}^{L} \sum_{q=1}^{Q} \ell(i,q)$
15 $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$
16 $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}$
17 **end**

(lines 8 to 10) to discover class-level global structure of data and enforce similar classes into separate clusters in the embedding space. Note that this step remains to be unsupervised in that the class labels are not required. The re-ranking step (line 8) makes use of the k-reciprocal nearest neighbors as the distance metric between latent embeddings [15], which has been shown to outperform the Euclidean distance [3] when used for subsequent clustering. HDBSCAN clustering [16] is then applied on the re-ranked embeddings $\mathbf{R}$ and returns a set of clusters populated in $\mathcal{C}$. HDBSCAN is versatile enough to discover and create required number of clusters P. With clusters at hand, we are now in a position to extend the standard loss proposed in [1] to contain a **class-cognizant term** (in lines 11 and 13), with lines 12 and 14 reflecting on the classical contrastive loss of ProtoTransfer [1]. This new loss term $\mathcal{L}_1$ enables a progressive improvement in class-level cluster formation and in turn learning similar representations for cluster members, while $\mathcal{L}_2$ encourages clustering of the embeddings of the augmented query samples $\{f_\phi(\tilde{\boldsymbol{x}}_{i,q})\}$ around their prototypes $\{f_\phi(\boldsymbol{x}_i)\}$. Here, both terms use an Euclidean distance metric in the embedding space denoted by $d[\cdot,\cdot]$. Finally, the new loss $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$ is optimized with mini-batch stochastic gradient decent with respect to the parameters ϕ of the encoder networks f_ϕ.

2.3. Supervised Fine-Tuning

The pre-trained encoder f_ϕ will be used for the downstream few-shot classification task. To this aim, following [1, 11], we

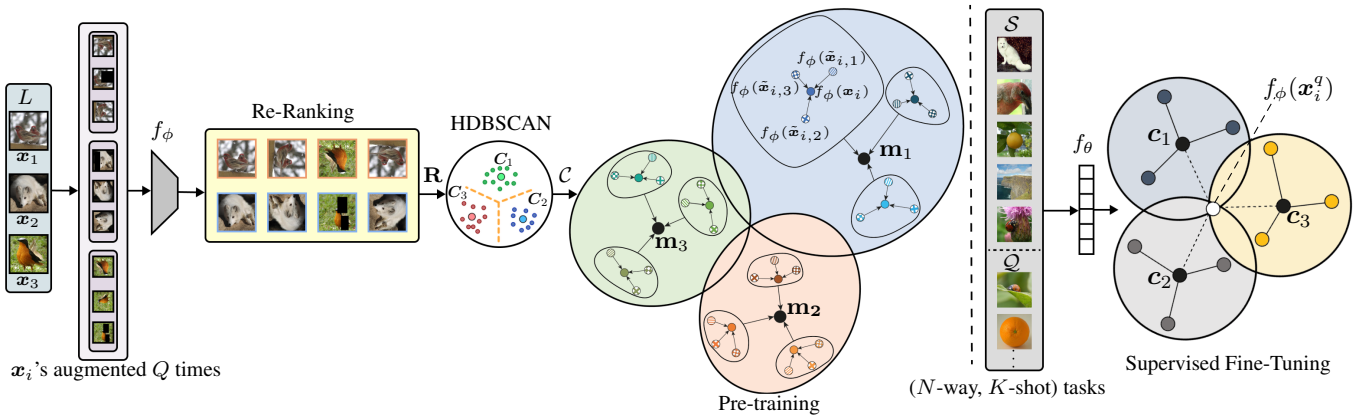

Fig. 1: C^{3}LR schematic view and training procedure. In the figure, $\boldsymbol{x}_i^q$ is an image sampled from the query set $\mathcal{Q}$.

concatenate f_ϕ with a single-layer nearest-neighbor classifier f_θ (resulting in a similar architecture as in ProtoNet [11]) and fine-tune this last layer. In this phase, we first calculate the class prototypes $\boldsymbol{c}_n$ (embeddings) for class n using the encoder f_ϕ on the support set $\mathcal{S}_n$:

$$\boldsymbol{c}_n = \frac{1}{|\mathcal{S}_n|} \sum_{(\boldsymbol{x}_i, y_i) \in \mathcal{S}_n} f_\phi(\boldsymbol{x}_i).$$

These prototypes are then used to initialize the classifier f_θ following [1].

3. EXPERIMENTATION

In this section, we first discuss our experimental setup; we then present our numerical results.

3.1. Experimental Setup

Datasets. We conduct several in-domain experiments to benchmark C^{3}LR. For this purpose, we make use of commonly adopted datasets Omniglot [12] and mini-Imagenet [13] to compare against unsupervised few-shot learning approaches. Omniglot contains 1623 different handwritten characters borrowed from 50 unique alphabets out of which we use 1028 characters for training, 172 for validation and 423 for testing. We resize the grayscale images to 28×28 pixels. Mini-ImageNet contains 100 classes with 600 samples in each class amounting to a total of $60,000$ images that we resize to 84×84 pixels. Out of the 100 classes, we use 64 classes for training, 16 for validation and 20 for testing. For both datasets, the settings are the most commonly adopted ones in literature [1, 3, 7, 13]. The augmentations (in $\mathcal{A}$) used for the experimentations follow [1]. We also compare our method on a more challenging cross-domain few-shot learning (CDFSL) benchmark [14]. This benchmark consists of four datasets with increasing similarities to mini-ImageNet. In that order, we have grayscale chest X-ray images from ChestX [17], dermatological skin lesion images from ISIC2018 [18], satellite aerial images from EuroSAT [19], and crop disease images from CropDiseases [20]. We also use Caltech-UCSD Birds (CUB) dataset [21] for further analysis of cross-domain performance. CUB is composed of $11,788$ images from 200 unique bird species. We use 100 images for training, 50 for validation and 50 for test.

Training. The Conv4 model [13] is pre-trained on the respective training splits of the datasets, with an initial learning rate of 0.001, multiplied by 0.5 every $25,000$ steps via the Adam optimizer [22]. Based on the derivations in [11] and similar usage in [1], we initialize the classification layer f_θ with weights set to $\mathbf{W}_n = 2\boldsymbol{c}_n$ and biases set to $b_n = -\|\boldsymbol{c}_n\|^2$. For validation, we create 15 (N-way, K-shot) tasks using the validation split from which the corresponding validation accuracy and loss are calculated. Experiments involving CDFSL benchmark follow [1, 14], where we pre-train a ResNet10 encoder using C^{3}LR on mini-ImageNet images of size 224×224 for 400 epochs with the Adam optimizer and a constant learning rate of 0.001.

Evaluation scenarios and baseline. Our testing scheme uses 600 test episodes on which the pre-trained encoder (using C^{3}LR) is fine-tuned and tested. All our results indicate 95% confidence intervals over 3 runs each with 600 test episodes. The standard deviation values are thus calculated according to the 3 runs to provide more solid measures for comparison. For our in-domain benchmarks, we test on (5-way, 1-shot) and (5-way, 5-shot) classification tasks. While our cross-domain testing is done using (5-way, 5-shot) and (5-way, 20-shot) classification tasks. We compare our performance with a suit of recent self-supervised few-shot baselines such as ProtoTransfer [1], UFLST [3], LASIUM [23] and CACTUS [6], to name a few. Furthermore, we also compare with a set of supervised approaches (such as MAML [5], ProtoNet [11] , etc.) the best performing of which are obviously expected to outperform ours as well as other self-supervised methodologies.

3.2. Performance Evaluation

In-domain evaluation. Table 1 summarizes our performance evaluation results on Omniglot and mini-ImageNet datasets for (N-way, K-shot) scenarios with $N = 5$ and $K = 1, 5$. The top

Table 3: Accuracy (%± std.) of (N-way, K-shot) classification on the CDFSL benchmark. Style: **best** and second best.

Method(N, K)	(5,5)	(5,20)	(5,5)	(5,20)	(5,5)	(5,20)	(5,5)	(5,20)
	ChestX		**ISIC**		**EuroSAT**		**CropDiseases**	
UMTRA-ProtoNet [1]	24.94 ± 0.43	28.04 ± 0.44	39.21 ± 0.53	44.62 ± 0.49	74.91 ± 0.72	80.42 ± 0.66	79.81 ± 0.65	86.84 ± 0.50
UMTRA-ProtoTune [1]	25.00 ± 0.43	30.41 ± 0.44	38.47 ± 0.55	51.60 ± 0.54	68.11 ± 0.70	81.56 ± 0.54	82.67 ± 0.60	92.04 ± 0.43
ProtoTransfer [1]	**26.71** ± 0.46	**33.82** ± 0.48	45.19 ± 0.56	59.07 ± 0.55	75.62 ± 0.67	86.80 ± 0.42	86.53 ± 0.56	95.06 ± 0.32
C^{3}LR (**ours**)	26.00 ± 0.41	33.39 ± 0.47	**45.93** ± 0.54	**59.95** ± 0.53	**80.32** ± 0.65	**88.09** ± 0.45	**87.90** ± 0.55	**95.38** ± 0.31
ProtoNet [14] (sup.)	24.05 ± 1.01	28.21 ± 1.15	39.57 ± 0.57	49.50 ± 0.55	73.29 ± 0.71	82.27 ± 0.57	79.72 ± 0.67	88.15 ± 0.51
Pre+Mean-Cent. [14] (sup.)	26.31 ± 0.42	30.41 ± 0.46	47.16 ± 0.54	56.40 ± 0.53	82.21 ± 0.49	87.62 ± 0.34	87.61 ± 0.47	93.87 ± 0.68
Pre+Linear [14] (sup.)	25.97 ± 0.41	31.32 ± 0.45	48.11 ± 0.64	59.31 ± 0.48	79.08 ± 0.61	87.64 ± 0.47	89.25 ± 0.51	95.51 ± 0.31

Table 1: Accuracy (%± std.) for (N-way, K-shot) classification tasks. Style: **best** and second best.

	Omniglot		**mini-ImageNet**	
Method(N, K)	(5,1)	(5,5)	(5,1)	(5,5)
CACTUs-MAML [6]	68.84 ± 0.80	87.78 ± 0.50	39.90 ± 0.74	53.97 ± 0.70
CACTUs-ProtoNet [6]	68.12 ± 0.84	83.58 ± 0.61	39.18 ± 0.71	53.36 ± 0.70
UMTRA [7]	83.80	95.43	39.93	50.73
AAL-ProtoNet [24]	84.66 ± 0.70	89.14 ± 0.27	37.67 ± 0.39	40.29 ± 0.68
AAL-MAML++ [24]	88.40 ± 0.75	97.96 ± 0.32	34.57 ± 0.74	49.18± 0.47
UFLST [3]	**97.03**	**99.19**	33.77 ± 0.70	45.03 ± 0.73
ULDA-ProtoNet [25]	-	-	40.63 ± 0.61	55.41 ± 0.57
ULDA-MetaOptNet [25]	-	-	40.71 ± 0.62	54.49 ± 0.58
U-SoSN+ ArL [26]	-	-	41.13 ± 0.84	55.39 ± 0.79
LASIUM [23]	83.26 ± 0.55	95.29 ± 0.22	40.19 ± 0.58	54.56 ± 0.55
ProtoTransfer ($L = 50$) [1]	88.00 ± 0.64	96.48 ± 0.26	45.67 ± 0.79	62.99 ± 0.75
ProtoTransfer ($L = 200$)	88.37 ± 0.74	96.54 ± 0.41	44.17 ± 1.08	61.07 ± 0.82
C^{3}LR (**ours**)	89.30 ± 0.64	97.38 ± 0.23	**47.92** ± 1.2	**64.81** ± 1.15
MAML [5] (supervised)	94.46 ± 0.35	98.83 ± 0.12	46.81 ± 0.77	62.13± 0.72
ProtoNet [11] (supervised)	97.70± 0.29	99.28 ± 0.10	46.44± 0.78	66.33± 0.68
MMC [27] (supervised)	97.68± 0.07	-	50.41 ± 0.31	64.39 ± 0.24
FEAT [4] (supervised)	-	-	55.15	71.61
Pre+Linear [1] (supervised)	94.30 ± 0.43	99.08 ± 0.10	43.87 ± 0.69	63.01 ± 0.71

Table 2: Accuracy (%± std.) for (N-way, K-shot) classification on mini-ImageNet with pre-training on CUB.

Training	**Testing**	(5,1)	(5,5)
ProtoTransfer ($L = 50$) [1]	ProtoTune [1]	35.37 ± 0.63	52.38 ± 0.66
ProtoTransfer ($L = 200$)	ProtoTune	34.67 ± 0.84	51.45 ± 0.72
C^{3}LR (**ours**)	ProtoTune	**39.61** ± 1.11	**55.53** ± 1.42

section compares the performance of the proposed approach (C^{3}LR) with the most recent relevant self-supervised competitors. As can be seen, for Omniglot, we outperform ProtoTransfer [1] (which we build on) by about 1% in both $K = 1, 5$ shot scenarios. We score the second overall best in (5-way, 1-shot) falling behind UFLST [3]. For the mini-ImageNet benchmark, to our knowledge, we set a new SoTA outperforming ProtoTransfer by $2\%+$. Interestingly, our performance beats some of the supervised baselines (bottom section of the table) adopting similar encoder architecture Conv4 for mini-ImageNet and comes close to $K = 5$-shot performances on Omniglot. Obviously, the SoTA supervised few-shot learning approaches have the advantage of having access to the all the labels, as such due to the supervision signal, are expected to outperform the unsupervised approaches like ours.

Cross-domain evaluation. So far we have demonstrated that the proposed approach excels for in-domain scenarios. The next step is to assess the performance under more challenging cross-domain scenarios (Table 2 and Table 3) where we pre-train on a certain dataset in an unsupervised fashion, then fine-tune and test on a different dataset. Table 2 illustrates the results of a Conv4 encoder trained on CUB and tested on tasks derived from mini-ImageNet. Here again C^{3}LR shows a clear improvement of $3\%+$ compared to ProtoTransfer (with pre-training sample sizes $L = 50, 200$). The important message here is that the proposed approach enhances ProtoTransfer in generalizing to truly unseen data. To further investigate the performance on cross-domain scenarios, we next focus on CDFSL benchmark [14] containing several datasets. Here, we pre-train on mini-ImageNet and fine-tune and test on ChestX [17], ISIC2018 [18], EuroSAT [19], and CropDiseases [20]. We compare the performance against ProtoTransfer and two of its variants with UMTRA [7] as pre-training strategy (all proposed in [1]). We also compare with a couple of closely related supervised approaches from [14], for the sake of reference. As can be seen, except for ChestX where we marginally come short of ProtoTransfer, for the other three datasets we outperform the second best competitor (ProtoTransfer) by about $0.5\%+$ to $4.5\%+$ with the most significant improvement in the case of EuroSAT. Interestingly, once again the performance of C^{3}LR is not far off that of the related supervised approaches (bottom of the table) even sometimes outperforming the supervised approaches especially in (5-way, 20-shot) scenarios.

4. CONCLUDING REMARKS

Inspired by the idea of using contrastive learning for unsupervised few-shot classification, we build upon the recently proposed idea of ProtoTransfer [1] by incorporating class cognizance through: (i) an unsupervised iterative re-ranking and clustering step, followed by (ii) an adjusted optimization loss formulation. We demonstrate that our proposed approach (C^{3}LR) offers considerable performance improvement above its predecessor ProtoTransfer in both in/cross-domain few-shot classification scenarios setting a new SoTA in mini-ImageNet and CDFSL benchmarks.

5. REFERENCES

[1] Carlos Medina, Arnout Devos, and Matthias Grossglauser, "Self-supervised prototypical transfer learning for few-shot classification," *arXiv preprint arXiv:2006.11325*, 2020.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[3] Zilong Ji, Xiaolong Zou, Tiejun Huang, and Si Wu, "Unsupervised few-shot learning via self-supervised training," *arXiv preprint arXiv:1912.12178*, 2019.

[4] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha, "Few-shot learning via embedding adaptation with set-to-set functions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8808–8817.

[5] Chelsea Finn, Pieter Abbeel, and Sergey Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.

[6] Kyle Hsu, Sergey Levine, and Chelsea Finn, "Unsupervised learning via meta-learning," *arXiv preprint arXiv:1810.02334*, 2018.

[7] Siavash Khodadadeh, Ladislau Boloni, and Mubarak Shah, "Unsupervised meta-learning for few-shot image classification," *Advances in neural information processing systems*, vol. 32, 2019.

[8] 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 *European Conference on Computer Vision*. Springer, 2020, pp. 266–282.

[9] Guneet S Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto, "A baseline for few-shot image classification," *arXiv preprint arXiv:1909.02729*, 2019.

[10] Da Chen, Yuefeng Chen, Yuhong Li, Feng Mao, Yuan He, and Hui Xue, "Self-supervised learning for few-shot image classification," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 1745–1749.

[11] Jake Snell, Kevin Swersky, and Richard Zemel, "Prototypical networks for few-shot learning," *Advances in neural information processing systems*, vol. 30, 2017.

[12] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 12 2015.

[13] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al., "Matching networks for one shot learning," *Advances in neural information processing systems*, vol. 29, 2016.

[14] Yunhui Guo, Noel CF Codella, Leonid Karlinsky, John R Smith, Tajana Rosing, and Rogerio Feris, "A New Benchmark for Evaluation of Cross-Domain Few-Shot Learning," *arXiv preprint arXiv:1912.07200*, 2019.

[15] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li, "Re-ranking person re-identification with k-reciprocal encoding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1318–1327.

[16] Leland McInnes, John Healy, and Steve Astels, "hdbscan: Hierarchical density based clustering.," *J. Open Source Softw.*, vol. 2, no. 11, pp. 205, 2017.

[17] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2097–2106.

[18] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al., "Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic)," *arXiv preprint arXiv:1902.03368*, 2019.

[19] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth, "Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification," 2017.

[20] Sharada P Mohanty, David P Hughes, and Marcel Salathé, "Using Deep Learning for Image-Based Plant Disease Detection," *Frontiers in Plant Science*, vol. 7, pp. 1419, 2016.

[21] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.

[22] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] Siavash Khodadadeh, Sharare Zehtabian, Saeed Vahidian, Weijia Wang, Bill Lin, and Ladislau Bölöni, "Unsupervised meta-learning through latent-space interpolation in generative models," *arXiv preprint arXiv:2006.10236*, 2020.

[24] Antreas Antoniou and Amos Storkey, "Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation," *arXiv preprint arXiv:1902.09884*, 2019.

[25] Tiexin Qin, Wenbin Li, Yinghuan Shi, and Yang Gao, "Diversity helps: Unsupervised few-shot learning via distribution shift-based data augmentation," *arXiv preprint arXiv:2004.05805*, 2020.

[26] Hongguang Zhang, Piotr Koniusz, Songlei Jian, Hongdong Li, and Philip HS Torr, "Rethinking class relations: Absolute-relative supervised and unsupervised few-shot learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9432–9441.

[27] 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," *arXiv preprint arXiv:1803.00676*, 2018.