Towards Label-Free Few-Shot Learning: How Far Can We Go?

Aditya Bharti[1], Vineeth N. B.[2], and C.V. Jawahar[1]

[1] International Institute of Information Technology Hyderabad
[2] Indian Institute of Technology Hyderabad

Abstract. Few-shot learners aim to recognize new categories given only a small number of training samples. The core challenge is to avoid overfitting to the limited data while ensuring good generalization to novel classes. Existing literature makes use of vast amounts of annotated data by simply shifting the label requirement from novel classes to base classes. Since data annotation is time-consuming and costly, reducing the label requirement even further is an important goal. To that end, our paper presents a more challenging few-shot setting with almost no class label access. By leveraging self-supervision to learn image representations and similarity for classification at test time, we achieve competitive baselines while using **almost zero** (0-5) class labels. Compared to existing state-of-the-art approaches which use 60,000 labels, this is a **four orders of magnitude (10,000 times) difference**. This work is a step towards developing few-shot learning methods that do not depend on annotated data. Our code is publicly released at https://github.com/adbugger/FewShot.[3]

Keywords: Few Shot · Self-supervised · Deep Learning

1 Introduction

Few-shot learners [39,33,10] aim to learn novel categories from a small number of examples. Since getting annotated data is extremely difficult for many natural and man-made visual classes [20], such systems are of immense importance as they alleviate the need for labelled data.

Few-shot learning literature is extremely diverse [41] with multiple classes of approaches. Meta-learning [10,30,29] is a popular class of methods which use experience from multiple base tasks to learn a base learner which can quickly adapt to novel classes from few examples. There has been immense progress using the meta-learning frameworks [11,43,16,16,27,1]. While extremely popular, such approaches are computationally expensive, require that the base tasks be related to the final task, and need many training class labels for the base tasks. Other approaches focus on combining supervised and unsupervised pipelines [12,34,4] and others alleviate the data requirement by generating new labelled data using

[3] This work was supported by the IMPRINT program.

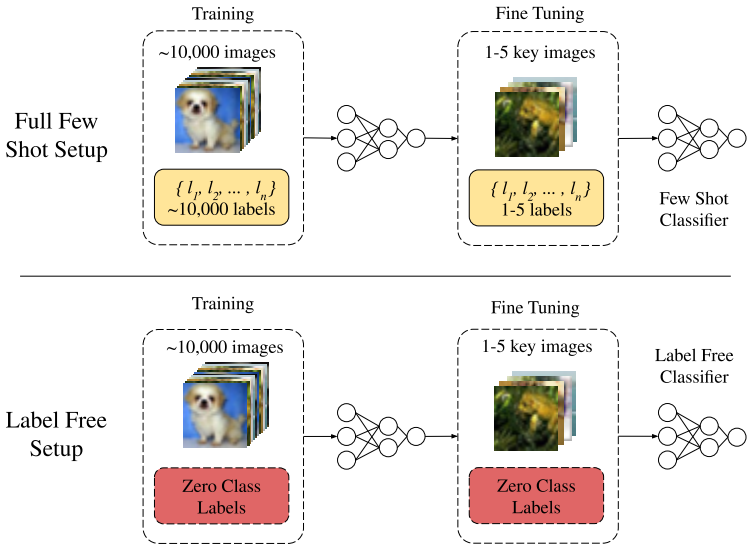

Fig. 1. Label-free Few-shot Classification: Proposed setting *(Best viewed in color)*

hallucinations [17]. Recent methods [40,5] have also established strong baselines with computationally extremely simple training pipelines and classifiers. However, these methods either do not address the label requirement, or cannot be easily extended to new network architectures. This work is a step towards developing few-shot learning methods that do not depend on annotated data.

Recent work in contrastive learning [21,19,5] has shown that it is possible to learn useful visual representations without class labels by learning image similarity over multiple augmented views of the same data, paired with a suitable training strategy and a loss function. We leverage SimCLR [5] and MoCo [19] to develop training methods with restricted label access. Since image similarity is an effective pre-training task for few-shot [22], we perform image classification using image similarity as shown in Figure 2. We perform test time classification by choosing the key image most similar to the input to be classified. The network is thus completely unaware of any class label information.

Our key contributions are as follows:

- A new challenging label-free few-shot learning setup.
- An easy to adapt, computationally and conceptually simple, end to end label free pipeline.
- Competitive performance using **almost zero** class labels. Compared to the approximately $60,000$ class labels used by existing state-of-the-art, this is a **four orders of magnitude** improvement.
- We examine classification quality and the impact of limited label information in our ablations.

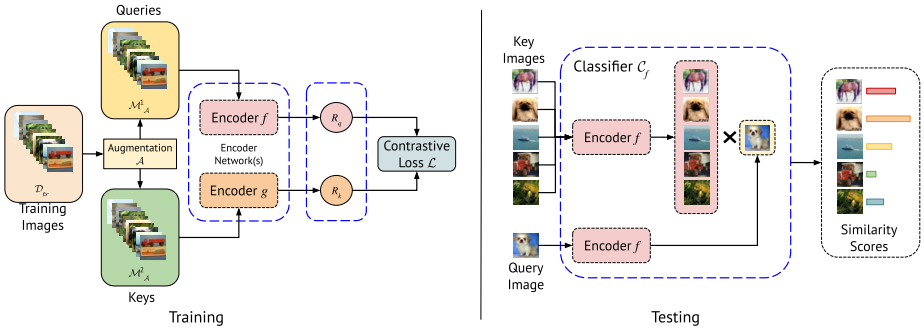

Fig. 2. General overview of our pipeline. **Left**: Self-supervised training to learn contrastive representations without labels. See Algorithm 1 for further details. **Right**: Image classification using image similarity for few-shot classification without using labels. Images are encoded using the model learned in the training phase. Further details in Algorithm 2. *(Best viewed in color)*

2 Related Work

There is a great diversity of few-shot learning literature [41]. In this section we discuss motivating works from related fields.

2.1 Related Perspectives

Metric learning methods "learn to compare". By learning image similarity, a model can use similarity to label instances of novel classes by comparing with known examples. It is also an effective pre-text task for few-shot learning [22]. These models learn by conditioning predictions on distance metrics such as cosine similarity [39], Euclidean distances [33], network based [36], ridge regression [3], convex optimization based [25], or graph neural networks [32]. Regularization techniques such as manifold mixup [26], combined with supervised pipelines, also improve accuracies.

Self supervised methods remove the need for annotated data by using a supervisory signal from the data itself. A number of pretext tasks such as predicting image coloration [24,44], predicting image patch positions [28,9], and predicting image rotations [15] are used in the literature. Combining self-supervision with supervised approaches [12,34,4] has also resulted in improved accuracies over few-shot tasks. Finally, [5,19,37,21] learn contrastive representations by applying simple transforms on input images and predicting image similarity. By learning to predict image similarity in the presence of distortions, the network can effectively distill information, making it suitable for quick adaptation on novel classes. We leverage two recent contrastive approaches in our work: SimCLR [5] and MoCo [19].

3 Approach: Few-Shot Learning with almost No Labels

Since learning image similarity is useful for few-shot tasks [22], we focus on learning contrastive representations during our training phase. This allows us to ignore the labels completely, unlike [22]. Following a contrastive learning approach, we first apply two different data augmentations to an input image, generating two augmented images. The task for our neural network $f(\cdot)$ is to learn image representations such that the encoding of two augmented images generated from the same input are as similar as possible. Algorithm 1 presents a detailed description of one training epoch, and Figure 2 presents a visual overview.

Given an input minibatch $\mathcal{M}$ a stochastic augmentation module $\mathcal{A}$ generates two minibatches, one of the query images $\mathcal{M}_{\mathcal{A}}^{q}$, and the other of the key images $\mathcal{M}_{\mathcal{A}}^{k}$ by performing two different image transforms. For a given query image, q the key image generated from the same input is denoted k_+, and k_- otherwise. Pairs of query and key generated from the same input (q, k_+) are denoted positive and negative (q, k_-) otherwise.

Encoder networks $f(\cdot)$ and $g(\cdot)$ are used to learn representations of key and query images respectively. Note that depending on the setting, these networks may be the same. Network $f(\cdot)$ is used for downstream test time tasks. After computing the encoded representations R_k of the key, and R_q of the query, the networks are trained to maximize the representation similarity for positive pairs, and minimize for negative pairs. This is achieved by minimizing the following contrastive loss in Eqn 1

$$\mathcal{L}(R_q, R_{k^+}, \{R_{k^-}\}) = -\log \frac{\exp s(R_q, R_{k^+})/\tau}{\exp s(R_q, R_{k^+})/\tau + \sum\limits_{R_{k^-}} \exp s(R_q, R_{k^-})/\tau} \tag{1}$$

where τ is a temperature hyperparameter, and $s(\cdot, \cdot)$ is a similarity function.

SimCLR Base In this setting, we treat augmented minibatches of key and query images on an equal footing with no distinction. Starting from an input minibatch of N images, there are $2N$ positive pairs and $2N(2N-2)$ negative pairs. The same network $f(\cdot)$ is used to embed both keys and queries. A cosine similarity function is used in the contrastive loss (Eqn 1), $s(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y}/|\mathbf{x}||\mathbf{y}|$. This setup is referred to as OURS_S in our results.

MoCo Base This setting decouples the number of negative samples from the batch size. Once the key and query images have been generated from the input, the few-shot task is formulated as a dictionary lookup problem. The dictionary consists of *key* images, and the unknown image to be looked up is the *query*. The keys are encoded using a momentum encoder, which maintains the set of positive and negative samples per query. The query (non-momentum) encoder is used for downstream few-shot tasks. This setting uses a dot product as the similarity function for contrastive loss $s(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T\mathbf{y}$ and is referred to as OURS_M in our results.

Algorithm 1: Overall Training Methodology

Input: Augmentation Module $\mathcal{A}(\cdot)$
Input: Encoders $f(\cdot)$ $g(\cdot)$
Input: Constrastive Loss Module $\mathcal{L}(q, k^+, \{k^-\})$
Data: Training dataset $\mathcal{D}_{tr}$
Result: Trained network $f(\cdot)$

for *minibatch* $\mathcal{M}$ *in* $\mathcal{D}_{tr}$ **do**
  $\mathcal{M}^q_{\mathcal{A}}, \mathcal{M}^k_{\mathcal{A}} = \mathcal{A}(\mathcal{M})$; `// get augmented minibatches`
  $\{R_q\} = f(\mathcal{M}^q_{\mathcal{A}})$; `// encode query representations`
  $\{R_k\} = g(\mathcal{M}^k_{\mathcal{A}})$; `// encode key representations`
  for *query* R_q *in* $\{R_q\}$ **do**
    R_{k^+} = `ChoosePositive(`R_q, $\{R_k\}$`)` ; `// positive key image`
    $\{R_{k^-}\}$ = `ChooseNegative(`R_q, $\{R_k\}$`)` ; `// negative key images`
    $\mathcal{L}(R_q, R_{k^+}, \{R_{k^-}\})$; `// minimize contrastive loss`
    `UpdateParams(`f, g`)` ; `// update network parameters`
  end
end
return f

3.1 Testing Framework

We present our general testing framework and provide details of the specific test time classifiers used for our experiments. The testing phase consists of multiple few-shot tasks, following accepted practice [40]. Each C-way K-shot task consists of K key images, and Q query images from C novel classes each. Using the $C * K$ key images, the trained network must classify the $C * Q$ query images.

Given the set of key images $\{k\}$, a query image q to be classified, and the trained network $f(\cdot)$ from the training phase, a classifier $\mathcal{C}_f$ matches q with its most similar key image k_j. The classification is deemed correct if q and k_j have the same label, as determined by a separate verifier since the classifier does not have access to labels. See Algorithm 2 for a concise representation of our testing framework.

Inspired by [6,40], we study the use of two different test time classifiers: the 1-Nearest Neighbor classifier (1NN) from SimpleShot [40] and a soft cosine attention kernel (Attn) adapted from Matching Networks [39].

The 1NN classifier chooses the key image which minimizes the Euclidean distance between the key and the query image under consideration.

$$\mathcal{C}_f(q, \{k\}) = \arg\min_j |f(q) - f(k_j)|^2 \tag{2}$$

The Attn classifier chooses the key image corresponding to each query using an attention mechanism that provides a softmax over the cosine similarities. Unlike Matching Networks [39], we take an arg max instead of a weighted average over the labels of the key image set since the classifier has no access to the probability

Algorithm 2: Test Phase: N-way, K-shot Task

Input: Trained Encoder f
Input: Classifier $\mathcal{C}_f$
Input: Similarity Function $s(\mathbf{x}, \mathbf{y})$
Data: $N \times Q$ query images: $\{(q_i, y_{q_i})\}$
Data: $N \times K$ test images: $\{(k_i, y_{k_i})\}$
Result: Accuracy on task
correct $\leftarrow 0$;
foreach *query image* q_i **do**
  `// return index of most similar key since classifier has no label access`
  $l = \mathcal{C}_f(q_i, \{k_j\})$;
  if $y_{q_i} == y_{k_l}$ **then** correct = correct + 1;
end
return correct$/(N \times Q)$

distribution over the labels, or the number of labels.

$$\begin{aligned} \mathcal{C}_f(q, \{k\}) &= \arg\max_j a_{\{k\}}(q, k_j) \\ a_{\{k\}}(q, k_j) &= \frac{\exp c(f(q), f(k_j))}{\sum_i \exp c(f(q), f(k_i))} \\ c(\mathbf{x}, \mathbf{y}) &= \frac{\mathbf{x} \cdot \mathbf{y}}{|\mathbf{x}| \cdot |\mathbf{y}|} \end{aligned} \tag{3}$$

We introduce the effect of limited label information in the multi-shot setting as part of our ablation studies. Inspired by ProtoNets [33] and MatchNets [39], we compute class centroids as representatives for classification. Since computing class centroids requires label information, we present those experiments as part of our ablation studies separately in Section 4.3.

4 Experiments

4.1 Experimental Setup

We describe the experimental setup in this section, including datasets, evaluation strategy, and hyperparameters for reproducibility.

Datasets We use experiments on three popular few-shot image classification benchmarks.

The ***mini*ImageNet** dataset [39] is a subset of ImageNet [8] and is a common few-shot learning benchmark. The dataset contains 100 classes and 600 examples per class. Following [30], we split the dataset into 64 base classes, 16 validation classes, and 20 novel classes. Following [39], we resize the images to 84×84 pixels via rescaling and center cropping.

Table 1. Average accuracy (in %) on the miniImageNet dataset. [1]Results from [2], which did not report confidence intervals. [2]AmDimNet [4] used extra data from the ImageNet dataset for training the network used to report mini-Imagenet numbers. [3]Results from our experiments adapting the published training code from [42]. OURS was implemented here using OURS_S pipeline and ATTN classifier. See Table 4 for more pipeline and classifier variants.

	Approach	Setting		Labels Used
		1-shot	5-shot	
Fully Supervised	MAML [10]	49.6 ± 0.9	65.7 ± 0.7	50,400
	CloserLook [6]	51.8 ± 0.7	75.6 ± 0.6	50,400
	RelationNet [36]	52.4 ± 0.8	69.8 ± 0.6	50,400
	MatchingNet [39]	52.9 ± 0.8	68.8 ± 0.6	50,400
	ProtoNet [33]	54.1 ± 0.8	73.6 ± 0.6	50,400
	Gidaris *et al.* [13]	55.4 ± 0.8	70.1 ± 0.6	50,400
	TADAM [29]	58.5 ± 0.3	76.7 ± 0.3	50,400
	SimpleShot [40]	62.8 ± 0.2	80.0 ± 0.1	38,400
	Tian *et al.* [38]	64.8 ± 0.6	82.1 ± 0.4	50,400
	S2M2 [26]	**64.9 ± 0.2**	**83.2 ± 0.1**	50,400
	Gidaris *et al.* [12]	63.77 ± 0.45	80.70 ± 0.33	50,400
Semi Supervised With Finetuning	Antoniou *et al.* [2][1]	33.30	49.18	21,600
	AmDimNet [4][2]	**77.09 ± 0.21**	**89.18 ± 0.13**	21,600
Semi Supervised And Label Free	Wu *et al.* [42][3]	32.4 ± 0.1	39.7 ± 0.1	0
	BoWNet [14]	**51.8**	**70.7**	0
	Ours	50.1 ± 0.2	60.1 ± 0.2	0

Table 2. Average accuracy (in %) on the CIFAR100FS dataset. [1]Results from [25]. [2]Results from our experiments adapting the published training code from [42]. OURS was implemented here using OURS_S pipeline and ATTN classifier. See Table 4 for more pipeline and classifier variants.

	Approach	Setting		Labels Used
		1-shot	5-shot	
Fully Supervised	MAML [10][1]	58.9 ± 1.9	71.5 ± 1.0	48,000
	RelationNet [36][1]	55.0 ± 1.0	69.3 ± 0.8	48,000
	ProtoNet [33][1]	55.5 ± 0.7	72.0 ± 0.6	48,000
	R2D2 [3][1]	65.3 ± 0.2	79.4 ± 0.1	48,000
	MetaOptNet [25]	72.8 ± 0.7	85.0 ± 0.5	60,000
	Tian *et al.* [38]	73.9 ± 0.8	86.9 ± 0.5	48,000
	S2M2 [26]	**74.8 ± 0.2**	**87.5 ± 0.1**	48,000
	Gidaris *et al.* [12]	73.62 ± 0.31	86.05 ± 0.22	48,000
Semi Supervised And Label Free	Wu *et al.* [42][2]	27.1 ± 0.1	31.3 ± 0.1	0
	Ours	**53.0 ± 0.2**	**62.5 ± 0.2**	0

Table 3. Avg accuracy (in %) on FC100 dataset. [1]Results from [25]. [2]Results from our experiments adapting published training code from [42]. OURS was implemented here using OURS_S pipeline and ATTN classifier. See Table 4 for more pipeline and classifier variants.

	Approach	Setting		Labels Used
		1-shot	5-shot	
Fully Supervised	ProtoNet [33][1]	35.3 ± 0.6	48.6 ± 0.6	48,000
	TADAM [29][1]	40.1 ± 0.4	56.1 ± 0.4	48,000
	MTL [35]	45.1 ± 1.8	57.6 ± 0.9	60,000
	MetaOptNet [25]	**47.2 ± 0.6**	**62.5 ± 0.6**	60,000
	Tian *et al.* [38]	44.6 ± 0.7	60.9 ± 0.6	48,000
Semi Supervised And Label Free	Wu *et al.* [42][2]	27.4 ± 0.1	32.4 ± 0.1	0
	Ours	**37.1 ± 0.2**	**43.4 ± 0.2**	0

Table 4. An ablation study of multiple classifiers on various backbones. Average accuracy and 95% confidence intervals are reported over 10,000 rounds. The '`-C`' classifiers use class labels to compute the centroids per class. Best results per dataset and few-shot task are in **bold**.

Train	Test	miniImagenet		CIFAR100		FC100	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Ours_S	1NN	48.7 ± 0.2	59.0 ± 0.2	52.0 ± 0.2	61.7 ± 0.2	36.0 ± 0.2	42.6 ± 0.2
	Attn	**50.1±0.2**	60.1 ± 0.2	**53.0±0.2**	62.5 ± 0.2	**37.1±0.2**	43.4 ± 0.2
	1NN-C	-	**64.6±0.2**	-	**65.8±0.2**	-	**47.2±0.2**
	Attn-C	-	63.6 ± 0.2	-	63.8 ± 0.2	-	46.0 ± 0.1
Ours_M	1NN	29.7 ± 0.1	39.0 ± 0.1	26.7 ± 0.1	31.1 ± 0.2	28.1 ± 0.1	33.2 ± 0.2
	Attn	36.0 ± 0.2	45.2 ± 0.1	27.9 ± 0.1	32.3 ± 0.2	30.4 ± 0.1	35.2 ± 0.2
	1NN-C	-	45.1 ± 0.2	-	32.4 ± 0.2	-	34.9 ± 0.2
	Attn-C	-	48.4 ± 0.2	-	32.4 ± 0.1	-	35.6 ± 0.2

We also perform experiments on a subset of the CIFAR-100 [23] dataset, as in [29]. This dataset consists of 100 image classes in total with each class having 600 images of size 32 × 32 pixels. Following the setup in [29], we split the classes into 60 base, 20 validation, and 20 novel classes for few-shot learning. This dataset is referred to as **CIFAR-100FS** in our experiments.

We also use the **FC100** [29] (FewShot CIFAR100) dataset for our experiments. The 100 classes of the CIFAR-100 [23] dataset are grouped into 20 superclasses to minimize information overlap. The train split contains 60 classes belonging to 12 superclasses, the validation and test splits contain 20 classes belonging to 5 superclasses each. Q (as in Section 3.1) is chosen to be 15 across all datasets.

Evaluation Protocol We follow a standard evaluation protocol following earlier literature in the field [31,40]. The classifier is presented with 10,000 tasks and average accuracy is reported. Given a test set consisting of C novel classes, we generate an N-way K-shot task as follows. N classes are uniformly sampled from the set of C classes without replacement. From each class, K key and $Q = 15$ query images are uniformly sampled without replacement. The classifier is presented with the key images and then used to classify the query images. Following prior work [40], we focus on 5-way 1-shot and 5-way 5-shot benchmarks.

Models and Implementation Details All experiments use a ResNet-50 [18] backbone. SimCLR [5] pre-training is done for 500 epochs with a learning rate of 0.1, Nesterov momentum of 0.9, and weight decay of 0.0001 on the respective datasets. Data augmentations of RandomResizedCrop and ColorDistortion were found to achieve the best results. The augmentations use default hyperparameters from [5].

MoCo [19] pre-training is done for 800 epochs over the respective training sets using the default parameters from MoCo-v2 [7]. Downstream tasks use the query (non-momentum) encoder network.

4.2 Results

Tables 1, 2 and 3 present our results on the *mini*ImageNet, CIFAR100FS and FC100 datasets respectively. For a more comprehensive comparison, we also adapt the work presented in Wu *et al.* [42] to include another unsupervised method in these results. Accuracies are averaged over 10,000 tasks and reported with 95% confidence intervals. Note that we report the number of labels used by each method in each of the above tables. The number of labels used by the methods are computed as follows: if the network trains by performing gradient updates over the training labels, we count the labels in the training set; if the network fine-tunes over the test labels or uses test labels to compute class representations, we count the labels in the test set; if the network uses training and validation data to report results, we count training and validation labels. Unless otherwise specified in the respective works, we assume that the validation set is not used to publish results, and that the train and test pipelines are the same.

Our method achieves strong baselines on the benchmarks while using extremely limited label information, as can be seen in the comparison with Wu *et al.* [42] which operates in the same setting. These are the only two methods across all benchmark datasets that use **almost no** label information. BowNet [14] operates in a similar setting and performs well on the mini-Imagenet benchmark by computing cluster centres in the representation space as a visual vocabulary. Other methods are provided for comparison and the label count is calculated accordingly. The supervised methods use tens of thousands of labels, which can be very expensive depending on a particular domain. Our methodology seeks to provide a pathway to solving problems in such settings with no annotation cost whatsoever.

A higher number of input images increases the classification accuracy, as seen in our 5-way-5-shot tasks. The best results are achieved over the challenging miniImageNet dataset, followed by CIFAR100FS and FC100 datasets. This is expected as FC100 is a coarse-grained classification task and is specifically constructed to have dissimilar classes.

In Section 3.1, we proposed the use of two test-time classifiers: 1NN and ATTN. We report ablation studies on their performances in Table 4. While the 1NN classifier achieves strong baselines (in line with previous work [5]), the ATTN classifier consistently improves accuracies by 2-10%, with more impressive gains in the multi-shot setting. This suggests that using different distance measures in the representation space is a valid area for future inquiry.

4.3 Ablation Studies

In this section, we explore different variations of our pipeline and investigate the effect on performance across datasets. Table 4 presents the results.

What if we had labels? To investigate the effect of introducing labels at test time, we introduce CENTROID versions of our classifiers: 1 Nearest Neighbours Centroid (1NN_CENTROID), and Soft Cosine Attention Centroid (ATTN_CENTROID), in the multi-shot setting. Following [33,40,39], the CENTROID versions of these

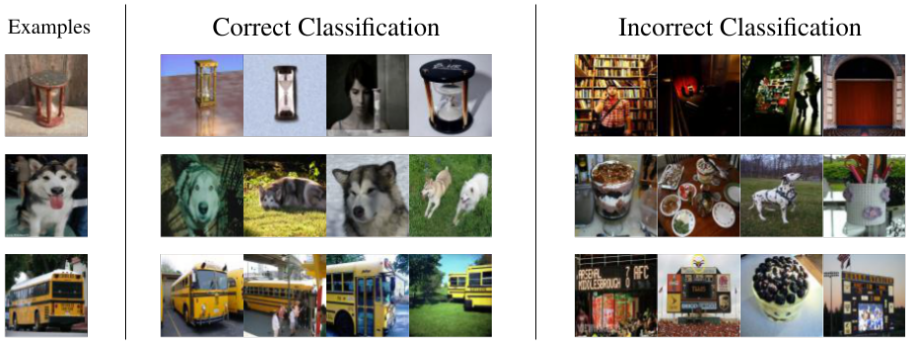

Fig. 3. Visualizing a few examples from the miniImagenet test set using the OURS_S pipeline. **Far Left:** One labelled example visualized per class. **Middle:** Few correctly classified examples from the test set. **Right:** Mis-classified examples. Similarity in texture and coarse object category are contributing factors for mis-classification.

classifiers compute class representatives as the centroids of the key images provided at test time. Few-shot classification is then done by comparing each query image against each class centroid, essentially treating the class representative (or exemplar) as the new key image for that class. Using label information to compute class centroids increases performance by 2-4%.

Qualitative Analysis: Figure 3 presents a few qualitative examples from our results on the ***mini*ImageNet** dataset using our OURS_S pipeline. In the second row, we observe that the network fails on a fine-grained classification task. It classifies a DALMATIAN image (black and white polka-dotted dog) as a HUSKY. Since both categories are dog breeds, they are closely related and pose a difficult few-shot problem. However, when the classes are coarse-grained and fairly well-separated, our method shows that one can achieve reasonable performance with limited label information.

5 Conclusion

We present a new framework for few-shot classification extremely limited label information using computationally simple pipelines. This is more challenging than existing work which uses label information at various points during training or inference. By learning contrastive representations using self supervision, we achieve competitive baselines while using **almost no** labels, which is orders of magnitude fewer labels than existing work. In our ablation studies, we present a qualitative analysis of our classifier and investigate the effect of limited label information. Our results indicate that the choice of self-supervised training task and distance function in the representation space are interesting lines of future inquiry. We also show that using limited label information to compute class representatives at test time is beneficial. This suggests that clustering quality has a direct impact on performance. The objective was to achieve a reasonable per-

formance using few-shot classification with limited label information. We believe this work is an important first step towards label-free few-shot learning methods.

References

1. Antoniou, A., Edwards, H., Storkey, A.: How to train your MAML. In: ICLR (2019) 1
2. Antoniou, A., Storkey, A.: Assume, augment and learn: Unsupervised few-shot meta-learning via random labels and data augmentation. ICML (2019) 7
3. Bertinetto, L., Henriques, J.F., Torr, P., Vedaldi, A.: Meta-learning with differentiable closed-form solvers. In: ICLR (2019) 3, 7
4. Chen, D., Chen, Y., Li, Y., Mao, F., He, Y., Xue, H.: Self-supervised learning for few-shot image classification (2019) 1, 3, 7
5. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML (2020) 2, 3, 8, 9
6. Chen, W.Y., Liu, Y.C., Kira, Z., Wang, Y.C., Huang, J.B.: A closer look at few-shot classification. In: ICLR (2019) 5, 7
7. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020) 8
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR (2009) 6
9. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: ICCV (2015) 3
10. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML (2017) 1, 7
11. Finn, C., Xu, K., Levine, S.: Probabilistic model-agnostic meta-learning. In: NeurIPS (2018) 1
12. Gidaris, S., Bursuc, A., Komodakis, N., Pérez, P.P., Cord, M.: Boosting few-shot visual learning with self-supervision. In: ICCV (2019) 1, 3, 7
13. Gidaris, S., Komodakis, N.: Dynamic few-shot visual learning without forgetting. In: CVPR (2018) 7
14. Gidaris, S., Bursuc, A., Komodakis, N., Pérez, P., Cord, M.: Learning representations by predicting bags of visual words. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6928–6938 (2020) 7, 9
15. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. In: ICLR (2018) 3
16. Grant, E., Finn, C., Levine, S., Darrell, T., Griffiths, T.: Recasting gradient-based meta-learning as hierarchical bayes. arXiv preprint arXiv:1801.08930 (2018) 1
17. Hariharan, B., Girshick, R.: Low-shot visual recognition by shrinking and hallucinating features. In: ICCV (2017) 2
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 8
19. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR (2020) 2, 3, 8
20. Horn, G.V., Perona, P.: The devil is in the tails: Fine-grained classification in the wild. CoRR **abs/1709.01450** (2017), http://arxiv.org/abs/1709.01450 1
21. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: ICCV (2019) 2, 3

22. Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: ICML-W (2015) 2, 3, 4
23. Krizhevsky, A.: Learning multiple layers of features from tiny images. University of Toronto (2009) 8
24. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: ECCV (2016) 3
25. Lee, K., Maji, S., Ravichandran, A., Soatto, S.: Meta-learning with differentiable convex optimization. In: CVPR (2019) 3, 7
26. Mangla, P., Kumari, N., Sinha, A., Singh, M., Krishnamurthy, B., Balasubramanian, V.N.: Charting the right manifold: Manifold mixup for few-shot learning. In: WACV (2020) 3, 7
27. Nguyen, C., Do, T.T., Carneiro, G.: Uncertainty in model-agnostic meta-learning using variational inference. In: WACV (2020) 1
28. Noroozi, M., Favaro, P.: Unsupervised learning of visual represention by solving jigsaw puzzles. In: ECCV (2016) 3
29. Oreshkin, B., Rodríguez López, P., Lacoste, A.: Tadam: Task dependent adaptive metric for improved few-shot learning. In: NeurIPS (2018) 1, 7, 8
30. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning. In: ICLR (2017) 1, 6
31. Rusu, A.A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S., Hadsell, R.: Meta-learning with latent embedding optimization. In: ICLR (2019) 8
32. Satorras, V.G., Estrach, J.B.: Few-shot learning with graph neural networks. In: ICLR (2018) 3
33. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: NeurIPS (2017) 1, 3, 6, 7, 9
34. Su, J.C., Maji, S., Hariharan, B.: Boosting supervision with self-supervision for few-shot learning. ArXiv **abs/1906.07079** (2019) 1, 3
35. Sun, Q., Liu, Y., Chua, T.S., Schiele, B.: Meta-transfer learning for few-shot learning. In: CVPR (2019) 7
36. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: CVPR (2018) 3, 7
37. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. arXiv preprint arXiv:1906.05849 (2019) 3
38. Tian, Y., Wang, Y., Krishnan, D., Tenenbaum, J.B., Isola, P.: Rethinking few-shot image classification: a good embedding is all you need? ECCV (2020) 7
39. Vinyals, O., Blundell, C., Lillicrap, T., kavukcuoglu, k., Wierstra, D.: Matching networks for one shot learning. In: NeurIPS (2016) 1, 3, 5, 6, 7, 9
40. Wang, Y., Chao, W.L., Weinberger, K.Q., van der Maaten, L.: Simpleshot: Revisiting nearest-neighbor classification for few-shot learning (2019) 2, 5, 7, 8, 9
41. Wang, Y., Yao, Q., Kwok, J.T., Ni, L.M.: Generalizing from a few examples: A survey on few-shot learning. ACM Computing Surveys (CSUR) (2019) 1, 3
42. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR. pp. 3733–3742 (2018) 7, 9
43. Yoon, J., Kim, T., Dia, O., Kim, S., Bengio, Y., Ahn, S.: Bayesian model-agnostic meta-learning. In: NeurIPS (2018) 1
44. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV (2016) 3