Abstract

Few-shot learning methods offer pre-training techniques optimized for easier later adaptation of the model to new classes (unseen during training) using one or a few examples. This adaptivity to unseen classes is especially important for many practical applications where the pre-trained label space cannot remain fixed for effective use and the model needs to be "specialized" to support new categories on the fly. One particularly interesting scenario, essentially overlooked by the few-shot literature, is Coarse-to-Fine Few-Shot (C2FS), where the training classes (e.g. animals) are of much 'coarser granularity' than the target (test) classes (e.g. breeds). A very practical example of C2FS is when the target classes are sub-classes of the train classes (seen) or sub-classes of classes unseen during training.

In this paper, we introduce a novel 'Angular normalization' module that allows to effectively combine supervised and self-supervised contrastive pre-training to approach the proposed C2FS task, demonstrating significant gains in a broad study over multiple baselines and datasets. We hope that this work will help to pave the way for future research on this new, challenging, and very practical topic of C2FS classification.

1. Introduction

In the most commonly encountered learning scenario, supervised learning, a set of target (class) labels is provided for a set of samples (images) using which we train a model (CNN [26, 18, 53] or Transformer [66, 40]) that casts these samples into some representation space from which predictions are made, e.g. using a linear classifier. Nevertheless, while supervised learning is a very common setting, in many practical applications the set of the target labels of interest is not static, and may change over time. One good example is few-shot learning [57, 49, 55], where a model is pre-trained in such a way that more classes could be added later with only very few additional labeled examples used for adapting the model to support these new classes.

However, in previous few-shot learning works most (if not all) of the new classes: (i) are separate from the classes the model already knows, in the sense that they either belong to a different branch of the class hierarchy or are siblings to the known classes; and (ii) are of same or similar level of granularity (same level of class hierarchy). But what about the very practical situation when the new classes are fine-grained sub-classes strictly included inside the known (coarse) classes being their descendants in the class taxonomy? This situation typically occurs during the lifespan of the model when the application requires separating some sub-classes of the current classes into separate classes and yet when the training dataset was created these (unknown in advance) sub-classes were not annotated. For example, this could occur in product specialization for product search, or during personalizing a generic model to a specific customer. Naturally, going back to re-labeling each time this occurs is much too costly to be an option.

In this paper, we target the Coarse-to-Fine Few-Shot (C2FS) task (Fig. 1) where a model pre-trained on a set of base classes (denoted as the 'coarse' classes), needs to
adapt on the fly to an additional set of target (‘fine’) classes of much ‘finer granularity’ than the training classes. The target classes could be sub-classes of the base classes (a particularly interesting case), or they could be a separate set, yet requiring much stronger (than base classes) attention to fine-grained details in order to visually separate. To be efficient, we want this adaptation to occur using only one or few samples of the fine (sub-)classes. Intuitively, this setup is particularly challenging for models pre-trained on the coarse classes in ‘the standard’ supervised manner, as: (a) standard supervised learning losses do not care about the intra-class arrangement of the samples belonging to the same class in the model’s feature space $F$, as long as these samples are close to each other and the regions associated with different classes are separable (Fig. 2 top-left) - potentially causing the sub-classes to spread arbitrarily inside same-class-associated regions of $F$ thus hindering their separability (Fig. 2 top-right); and (b) $F$ is retaining the information on the attributes needed to predict the set of target ‘coarse’ labels, while at the same time reducing intra-class variance and suppressing attributes not relevant to the task for better generalization, which may eliminate the intra-class distinctions between sub-classes (Fig. 2 top-right).

In contrast to supervised learning, recently emerged contrastive self-supervised methods [3, 17, 2, 13] were proven highly instrumental in learning good features without any labels. These methods are able to pre-train effectively attaining almost the same representation (feature) quality as fully supervised counterparts, and even surpassing it when transferring to other tasks (e.g. detection [17]). Even more importantly, these methods are optimizing features for ‘in-

Figure 2. Learned embedding tSNE visualization: Top - coarse-supervised baseline, Bottom - ours (ANCOR). Left - coarse classes, right - fine sub-classes of one arbitrary coarse class. Stars are embeddings of the linear classifier (class) weight vectors, black arrows point from the class weight to the fine sub-classes centroids. Clearly, ANCOR induces order on the sub-classes arranging them nicely around the class weight and making them separable.

Figure 3. Angular Normalized COntrastive Regularization (ANCOR): our method jointly employs inter-class supervised and intra-class self-supervised contrastive losses that would pull to different directions without our proposed Angular normalization component that separates the forces applied by the two losses to different planes leading to significant performance gains.

To summarize, our contribution is threefold: (i) we propose the Coarse-to-Fine Few-Shot (C2FS) task of training using only coarse class labels and adapting to support finer (sub-)classes with few (even one) examples; (ii) we propose the ANCOR approach for C2FS task, based on effective multi-task combination of supervised inter-class and self-supervised intra-class learning, featuring a novel angular normalization component to minimize friction and maximize the synergy between the two tasks; (iii) we offer extensive evaluation and analysis showing the strength of our proposed ANCOR approach on a variety of datasets
and compared to a diverse set of baselines.

2. Related Work

Self-supervised learning. While the onset of deep-learning was pre-dominantly ruled by supervised learning [26, 18, 53], recently many self-supervised representation learning methods have emerged. These works generate different self-induced (pretex) pseudo-labels for unlabeled data and drive the visual feature learning without any external supervision. Earlier works used predicting patch position [7], image colorization [65], jigsaw puzzles [36], image in-painting [38], predicting image rotations [12], and others as pretext tasks. Yet, more recently, [56, 54, 3, 17, 5, 13, 2] have demonstrated the power of contrastive instance discrimination, significantly surpassing previous results and narrowing the gap with supervised methods. SimCLR [3] defined positive pairs as two augmentations of the same image and contrasted them with other images of the same batch. Instead, MoCo [17, 5] contrasted with samples extracted from a dynamic queue produced by a slowly progressing momentum encoder. SWAV [2] uses a clustering objective for computing the contrastive loss, BYOL [13] replaces the contrastive InfoNCE loss with direct regression between positive pairs, essentially removing the need for negative samples, and [59] explores the effect of different contrastive augmentation strategies. Interestingly, contrastive methods have recently shown promising results for domain adaptation [22] and supervised learning [23]. Intuitively, for solving our C2FS task both inter-class (between the coarse classes) and intra-class (within the classes) separation are jointly required. Supervised methods are better at inter-class separation, but are worse in intra-class separation (Fig. 2 top right), while contrastive self-supervised methods are better on intra-class and are worse on inter-class discrimination (Tab. 6). In this paper, we show how to properly combine the two to enjoy the benefits of both.

Few-shot learning. Meta-learning methods, which are very popular in the few-shot literature [57, 49, 51, 28, 10, 30, 67, 41, 35, 45, 4, 48, 37, 64, 63, 9, 27, 16, 62, 19, 32, 8], learn from few-shot tasks (or episodes) rather than from individual labeled samples. Such tasks are small datasets, with a few labeled training (support) examples, and a few test (query) examples. The goal is to learn a model that at test time can be adapted to support novel categories, unseen during training on the base categories (with abundant train data). In [39, 29, 24, 11, 31] additional unlabeled data is used, [60, 47] leverage additional semantic information available for the classes, and [11, 21, 1, 50] examine the usage of unsupervised or self-supervised training in the context of a standard few-shot learning. Recently, several works have noted that standard supervised pre-training on the base classes followed by simple fine-tuning attains (if done right) comparable and mostly better performance than the leading meta-learning methods [58, 55, 31], even more strikingly so when the target (test) classes are in a different visual domain [14]. Here, we build upon this intuition and do not use meta-learning for pre-training. Note though that in all these approaches, the base categories of the training set and the set of test categories are assumed to be of similar granularity (e.g. some ImageNet categories as the base and others as the target, or species of birds as the base and as the target, etc.). In particular, no method was proposed to tackle a commonly occurring (and hence very practical) situation of target classes being the sub-classes of the base classes. Generally, situations when the target classes are from lower level of the classes hierarchy than the base classes have not been considered in the above works.

Coarse and fine learning. Relatively few works have considered learning problems entailing mixing of coarse and fine labels. Several works [43, 15, 52] consider the partially fine-supervised training setting, where during training a mix of (equal #) coarse- and fine-labeled samples is used for training. In contrast, in this work we focus on training using only coarse-labeled data, while fine categories are added at test time and from very few examples (usually one). [20] extends the partially fine-supervised setting to unbalanced splits between coarse- and fine-labeled samples via a MAML-like [10] optimization, albeit in multi-label classification setting (and non-standard performance measure). Similarly, [44] also explores the unbalanced partially fine-supervised setting. Prototype propagation between coarse and fine classes on a known (in advance) class ontology graph is explored in [33] in the few-shot context and in partially fine-supervised training setting, also assuming full knowledge of the classes graph (which also includes the test classes). Finally, a concurrent work of [61] focuses on a scenario similar to C2FS. In [61] the learning is split into three separate consecutive steps: (i) feature learning using a combination of supervised learning with coarse labels and general batch-contrastive learning disregarding classes; (ii) greedy clustering of each coarse class into a set of (fine) pseudo-sub-classes in the resulting feature space; (iii) applying a meta-learning techniques to fine-tune the model to the set of generated pseudo-sub-classes. In contrast, in our ANCOR approach the model is trained end-to-end, contrastive learning is done within the coarse classes, and we propose a special angular loss component for significantly enhancing the supervised and self-supervised contrastive learning synergies. As a result, in section 4.4.2, we obtain good gains over [61] on the same tieredImageNet test.

3. Method

3.1. Coarse-to-Fine Few-Shot (C2FS) task

Denote by \(Y_{\text{coarse}} = \{y_1, ..., y_R\}\) a set of \(R\) coarse training classes (e.g. kinds of animals: dog, cat, fish, ...), and let \(S_{\text{train}}^{\text{coarse}} = \{(I_j, y_j)|y_j \in Y_{\text{coarse}}\}_{j=1}^N\) be a set of
Figure 4. ANCOR method overview - training flow. For illustrative purposes, showing a single image flow instead of a batch. The query $q$ and the positive key $k_+$ are computed from two random augmentations of the input image via the model $(B \rightarrow E)$ and its momentum-updated counterpart $(B_k \rightarrow E_k)$ [17]. The negative keys $k_-$ are taken from the queue $Q_y$ corresponding to the coarse class $y$ of the image. $q$ is classified to the coarse classes by linear classifier $C$, which is followed by the (supervised) CE loss $L_{CE}$. The (self-supervised) InfoNCE loss $L_{cont}$ is not applied directly on $q$, $k_+$, and $k_-$. Doing so would result in disagreement between supervised and self-supervised losses objectives. Instead, $q$, $k_+$, and $k_-$ are first normalized with the proposed Angular normalization component $A$ using the linear classifier $C$ weights $W$ corresponding to $y$. This disentangles the forces applied by the two losses, effectively leading to greater synergy between them and consequently to increased performance, as demonstrated by our experiments and ablations.

$N$ training images annotated (only) with $Y_{\text{coarse}}$. Let $Y_{\text{fine}} = \{y_1, ..., y_{1,k_1}, y_{2,1}, ..., y_{2,k_2}, ..., y_{R,1}, ..., y_{R,k_R}\}$ be a set of fine sub-classes (e.g. animal breeds) of the coarse classes $Y_{\text{coarse}}$. In our experiments we also explore the case when fine classes are sub-classes of unseen coarse classes. Let $B$ be an encoder (CNN backbone) mapping images to a $d$-dimensional feature space $F \subset \mathbb{R}^d$ (i.e. $B(I_j) = F_j \in F$) trained on $\mathcal{S}_{train}$. Provided at test time with a small $k$-shot training set for a subset $Y_{\text{fine}}^m \subseteq Y_{\text{fine}}$ of $m$ fine classes: $\mathcal{S}_{\text{train}}^m = \{(I_r, y_r) | y_r \in Y_{\text{fine}}^m\}$ and let $k_{r,m}$ be a maximal accuracy on the $Y_{\text{fine}}^m$ fine classes test set. For example, $m$ could also be $\sum k_{r,m}$ making $Y_{\text{fine}}^m = Y_{\text{fine}}$ (‘all-way’). Note that during training of $B$ the set of fine classes $Y_{\text{fine}}$ is unknown. Also note that according to [58, 55, 31], SOTA few-shot performance can be achieved even without modifying $B$ when adapting to (unseen) test classes.

3.2. The (ANCOR) approach

At its core, our method focuses on learning $B$ combining (with added synergy) supervised learning for inter-class separation of the coarse classes $Y_{\text{coarse}}$ and contrastive self-supervised learning for separating the fine sub-classes within each coarse class ($Y_{\text{fine}}$). The training architecture of ANCOR is illustrated in Fig. 4. Our model is comprised of: (i) a CNN encoder $B: I \rightarrow \mathbb{R}^d$ with Global Average Pooling (GAP) on top (e.g. ResNet50 mapping images to 2048-dimensional vectors); (ii) an MLP embedder module $E: \mathbb{R}^d \rightarrow \mathbb{R}^e$ with $e < d$ (e.g. 2048 $\rightarrow$ 2048 $\rightarrow$ 128), $E$ also includes $L_2$ normalization of the final vector; (iii) second pair of (momentum) encoder $B_k$ and (momentum) embedder $E_k$ for encoding the positive keys in the contrastive objective that are momentum updated from $B$ and $E$ respectively (following [17]); (iv) a linear classifier $C: \mathbb{R}^e \rightarrow Y_{\text{coarse}}$ without bias and $W \in \mathbb{R}^{R \times e}$ its weight matrix (so $W \cdot E(B(I))$ are the $R$ coarse classes logits of $C$); (v) a set of per-class ‘negative-instance’ queues $\{Q_i\}_{i=1}^R$ with each queue: $Q_i \in \mathbb{R}^{e \times K}$ of length $K$ (different from [17] that utilized a single queue for the entire dataset); and (vi) an Angular normalization module explained below, that is used for inducing synergy between the supervised and self-supervised contrastive losses by disentangling the forces they apply on samples in the feature space.

The training (Fig. 4) proceeds in batches, but for clarity here we describe the training process for a single training image $I$ annotated with a coarse class label $y \in Y_{\text{coarse}}$. Abusing notation let $y$ also denote the index of the coarse class in $Y_{\text{coarse}}$. We first augment $I$ twice to get $I_e$ and $I_k$ which are then passed through the corresponding encoders and embedders to retrieve $q = E(B(I_e))$ and $k_+ = E_k(B_k(I_k))$. We also set $k_- = Q_y$ (being ‘momentum’ embeddings of previously encountered samples of the same coarse class $y$, at the end of the training cycle $k_+$ is also added to $Q_y$). Our two loss functions are: $L_{CE}(C(q), y)$ being the coarse-supervised softmax Cross Entropy (CE) on $C(q) = W \cdot q$ logits and 1-hot vector for $y$ coarse class label; and $L_{cont}(q, k_+, k_-)$ being the $y$ class-specific self-supervised contrastive InfoNCE loss applied to query $q$ positive key $k_+$ and negative keys $k_-$. However, if used naively, $L_{CE}$ would try to push $q$ towards the other samples of the same class $y$, while at the same time $L_{cont}$ would try to push it away from them (as $k_-$ represents the other samples of the class). This would diminish the synergy between the losses and, as shown in the ablation study 4.5.1, would
result in significant performance drop on the C2FS task.

Angular normalization. To improve this synergy, we propose a new module which we name ‘angular normalization’. For a given image $I$ with embedding $q$ and coarse label $y$, the logit for class $y$ in classifier $C$ is $W_y \cdot q$, where $W_y$ is the $y^{th}$ row of $W$. Thus, the supervised loss $L_{CE}(C(q), y)$ is minimized when $W_y \cdot q$ is maximized and $W_{i\neq y} \cdot q$ is minimized, or in other words, when $q$ (unit vector, as the embedder $E$ ends with $L_2$ normalization) shifts towards being in the direction of $W_y$. And this is the same for all images of class $y$ essentially encouraging their collapse to $W_y/||W_y||$ (the unit vector closest to $W_y$).

But this collapse is in direct conflict of interests with the $y$ class-specific InfoNCE contrastive loss $L_{cont}(q, k_+, k_-)$ that tries to push $y$’s samples away from each other (Fig. 5 top). To solve this, we propose a simple method that can be used to induce synergy between $L_{CE}$ and $L_{cont}$. We define the $y$-class specific angular normalization:

$$A(x, W, y) = \angle x = \frac{x \cdot W_y}{||x|| \cdot ||W_y||}, \quad (1)$$

which converts any unit vector $x/||x||$ into a unit vector representing its angle around $W_y/||W_y||$. With the above definition of angular normalization $A$, we replace the $q$, $k_+$, and $k_-$ in $L_{cont}$ with their $y$-class-specific normalized versions:

$$\angle q = A(q, W, y) \quad (2)$$
$$\angle k_+ = A(k_+, W, y) \quad (3)$$
$$\angle k_- = A(k_-, W, y) \quad (4)$$

and obtain our final (full) loss function:

$$L = L_{CE}(C(q), y) + L_{cont}(\angle q, \angle k_+, \angle k_-) \quad (5)$$

where the angular normalized contrastive loss $L_{cont}(\angle q, \angle k_+, \angle k_-)$ operates in the space of angles

---

Table 1. Datasets: L17, NL26, E13 & E30 are the Living17, NonLiving17, Entity13 & Entity30 from BREEDS. For Tiered the train/val/test classes are non overlapping; For CIFAR-100 and BREEDS val was 10% of the train, Tiered has pre-defined val set.

3.3. Few-shot testing on fine classes

At test time, only the encoder $B$ followed by $L_2$ normalization is retained as the feature extractor and following [6] the MLP embedder $E$ is dropped (for higher performance). According to our definition of C2FS task, only a small $k$-shot and $m$-way training set $S_{train}^{fine}$ is available for adapting the model to support the fine-classes. For the few-shot classifier we use the method of [55]. For every few-shot episode, we create 5 augmented copies for every support sample, and train a logistic regression model on the support set encoded using $B$ followed by $L_2$ normalization. The model with the resulting logistic regression classifier on top is then used to classify the query samples of the episode.

4. Experiments

4.1. Datasets

Our experiments were performed on: (i) BREEDS [46], four datasets derived from ImageNet with class hierarchy re-calibrated by [46] so classes on same hierarchy level are of the same visual granularity (not so in the WordNet hierarchy); (ii) CIFAR-100 [25]; and (iii) tieredImageNet [42], a subset of ImageNet, with train/val/test built from different coarse classes. Datasets are summarised in Table 1.

4.2. Baselines

As C2FS task is new, we propose a diverse set of natural baselines and upper bounds for the task. We also compare to the concurrent work of [61] where applicable. For fair comparison, we use 200 train epochs for all models (ours, baselines). The effect of longer training is explored in section 4.5.5. Hyper-parameters of all compared methods were tuned on val sets. For the baselines and upper bounds we also used the best training practices from [55], including self-distillation, which consistently improved their performance. For fairness, in each experiment all compared methods use the same backbone architecture (for the encoder $B$).

Coarse Baselines: models trained using coarse labels
4.3. Additional implementation details

The encoder $B$ was: ResNet-50 [18] for the $224 \times 224$ datasets (BREEDS [46]), and ResNet-12 for the small resolution datasets (CIFAR-100 [25] and tieredImageNet [42]), as is common in the self-supervised and few-shot works respectively. The output dim of these networks is $d = 2048$ or $d = 640$, respectively. Our MLP embedder $E$ consisted of two stacked fully connected layers with ReLu activation: $d \rightarrow d \rightarrow e$, we used $e = 128$ in all experiments. We used cosine-annealing with warm restarts schedule [34] with 20 epochs per cycle. We trained on 4 V100 GPUs, with a batch size $b = 256$ and base learning rate $lr = 0.03$ for BREEDS, and $b = 1024$ and $lr = 0.12$ for CIFAR-100 and tieredImageNet. We used $wd = 1e^{-4}$ weight decay. We used queue size $K = 65536$, InfoNCE temperature of $\tau = 0.2$, and $m = 0.999$ for the momentum encoder ($m = 0.99$ for CIFAR-100). All hyper-parameters were optimized on val sets, using the same optimization for our method and the baselines. Code: github.com/guybuk/ANCOR.

4.4. Results

We report results for 5-way $k$-shot and all-way $k$-shot 15-query tests. Following [55] we evaluate on 1000 random episodes and report the mean accuracy and the 95% confidence interval. Unless explicitly stated $k = 1$. Effect of more shots is evaluated in section 4.5.2. The test classes of each episode are a random subset of the set of fine classes $\mathcal{Y}_{fine}$ (or all of them in all-way tests). In section 4.5.3, in order to further investigate the sources of our improvements, we evaluate a special case of 'intra-class' testing, when all the categories of the episode belong to the same (randomly sampled) coarse class.

4.4.1 Unseen fine sub-classes of seen coarse classes

We first evaluate the core use-case of the C2FS task, namely training on coarse classes $\mathcal{Y}_{coarse}$ and generalizing to fine sub-classes of those classes $\mathcal{Y}_{fine}$ as defined in section 3.1. We used BREEDS and CIFAR-100 for this evaluation, and the results are reported in Tables 2 and 3 respectively. As can be seen, ANCOR significantly outperforms the coarse baselines across all datasets, in both the 5-way and all-way tests (e.g. on BREEDS, by over 10% on 5-way and over 5% on all-way). Notably, in NONLIVING-26 5-way, our model surpasses even the Fine models. Moreover, on BREEDS we observe large gains (over 5% in 5-way and over 6% in all-way) over full ImageNet pre-trained self-supervised baselines in all datasets, which significantly outperforms the coarse baselines across all datasets, in both the 5-way and all-way tests (e.g. on BREEDS, by over 10% on 5-way and over 5% on all-way). Notably, in NONLIVING-26 5-way, our model surpasses even the Fine models. Moreover, on BREEDS we observe large gains (over 5% in 5-way and over 6% in all-way) over full ImageNet pre-trained self-supervised baselines in all datasets.

This suggests that even significantly larger and more diverse training data available to those models is not sufficient to bridge the gap of coarse classes supervision which is needed for the C2FS, and is effectively utilized by ANCOR in good synergy with the self-supervised objective (due to our angular component). In the supplementary we explore a 'sub-population shift' variant of this scenario where the sub-classes appearing in training (with the coarse class label only) are non-overlapping with those appearing in the test.

| Method                | LIVING-17 5-way | LIVING-17 all-way | NONLIVING-26 5-way | NONLIVING-26 all-way | ENTITY-13 5-way | ENTITY-13 all-way | ENTITY-30 5-way | ENTITY-30 all-way |
|-----------------------|-----------------|-------------------|--------------------|----------------------|-----------------|------------------|-----------------|------------------|
| Fine (upper-bound)    | 91.10 ± 0.47    | 58.95 ± 0.16      | 85.25 ± 0.49       | 47.68 ± 0.13         | 91.01 ± 0.39    | 50.19 ± 0.08     | 91.65 ± 0.41    | 56.54 ± 0.09     |
| Fine+ (upper-bound)   | 78.39 ± 0.64    | 46.92 ± 0.16      | 74.95 ± 0.57       | 39.57 ± 0.11         | 85.98 ± 0.55    | 47.87 ± 0.09     | 85.43 ± 0.57    | 45.87 ± 0.09     |

Table 2. Results for different baselines on the four BREEDS datasets. Top section contains models trained with fine-grained labels that serve as upper bounds, middle section contains baselines, and the bottom section is our results.
4.5.2 Unseen fine sub-classes of unseen coarse classes

We use the tiered/ImageNet dataset to evaluate the second use-case for C2FS task: when the fine classes \( \mathcal{Y}_{\text{fine}} \) are not sub-classes of the training coarse classes \( \mathcal{Y}_{\text{coarse}} \), and in fact belong to a different branch of the classes taxonomy, and yet \( \mathcal{Y}_{\text{fine}} \) are of significantly higher visual granularity than \( \mathcal{Y}_{\text{coarse}} \). We train using only the coarse labels of tiered/ImageNet train classes, and evaluate on its standard test set (with its fine labels). As tiered/ImageNet is a standard few-shot benchmark, here we compare to the few-shot SOTA methods [55, 19, 62] trained using coarse train labels. The results of this experiment are summarized in Table 4 showing significant advantage of ANCOR over the baselines. Interestingly, the self-supervised MoCoV2 lags (significantly) behind, underlining the benefit of additional coarse supervision even in situation when test classes are descendants of different coarse classes. Notably, ANCOR also has a good 3%-5% advantage over the results of the concurrent work of [61] also dealing with coarse-and-fine few-shot interplay and performing the same experiment.

4.5. Analysis

We used the LIVING-17 dataset for the analysis.

4.5.1 Ablation study

Here we evaluate several design choices of the ANCOR approach. In terms of architecture, we compare two variants of the coarse-supervised branch: (i) 'Seq' being \( \mathcal{B} \to \mathcal{E} \to C \to \mathcal{L}_{\text{cont}} \); and (ii) 'Fork' being \( \mathcal{B} \to \mathcal{E}_{\text{layer}\#1} \to C \to \mathcal{L}_{\text{cont}} \). In both cases the self-supervised contrastive branch remains the same: \( \mathcal{B} \to \mathcal{E} \to \mathcal{L}_{\text{cont}} \). We also test the contribution of the angular normalization \( \mathcal{A} \) in the self-supervised branch (between \( \mathcal{E} \) and \( \mathcal{L}_{\text{cont}} \)). Of Seq and Fork, only Seq admits the possibility for angular normalization, as in Fork the classifier weights are of different dimensionality (dim = \( \mathcal{d} \)) than the output of \( \mathcal{E} \) (dim = \( \mathcal{e} \)). Finally, we also evaluate the contribution of having a queue \( \mathcal{Q}_c \) for each class ('Multi') vs. simply stacking all negative keys in a single shared queue \( \mathcal{Q} \) ('Single'). The results of this ablation are presented in Table 5. Looking at all-way results we draw the following conclusions: (i) difference between Seq and Fork is small; (ii) without angular normalization single queue is better than multi-queue in all-way, likely because \( \mathcal{L}_{\text{cont}} \) negative keys \( k_i \) taken from the shared queue do not belong to the same class (as opposed to Multi with queue per class) reducing the \( \mathcal{L}_{\text{cont}} \) drive to disperse same class elements in feature space; (iii) angular normalization gives a significant boost both to Single+Seq and to Multi+Seq; and (iv) Multi+Seq is best when combined with angular normalization forming ANCOR.

4.5.2 The effect of adding more shots

Here we evaluate the effect of adding more shots (number of support samples per class) to the few-shot episodes during testing. The results are shown in Fig. 6 demonstrating ANCOR’s consistent advantage (of about +5% accuracy in all-way) above the strongest of self-supervised (MoCoV2-ImageNet) and supervised (Coarse+) baselines.

4.5.3 Closer look at the fine classes performance

Success in C2FS entails both not confusing the coarse classes as well as not confusing the fine sub-classes of each class. Therefore we ask ourselves, is the benefit of ANCOR coming from less coarse confusion or from less (intra-class) fine confusion? Looking at Table 6 we see that ANCOR has better all-way coarse accuracy (using coarse-only labels of the test set), as well as better intra-class fine accuracy evaluated by creating the few-shot episodes by random sampling a coarse class \( y_i \in \mathcal{Y}_{\text{coarse}} \) and then generating a random episode from (all of) it's sub-classes \( \{y_{i,1}, \ldots, y_{i,k_i}\} \).
Consequently, ANCOR has the highest performance on the all-way fine test that we also saw in Tab. 2. Interestingly, Coarse, Coarse+, and MoCoV2 results indicate there is a trade-off between all-way coarse and intra-class-fine performances, trade-off apparently topped by ANCOR.

### 4.5.4 Closer look at the features

We further explore the feature space $\mathcal{F}$ learned by ANCOR using visualizations. Fig. 2 visualizes $\mathcal{F}$ on ENTITY-13 via tSNE and compares it to the tSNE plots of the feature space of the Coarse baseline, both on the coarse classes level and when digging into a random coarse class, qualitatively showing an advantage for ANCOR ’s feature space. In addition, in figure 7 we show a simple heatmap visualization of activations of the last convolutional layer of $\mathcal{B}$ obtained by dropping the GAP at the end of $\mathcal{B}$ and for each feature vector $f$ corresponding to a spatial coordinate in the resulting tensor computing the norm of the activation $||E(f)||^2$. To obtain a higher resolution for this visualization we also increase the input image resolution by $\times 2$. Surprisingly, despite the relatively small number of coarse classes (not benefiting spatial specificity), and the instance recognition nature of the self supervised objective (that could in principle use background pixels to discriminate instance images), the features learned by ANCOR trained $\mathcal{B}$ are remarkably good at localizing the object instances essentially ignoring the background. We verified this phenomena is stable and repeating in almost all localizations we examined and would be very interesting to explore in future studies.

### 4.5.5 Longer training

In all above experiments, for fair comparison we used 200 epochs for training all models (ANCOR, baselines and upper bounds). In Fig. 8 we explore what happens if we train longer. As can be seen, there is still much to be gained from ANCOR with longer training. We attribute this positive effect to the contrastive component that is known to benefit from longer training [2, 5]. Almost 15% gains are observed for ANCOR when increasing the number of epochs from 200 to 800, and interestingly, ANCOR ’s ‘all-way’ gain above the baselines becomes larger with more epochs.

### 5. Summary and conclusions

We have proposed the C2FS task focusing on situations when a few-shot model needs to adapt to much finer-grained unseen classes than the base classes used during its pre-training, including the challenging case when the unseen target classes are sub-classes of the base classes. We introduced the ANCOR approach for the C2FS task based on effective combination of inter-class supervised and intra-class self-supervised losses featuring a novel ‘angular normalization’ component inducing synergy between these (otherwise conflicting) losses. We have demonstrated the effectiveness of ANCOR on a variety of datasets showing: (i) promising results of ANCOR on the C2FS task also in the more challenging ‘all-way’ setting; (ii) that the proposed angular component is instrumental in the success of ANCOR; (iii) the advantages of ANCOR are preserved when adding more shots for the fine classes; (iv) ANCOR does best also in the challenging ‘intra-class fine’ scenario when all target classes belong to the same coarse class; (v) promising properties of ANCOR feature space including surprisingly good spatial attention to object instances; and (vi) ANCOR can be improved considerably with longer training, getting larger improvements on C2FS task than even leading self-supervised methods trained on more data ([5, 2]). We hope that this work will serve as a good basis for future research into the exciting and challenging C2FS task further pushing its limits. Finally, we believe that the proposed angular normalization component is useful beyond the C2FS task for any situation involving supervised and contrastive self-supervised multi-tasking, and leave it to future works to explore its uses further.

### 6. Appendix

#### 6.1. Additional baselines

We include 7 additional baseline comparisons using the four BREEDS [46] datasets. These baselines were not included in the main paper due to lack of space. The com-
ensemble and cascade combinations are as described in the paper. (1) ensemble: averaging the two models predictions (probabilities after softmax); (2) cascade: classify into the max-scoring coarse classes \(\mathcal{Y}_{\text{coarse}}\) using the coarse-supervised model (with the linear classifier, paper Sec. 3.3) for intra-class classification within the chosen coarse class (limiting the target classes to its sub-classes); (3) concat: concatenate the feature vectors from the two models and then apply few-shot classification as in the paper Sec. 3.3; (4) Same as Coarse+ using the Supervised Contrastive loss [23] replacing CE.

| Method                                | LIVING-17 5-way | LIVING-17 all-way | NONLIVING-26 5-way | NONLIVING-26 all-way | ENTITY-13 5-way | ENTITY-13 all-way | ENTITY-30 5-way | ENTITY-30 all-way |
|----------------------------------------|-----------------|-------------------|--------------------|----------------------|----------------|------------------|----------------|-----------------|
| ensemble(Coarse+, MoCoV2)             | 74.07 ± 0.67    | 34.02 ± 0.13      | 73.14 ± 0.71       | 33.23 ± 0.11         | 85.27 ± 0.57  | 36.39 ± 0.08     | 84.22 ± 0.06  | 34.56 ± 0.08    |
| ensemble(Coarse, MoCoV2)              | 84.32 ± 0.62    | 36.39 ± 0.13      | 79.95 ± 0.65       | 34.31 ± 0.11         | 85.97 ± 0.59  | 34.03 ± 0.08     | 88.48 ± 0.55  | 34.49 ± 0.08    |
| cascade(Coarse,MoCoV2)                | 74.1 ± 0.66     | 34.04 ± 0.13      | 73.1 ± 0.7         | 33.23 ± 0.11         | 85.19 ± 0.57  | 36.38 ± 0.08     | 84.33 ± 0.06  | 34.57 ± 0.08    |
| cascade(Coarse,MoCoV2)                | 88.27 ± 0.63    | 38.02 ± 0.14      | 84.66 ± 0.64       | 36.05 ± 0.11         | 86.39 ± 0.59  | 35.46 ± 0.07     | 90.02 ± 0.55  | 37.1 ± 0.08     |
| concat(Coarse+,MoCoV2)                | 73.48 ± 0.66    | 33.4 ± 0.13       | 71.49 ± 0.71       | 31.28 ± 0.11         | 84.26 ± 0.59  | 35.64 ± 0.08     | 83.03 ± 0.62  | 33.5 ± 0.07     |
| concat(Coarse,MoCoV2)                 | 83.45 ± 0.63    | 35.89 ± 0.13      | 77.59 ± 0.67       | 32.48 ± 0.11         | 85.35 ± 0.6   | 33.89 ± 0.07     | 86.83 ± 0.57  | 33.82 ± 0.07    |
| Supervised Contrastive [23]           | 84.69 ± 0.67    | 35.11 ± 0.12      | 84.54 ± 0.61       | 37.44 ± 0.11         | 87.08 ± 0.58  | 28.57 ± 0.15     | 88.86 ± 0.52  | 33.67 ± 0.17    |

ANCOR (ours)                           | 89.23 ± 0.55    | 45.14 ± 0.12      | 86.23 ± 0.54       | 43.10 ± 0.11         | 90.58 ± 0.54  | 42.29 ± 0.08     | 88.12 ± 0.54  | 41.79 ± 0.08    |

Table 7. Additional baselines: evaluated using BREEDS [46] with 1-shot. The "Coarse", "Coarse+", and "MoCoV2" models used in ensemble and cascade combinations are as described in the paper. (1) ensemble: averaging the two models predictions (probabilities after softmax); (2) cascade: classify into the max-scoring coarse classes \(\mathcal{Y}_{\text{coarse}}\) using the coarse-supervised model (with the linear classifier resulting from pre-training), and then use the self-supervised model (with the LR classifier, paper Sec. 3.3) for intra-class classification within the chosen coarse class (limiting the target classes to its sub-classes); (3) concat: concatenate the feature vectors from the two models and then apply few-shot classification as in the paper Sec. 3.3; (4) Same as Coarse+ using the Supervised Contrastive loss [23] replacing CE.

| Method     | Good | Bad | 5-way | all-way |
|------------|------|-----|-------|--------|
| Coarse+    | ✓    | 70.77 ± 0.74 | 36.65 ± 0.19 |
| Coarse+    | ✓    | 73.44 ± 0.69 | 43.13 ± 0.22 |
| ANCOR (ours)| ✓    | 74.99 ± 0.71 | 40.69 ± 0.20 |
| ANCOR (ours)| ✓    | 77.32 ± 0.69 | 47.53 ± 0.22 |

Table 8. Sub-population Shift. Two hand-crafted partitions of the LIVING-17 dataset, created by [46], such that the test sub-classes are different from the (unlabeled) training sub-classes, yet share the common coarse classes. ‘Good’ and ‘Bad’ represent a less and more adversarial partitioning. Note that in practice these models train on half the data the models trained on LIVING-17 in the main paper have, due to the partitioning.
partitioning respectively. We leverage this to create a task that is on one hand allows having the same coarse classes in training and in testing (as in the task in Section 4.4.1), while still having the sub-classes of those coarse classes non-overlapping (different) between the train and the test (as in the task in Section 4.4.2). In other words, in this scenario the test sub-classes are completely different from the training ones, and yet they share the same parent coarse classes. Our evaluation on this task, provided in Table 8, shows that despite this challenging setting, ANCOR significantly outperforms the strongest baseline by > 4% in the all-way test.

### 6.3. Additional results for 800-epoch training

In this section we further examine the effect of longer training longer on the performance. As can be seen in Table 9, following longer 800 epoch training ANCOR obtains significant gains in all the experimental settings. The most noticeable gains are in the all-way tests, where we observe that the gap above the baselines grows with longer training. We attribute this improvement to the contrastive component that is known to benefit from longer training [2, 5]. Interestingly, with longer training the coarse baseline models have gained accuracy in the 5-way test, but lost accuracy in the all-way test (compared to the 200 epochs performance). This supports our hypothesis that the coarse-classes supervised objective encourages reducing intra-class variation, and as such, with the longer training, tends to decrease the distinction between fine-sub classes loosing their discriminability within the coarse class. This again underlines the merits of our ANCOR approach that retains and enhances the fine sub-classes discriminability thus significantly benefiting from the longer training regime.

### 6.4. Additional examples of ANCOR encoder last layer activations

Additional examples of ANCOR encoder last layer activations are provided in Figure 9, again illustrating an interesting attention to objects learned by ANCOR despite not being provided with any location supervision during training.

#### Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. FA8750-19-C-1001. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA. Raja Giryes was supported by ERC-StG grant no. 757497 (SPADE).

#### References

[1] Antreas Antoniou and Amos Storkey. Assume, Augment and Learn: Unsupervised Few-Shot Meta-Learning via Random Labels and Data Augmentation. Technical report. 3

[2] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. 6 2020. 2, 3, 6, 8, 10

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. ICML, 2020. 2, 3

[4] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang. A Closer Look At Few-Shot Classification. In ICLR, 2019. 3

[5] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved Baselines with Momentum Contrastive Learning. arXiv, 3 2020. 3, 6, 8, 10

[6] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved Baselines with Momentum Contrastive Learning. arXiv, 3 2020. 5

[7] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised Visual Representation Learning by Context Prediction. In ICCV, 2015. 3

[8] Sivan Doehv, Eli Schwartz, Chao Xue, Rogerio Feris, Alex Bronstein, Raja Giryes, and Leonid Karlinsky. MetaAdapt: Meta-Learned Task-Adaptive Architecture for Few-Shot Classification. Technical report, 2019. 3

[9] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with Cooperation: Ensemble Methods for Few-Shot Classification. In ICCV, 2019. 3

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In ICML, 2017. 3

---

**Table 9. Results for different baselines on the four BREEDS datasets. Every model was trained for 800 epochs.**

| Method                  | Method                  | Method                  | Method                  |
|------------------------|------------------------|------------------------|------------------------|
| LIVING-17              | NONLIVING-26           | ENTITY-13              | ENTITY-30              |
| 5-way                  | 5-way                  | 5-way                  | 5-way                  |
| 6-way                  | all-way                | all-way                | all-way                |
| Fine (upper-bound)     | MoCoV2                 | MoCoV2-ImageNet [5]    | SWAV-ImageNet [2]      |
| 91.94 ± 0.44 64.66 ± 0.17 | 81.45 ± 0.61 46.65 ± 0.16 | 89.27 ± 0.57 51.60 ± 0.15 | 80.11 ± 0.63 39.30 ± 0.14 |
| Fine + (upper-bound)   | MoCoV2                 | MoCoV2-ImageNet [5]    | SWAV-ImageNet [2]      |
| 90.25 ± 0.48 63.54 ± 0.17 | 86.27 ± 0.51 53.16 ± 0.14 | 81.90 ± 0.40 59.43 ± 0.09 | 91.03 ± 0.43 57.48 ± 0.09 |
| Coarse                 | Coarse                 | Coarse                 | Coarse                 |
| 89.04 ± 0.63 29.06 ± 0.23 | 84.72 ± 0.63 27.99 ± 0.18 | 82.66 ± 0.75 11.24 ± 0.08 | 90.09 ± 0.59 20.63 ± 0.12 |
| Coarse +               | Coarse +               | Coarse +               | Coarse +               |
| 89.41 ± 0.61 33.07 ± 0.23 | 84.69 ± 0.59 32.07 ± 0.20 | 85.23 ± 0.63 22.85 ± 0.13 | 88.43 ± 0.55 28.33 ± 0.15 |
| ANCOR (ours)           | ANCOR (ours)           | ANCOR (ours)           | ANCOR (ours)           |
| 92.59 ± 0.47 58.15 ± 0.16 | 88.25 ± 0.52 49.38 ± 0.13 | 92.04 ± 0.44 50.72 ± 0.09 | 92.13 ± 0.44 50.85 ± 0.09 |

Table 9. Results for different baselines on the four BREEDS datasets. Every model was trained for 800 epochs.
Figure 9. Additional examples of ANCOR encoder $B$ last layer activations.

[11] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting Few-Shot Visual Learning with Self-Supervision. In ICCV, 6 2019.

[12] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised Representation Learning by Predicting Image Rotations. In ICLR. arXiv, 3 2018.

[13] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheslaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised Learning. 6 2020.

[14] Yunhui Guo, Noel C. Codella, Leonid Karlinsky, James V. Codella, John R. Smith, Kate Saenko, Tajana Rosing, and Rogerio Feris. A Broader Study of Cross-Domain Few-Shot Learning. In ECCV, 12 2020.
[48] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Delta-Encoder: an Effective Sample Synthesis Method for Few-Shot Object Recognition. *NeurIPS*, 2018. 3

[49] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical Networks for Few-shot Learning. In *NIPS*, 2017. 1, 3

[50] Jong-Chyi Su, Subhransu Maji, and Bharath Hariharan. When Does Self-supervision Improve Few-shot Learning? *ECCV*, 2020. 3

[51] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to Compare: Relation Network for Few-Shot Learning. In *CVPR*, pages 1199–1208, 2018. 3

[52] Fariborz Taherkhani, Hadi Kazemi, Ali Dabouei, Jeremy Dawson, and Nasser Nasrabadi. A weakly supervised fine label classifier enhanced by coarse supervision. Technical report, 2019. 3

[53] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *ICML*, volume 2019-June, pages 10691–10700. International Machine Learning Society (IMLS), 2019. 1, 3

[54] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive Multiview Coding. 2019. 3

[55] 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*, 3 2020. 1, 3, 4, 5, 6, 7

[56] Aaron Van Den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. 7 2018. 3

[57] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching Networks for One Shot Learning. *NIPS*, 2016. 1, 3

[58] Yan Wang, Wei-Lun Chao, Kilian Q. Weinberger, and Laurens van der Maaten. SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning. 2019. 3, 4

[59] Tete Xiao, Xiaolong Wang, Alexei A. Efros, and Trevor Darrell. What Should Not Be Contrastive in Contrastive Learning. 8 2020. 3

[60] Chen Xing, Negar Rostamzadeh, Boris N Oreshkin, and Pedro O Pinheiro. Adaptive Cross-Modal Few-Shot Learning. In *NeurIPS*, 2019. 3

[61] Jinhai Yang, Hua Yang, and Lin Chen. Coarse-to-Fine Pseudo-Labeling Guided Meta-Learning for Few-Shot Classification. Technical report. 3, 5, 7

[62] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. DeepEMD: Few-Shot Image Classification with Differentiable Earth Mover’s Distance and Structured Classifiers. In *CVPR*, 2020. 3, 7

[63] Hongguang Zhang, Jing Zhang, and Piotr Koniusz. Few-shot learning via saliency-guided hallucination of samples. *CVPR*, 2019-June:2765–2774, 2019. 3

[64] Jian Zhang, Chenglong Zhao, Bingbing Ni, Minghao Xu, and Xiaokang Yang. Variational Few-Shot Learning. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 3

[65] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful Image Colorization. In *ECCV*, volume 9907 LNCS, pages 649–666. Springer Verlag, 2016. 3

[66] Hengshuang Zhao, Cuhk Jiaya, Jia Cuhk, and Vladlen Koltun. Exploring Self-attention for Image Recognition. In *CVPR*, 2020. 1

[67] Fengwei Zhou, Bin Wu, and Zhenguo Li. Deep Meta-Learning: Learning to Learn in the Concept Space. Technical report, 2 2018. 3