UNSUPERVISED FEW-SHOT OBJECT RECOGNITION BY INTEGRATING ADVERSARIAL, SELF-SUPERVISION, AND DEEP METRIC LEARNING OF LATENT PARTS

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper addresses unsupervised few-shot object recognition, where all training images are unlabeled and do not share classes with labeled support images for few-shot recognition in testing. We use a new GAN-like deep architecture aimed at unsupervised learning of an image representation which will encode latent object parts and thus generalize well to unseen classes in our few-shot recognition task. Our unsupervised training integrates adversarial, self-supervision, and deep metric learning. We make two contributions. First, we extend the vanilla GAN with reconstruction loss to enforce the discriminator capture the most relevant characteristics of "fake" images generated from randomly sampled codes. Second, we compile a training set of triplet image examples for estimating the triplet loss in metric learning by using an image masking procedure suitably designed to identify latent object parts. Hence, metric learning ensures that the deep representation of images showing similar object classes which share some parts are closer than the representations of images which do not have common parts. Our results show that we significantly outperform the state of the art, as well as get similar performance to the common episodic training for fully-supervised few-shot learning on the Mini-Imagenet and Tiered-Imagenet datasets.

1 INTRODUCTION

This paper presents a new deep architecture for unsupervised few-shot object recognition. In training, we are given a set of unlabeled images. In testing, we are given a small number K of *support images* with labels sampled from N object classes that do not appear in the training set (also referred to as unseen classes). Our goal in testing is to predict the label of a *query image* as one of these N previously unseen classes. A common approach to this N-way K-shot recognition problem is to take the label of the closest support to the query. Thus, our key challenge is to learn a deep image representation on unlabeled data such that it would in testing generalize well to unseen classes, so as to enable accurate distance estimation between the query and support images.

Our unsupervised few-shot recognition problem is different from the standard few-shot learning (Snell et al., 2017; Finn et al., 2017), as the latter requires labeled training images (e.g., for episodic training (Vinyals et al., 2016)). Also, our problem is different from the standard semi-supervised learning (Chapelle et al., 2009), where both unlabeled and labeled data are typically allowed to share either all or a subset of classes. When classes of unlabeled and labeled data are different in semi-supervised learning (Chapelle et al., 2009), the labeled dataset is typically large enough to allow transfer learning of knowledge from unlabeled to labeled data, which is not the case in our few-shot setting.

There is scant work on unsupervised few-shot recognition. The state of the art (Hsu et al., 2018) first applies unsupervised clustering (Caron et al., 2018) for learning pseudo labels of unlabeled training images, and then uses the standard few-shot learning on these pseudo labels for episodic training – e.g., Prototypical Network (Snell et al., 2017) or MAML (Finn et al., 2017). However, performance of this method is significantly below that of counterpart approaches to supervised few-shot learning.

Our approach is aimed at learning an image representation from unlabeled data that captures presence or absence of latent object parts. We expect that such a representation would generalize well

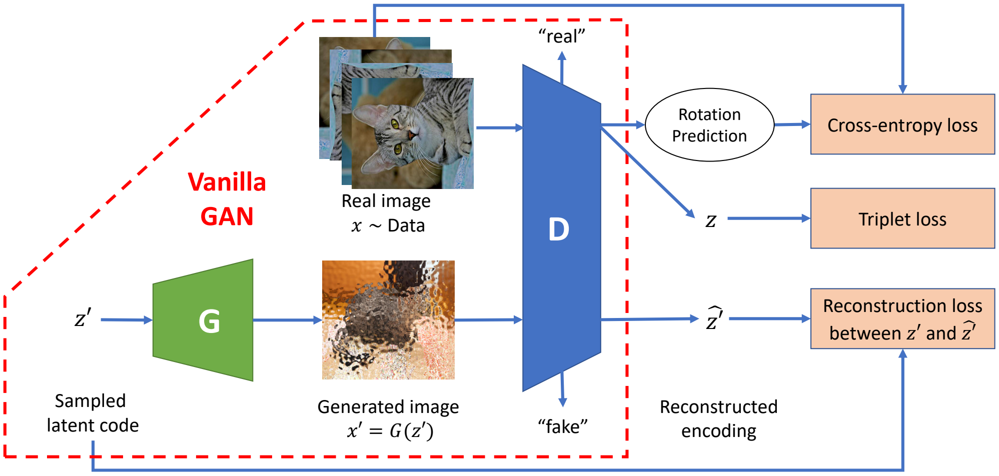

Figure 1: We use a GAN-like deep architecture to learn an image encoding z on unlabeled training data that will be suitable for few-shot recognition in testing. Our unsupervised training integrates adversarial, self-supervision, and metric learning. The figure illustrates our first contribution that extends the vanilla GAN (the red dashed line) with regularization so the encoding $\hat{z}'$ of a "fake" image is similar to the randomly sampled code z' which has been used for generating the "fake" image. The self-supervision task is to predict the rotation angle of rotated real training images. Deep metric learning is illustrated in greater detail in Fig. 3.

to unseen classes in our few-shot recognition task. This is because of the common assumption in computer vision that various distinct object classes share certain parts. Thus, while our labeled and unlabeled images do not show the same object classes, there may be some parts that appear in both training and test image sets. Therefore, an image representation that would capture presence of these common parts in unlabeled images is expected to also be suitable for representing unseen classes, and thus facilitate our N-way K-shot recognition.

Toward learning such an image representation, in our unsupervised training, we integrate adversarial, self-supervision, and deep metric learning. As shown in Fig. 1, we use a GAN-like architecture for training a discriminator network D to encode real images x in their d-dimensional deep representations $z = D_z(x) \in [-1, 1]^d$, which will be later used for few-shot recognition in testing. We also consider a discrete encoding $z = D_z(x) \in \{-1, 1\}^d$, and empirically discover that it gives better performance than the continuous counterpart. Hence our interpretation that binary values in the discrete z indicate presence or absence of d latent parts in images.

In addition to D_z, the discriminator has two other outputs (i.e., heads), $D_{r/f}$ and D_{rot}, for adversarial and self-supervised learning, respectively as illustrated in Fig. 2. D is adversarially trained to distinguish between real and "fake" images, where the latter x' are produced by a generator network G, $x' = G(z')$, from image encodings z' which are randomly sampled from the uniform distribution $U[-1, 1]^d$. Sampling from the uniform distribution is justified, because latent parts shared among a variety of object classes appearing in the unlabeled training set are likely to be uniformly distributed across the training set. We extend the vanilla GAN with regularization aimed at minimizing a reconstruction loss between the sampled z' and the corresponding embedding $\hat{z}' = D(G(z'))$. As our experiments demonstrate, this reconstruction loss plays an important role in training both D and G in combination with the adversarial loss, as both losses enforce G generate as realistic images as possible and D capture the most relevant image characteristics for reconstruction and real/fake recognition.

Furthermore, following recent advances in self-supervised learning (Doersch et al., 2015; Zhang et al., 2016; Noroozi & Favaro, 2016; Noroozi et al., 2017; Zhang et al., 2017), we also augment our training set with rotated versions of the real images around their center, and train D to predict their rotation angles, $\hat{\alpha} = D_{rot}(Rotate(x, \alpha)) \in \{0, 1, 2, 3\} * 90°$. As in other approaches that

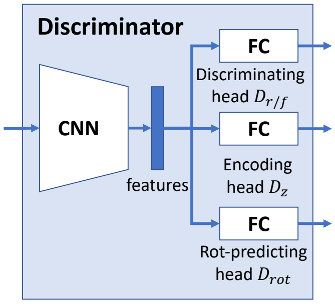

Figure 2: Details of the discriminator from Fig. 1 with three heads: discriminating head, encoding head and rotation-prediction head.

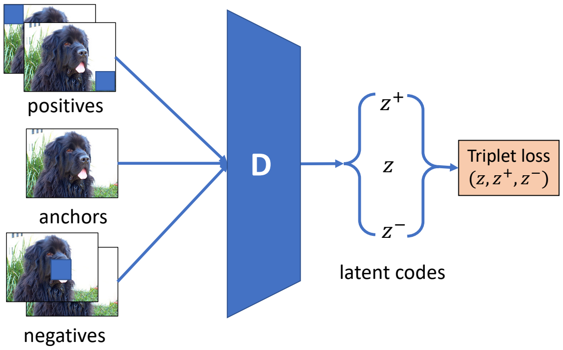

Figure 3: Our second contribution is to use metric learning of the discriminator shown in Figures 1, 2. We compile a set of training triplets $\langle anchor, positive, negative \rangle$ for estimating the standard triplet loss.

use self-supervised learning, our results demonstrate that this data augmentation strengthens our unsupervised training and improves few-shot recognition.

Finally, we use deep metric learning toward making the image encoding $z = D_z(x)$ represent latent parts and in this way better capture similarity of object classes for our few-shot recognition. We expect that various object classes share parts, and that more similar classes have more common parts. Therefore, the encodings of images showing similar (or different) object classes should have a small (or large) distance. To ensure this property, we use metric learning and compile a new training set of triplet images for estimating the standard triple loss, as illustrated in Fig. 3. Since classes in our training set are not annotated, we form the triplet training examples by using an image masking procedure which is particularly suitable for identifying latent object parts. In the triplet, the anchor is the original (unmasked) image, the positive is an image obtained from the original by masking rectangular patches at the image periphery (e.g., top corner), and the negative is an image obtained from the original by masking centrally located image patches. By design, the negative image masks an important object part, and thus the deep representations of the anchor and the negative should have a large distance. Conversely, masking peripheral corners in the positive image does not cover any important parts of the object, and thus the deep representation of the positive should be very close to that of the anchor. In this way, our metric learning on the triplet training examples ensures that the learned image representation z accounts for similarity of object classes in terms of their shared latent parts. As our results show, this component of our unsupervised training further improves few-shot recognition in testing, to the extent that not only do we significantly outperform the state of the art but also get a performance that is on par with the common episodic training for fully-supervised few-shot learning on the Mini-Imagenet (Vinyals et al., 2016; Ravi & Larochelle, 2016) and Tiered-Imagenet (Ren et al., 2018) datasets.

Our contributions are twofold:

- Extending the vanilla GAN with a reconstruction loss between uniformly sampled codes, $z' \sim U[-1, 1]^d$, and embeddings of the corresponding "fake" images, $\hat{z}' = D(G(z'))$.
- The masking procedure for compiling triplet image examples and deep metric learning of z so it accounts for image similarity in terms of shared latent parts.

The rest of this paper is organized as follows. Sec. 2 reviews previous work, Sec. 3 specifies our proposed approach, Sec. 4 presents our implementation details and our experimental results, and finally, Sec. 5 gives our concluding remarks.

2 Related Work

This section reviews the related work on few-shot learning including standard, semi-supervised and unsupervised few-shot learning. Few-shot learning is a type of transfer learning, where the goal is to transfer knowledge learned from a training set to the test set such that the model can recognize new classes from a few examples (Miller et al., 2000; Fe-Fei et al., 2003).

Approaches to supervised few-shot learning can be broadly divided into three main groups based on metric learning, meta-learning, and hallucination. Metric-learning based approaches seek to learn embeddings such that they are close for same-class examples and far away for others. Representative methods include Matching networks (Vinyals et al., 2016), Prototypical networks (Snell et al., 2017) and Relation networks (Sung et al., 2018). Meta-learning based approaches learn a meta-learner for learning a task-adaptive learner such that the latter performs well on new classes by parameter fine-tuning. Representative methods include MAML (Finn et al., 2017), Reptile (Nichol et al., 2018), and many others (Gidaris & Komodakis, 2018; Sun et al., 2019b; Jamal & Qi, 2019). Finally, hallucination based few-shot learning first identifies rules for data augmentation from the training set. These rules are then used in testing to generate additional labeled examples for few-shot recognition. Representative methods include Imaginary network (Wang et al., 2018), f-VEAGAN-D2 (Xian et al., 2019) and Delta-encoder (Schwartz et al., 2018).

Semi-supervised few-shot learning was introduced in (Ren et al., 2018), and further studied in (Sun et al., 2019a). These approaches augment the labeled training set with unlabeled images.

Hsu et al. (2018) introduced the unsupervised few-shot learning problem, where the entire training set is unlabeled. They first create pseudo labels from unsupervised training, then apply the standard supervised few-shot learning on these pseudo labels of training examples. While Hsu et al. (2018) use clustering to identify pseudo labels, Antoniou & Storkey (2019) and Khodadadeh et al. (2018) treat each training example as belonging to a unique class.

We differ from the above closely related approaches in two ways. First, we do not use the common episodic training for few-shot learning. Second, we ensure that our image representation respects distance relationships between dissimilar images when their important parts are masked.

3 Our Approach

Our training set consists of unlabeled examples x_u with hidden classes $y_u \in L_{train}$. In testing, we are given support images x_s with labels $y_s \in L_{test}$ sampled from $N = |L_{test}|$ unseen classes, $L_{train} \cap L_{test} = \emptyset$, where each unseen class has K examples. Our N-way K-shot task is to classify query images x_q into one of these N classes, $y_q \in L_{test}$. For this, we first compute deep image representations $z_q = D_z(x_q)$ and $z_s = D_z(x_s)$ of the query and support images using the discriminator of the deep architecture shown in Fig. 1. Then, for every unseen class $n = 1, \ldots, N$, we compute the prototype vector c_n as the mean of the K image encodings $z_s = D_z(x_s)$ of class n:

$$c_n = \frac{1}{K} \sum_{\substack{x_s \\ y_s = n}} D_z(x_s). \tag{1}$$

Finally, we take the label of the closest c_n to z_q as our solution:

$$\hat{y}_q = \hat{n} = \arg\min_n \Delta(z_q, c_n), \tag{2}$$

where Δ denotes a distance function, specified in Sec. 3.4. The same formulation of few-shot recognition is used in (Snell et al., 2017).

Our deep architecture consists of a generator G and a discriminator D networks, which are learned by integrating adversarial, self-supervision and metric learning. To this end, we equip D with three output heads: image encoding head D_z, rotation prediction head D_{rot} for self-supervision, and the standard discriminating head $D_{r/f}$ for distinguishing between real and "fake" images in adversarial training, as depicted in Fig. 3.

3.1 Adversarial Learning

We specify the adversarial loss functions for training D and G as

$$L_D^{\text{adv}} = \mathop{\mathrm{E}}_{x \sim p_{\text{data}}(x)} [\min(0, -1 + D_{r/f}(x))] + \mathop{\mathrm{E}}_{z \sim p(z)} [\min(0, -1 - D_{r/f}(G(z)))], \tag{3}$$

$$L_G^{\text{adv}} = - \mathop{\mathrm{E}}_{z \sim p(z)} [D_{r/f}(G(z))], \tag{4}$$

where E denotes the expected value, $p_{\mathrm{data}}(x)$ is a distribution of the unlabeled training images, and $p(z)$ is a distribution of latent codes which are sampled for generating "fake" images. In our experiments, we have studied several specifications for $p(z)$ aimed at modeling occurrences of latent parts across images, including the binomial distribution $\mathrm{Bin}(0.5)$, the Gaussian distribution $\mathcal{N}(0,1)$, and the uniform distribution $\mathrm{U}[-1,1]$. For all these specifications, we get similar performance. As shown in Lim & Ye (2017), optimizing the objectives in equation 3 and equation 4 is equivalent to minimizing the reverse KL divergence.

3.2 Self-Supervised Learning

For self-supervision, we rotate real images of the unlabeled training set around their center, and train D to predict the rotation angle α using the following cross-entropy loss:

$$L_D^{\text{rot}} = -\frac{1}{4}\sum_{\alpha=0}^{3}[\alpha * \log D_{rot}(\tilde{x}_\alpha)], \qquad \tilde{x}_\alpha = Rotate(x,\alpha), \tag{5}$$

where $\tilde{x}_\alpha$ is the rotated version of x with angle $\alpha \in \{0,1,2,3\} * 90^\circ$. We are aware that there are many other ways to incorporate self-supervision (e.g., "jigsaw solver" (Noroozi & Favaro, 2016)). We choose image rotation for its simplicity and ease of implementation, as well as state-of-the-art performance reported in the literature.

3.3 Regularization by Reconstruction of Latent Codes

We extend the vanilla GAN by making D reconstruct the probabilistically sampled latent code $z' \sim p(z)$, which is passed to G to generate synthetic images. Thus, we use z' as a "free" label for additionally training of D and G along with the adversarial and self-supervision learning. The reconstruction loss is specified as the binary cross-entropy loss

$$L_D^{\text{bce}} = L_G^{\text{bce}} = -\frac{1}{d}\sum_{m=1}^{d}[z'_m * \log\sigma(\widehat{z'_m}) + (1-z'_m) * \log(1-\sigma(\widehat{z'_m})], \qquad \hat{z'} = D_z(G(z'))), \tag{6}$$

where z' is converted to range $[0,1]^d$ for computing loss, d is the length of z', z'_m is the mth element of the latent code, $\widehat{z'_m}$ is the predicted mth value of the discriminator's encoding head D_z, and $\sigma(\cdot)$ is the sigmoid function.

3.4 Deep Metric Learning

We additionally train D to output image representations that respect distance relationships such that the more latent parts are shared between images, the closer their representations. To this end, we compile a training set of triplets $\langle anchor, positive, negative\rangle$. The anchor $z = D_z(x)$ represents an original image x from the unlabeled training set. The positives $\{z_i^+ : i = 1,\ldots,4\}$) represent four images, $z_i^+ = D_z(x_i^+)$, obtained by masking one of the four corners of the anchor image: top-left, top-right, bottom-left, and bottom-right. The masking patch is selected to be relatively small and thus ensure that no to little foreground is masked in the positives. The negatives $\{z_j^- : j = 1,2,\ldots\}$ represent images, $z_j^- = D_z(x_j^-)$, obtained by placing a masking patch over central locations in the anchor image so as to ensure covering foreground parts. Given the training set of triplets $\{\langle z, z_i^+, z_j^-\rangle\}$, we specify the triplet loss for deep metric learning as

$$L_D^{\text{triplet}} = \max[0, \min_j \Delta(z, z_j^-) - \max_i \Delta(z, z_i^+) + \rho], \tag{7}$$

where ρ is a distance margin, and Δ is the following distance function:

$$\Delta(z, z') = 1 - \frac{z^\top z'}{\|z\|_2 \cdot \|z'\|_2}. \tag{8}$$

3.5 Our Unsupervised Training

Alg. 1 summarizes our unsupervised training that integrates adversarial, self-supervision and deep metric learning. For easier training of D and G, we divide learning in two stages. First, we perform the adversarial and self-supervision training by following the standard GAN training, where for each image sampled from the training set, $t_1 = 1, \dots, T_1$, G is optimized once and D is optimized multiple times over $t_2 = 1, \dots, T_2$ iterations ($T_2 = 3$). After convergence of the first training stage ($T_1 = 50,000$), the resulting discriminator is saved and denoted as $D^{(1)}$. In the second training stage, we continue with metric learning of D over the triplet image examples in $t_3 = 1, \dots, T_3$ iterations ($T_3 = 20,000$), while simultaneously regularizing that the discriminator updates do not significantly deviate from the previously learned $D^{(1)}$.

Algorithm 1: Our unsupervised training consists of two stages. T_1 is the number of training iterations of the first stage aimed at adversarial and self-supervision learning; T_2 is the number of updates of D per one update of G in the first training stage; T_3 is the number of training iterations in the second stage aimed at metric learning. $\beta, \gamma, \delta, \lambda$ are non-negative hyper parameters.

```
                                                                        ▷ First stage (1)
for $t_1 = 1, \dots, T_1$ do
    Sample the latent code $z' \sim p(z)$;
    Generate the corresponding “fake” image $x' = G(z')$, and compute $\hat{z'} = D_z(x')$;
    Compute: $L_G^{\text{adv}}$ as in equation 4, and $L_G^{\text{bce}}$ as in equation 6;
    Back-propagate the total loss $L_G^{(1)} = L_G^{\text{adv}} + \beta L_G^{\text{bce}}$ to update $G$.
    for $t_2 = 1, \dots, T_2$ do
        Sample the latent code $z' \sim p(z)$, generate $x' = G(z')$, and compute $\hat{z'} = D_z(x')$;
        Randomly sample a real training image $x \sim p_{\text{data}}(x)$;
        Compute: $L_D^{\text{adv}}$ as in equation 3, $L_D^{\text{rot}}$ as in equation 5, $L_D^{\text{bce}}$ as in equation 6;
        Back-propagate the total loss $L_D^{(1)} = L_D^{\text{adv}} + \delta L_D^{\text{rot}} + \gamma L_D^{\text{bce}}$ to update $D^{(1)}$.
    end for
end for
                                                                        ▷ Second stage (2)
for $t_3 = 1, \dots, T_3$ do
    Randomly sample a real training image $x \sim p_{\text{data}}(x)$ and take it as anchor;
    Generate the positive and negative images by appropriately masking the anchor;
    Form the corresponding triplet examples;
    Compute: $L_D^{\text{triplet}}$ as in equation 7;
    Back-propagate the total loss $L_D^{(2)} = L_D^{\text{triplet}} + \lambda \| D_z^{(1)}(x) - D_z^{(2)}(x) \|_2^2$ to update $D^{(2)}$.
end for
Take $D^{(2)}$ as the learned discriminator $D$.
```

4 Results

Datasets: We evaluate our approach on the two common few-shot learning datasets: Mini-Imagenet (Vinyals et al., 2016; Ravi & Larochelle, 2016) and Tiered-Imagenet (Ren et al., 2018). *Mini-Imagenet* contains 100 randomly chosen classes from ILSVRC-2012 (Russakovsky et al., 2015). We split these 100 classes into 64, 16 and 20 classes for meta-training, meta-validation, and meta-testing respectively. Each class contains 600 images of size 84 × 84.

Tiered-Imagenet is a larger subset of ILSVRC-2012 (Russakovsky et al., 2015), consists of 608 classes grouped into 34 high-level categories. These are divided into 20, 6 and 8 categories for meta-training, meta-validation, for meta-testing. This corresponds to 351, 97 and 160 classes for meta-training, meta-validation, and meta-testing respectively. This dataset aims to minimize the semantic similarity between the splits as in Mini-Imagenet. All images are also of size 84 × 84.

We are the first to report results of unsupervised few-shot recognition on the Tiered-Imagenet dataset.

For our unsupervised few-shot problem, we ignore all ground-truth labeling information in the training and validation sets, and only use ground-truth labels of the test set for evaluation. We also resize all images to size 64×64 in order to match the required input size of the GAN. For hyper-parameter tuning, we use the validation loss of corresponding ablations.

Evaluation metrics: We first randomly sample N classes from the test classes and K examples for each sampled class, and then classify query images into these N classes. We report the average accuracy over 1000 episodes with 95% confidence intervals of the N-way K-shot classification.

Implementation details: We implement our approach, and conduct all experiments in Pytorch (Paszke et al., 2017). The backbone GAN that we use is the Spectral Norm GAN (SN-GAN) (Miyato et al., 2018) combined with the self-modulated batch normalization (Chen et al., 2018). The number of blocks of layers in both G and D is 4. The dimension of the latent code/representation z is $d = 128$. We use an Adam optimizer (Kingma & Ba, 2014) with the constant learning rate of $5e^{-4}$. D is updated in $T_2 = 3$ iterations for every update of G. In the first and the second training stages, the mini-batch size is 128 and 32, respectively. The latter is smaller, since we have to enumerate 16 masked images at 16 locations of a 4×4 grid for each original training image. That is, our image masking for generating positive and negative images is performed by placing a 16×16 patch centered at the 4×4 grid locations in the original image, where the patch brightness is equal to the average of image pixels. We empirically observe convergence of the first and second training stages of our full approach after $T_1 = 50000$ and $T_3 = 20000$ iterations, respectively. In all experiments, we set $\gamma = 1, \beta = 1, \delta = 1, \lambda = 0.2, \rho = 0.5$ as they are empirically found to give the best performance. It is worth noting that beyond generating data for self-supervision and metric learning, we do not employ the recently popular data-augmentation techniques in training (e.g., image jittering, random crop, etc.).

Ablations: We define the following simpler variants of our approach for testing how its individual components affect performance. The variants include:

- GAN: the Spectral Norm GAN (SN-GAN) (Miyato et al., 2018) with self-modulated batch normalization (Chen et al., 2018), as shown in Fig. 1 within the red dashed line.
- GAN + BCE: extends training of the GAN with the reconstruction loss.
- GAN + BCE + ROT: extends training of the GAN + BCE with the rotation prediction loss.
- GAN + BCE + ROT + METRIC: Our full model that extends the GAN + BCE + ROT with the triplet loss.

Ablation study and comparison with the state of the art: Table. 1 presents results of our ablations and a comparison with the state-of-the-art methods on Mini-Imagenet and Tiered-Imagenet, in 1-shot and 5-shot testing. For fair comparison, we follow the standard algorithm for assigning labels to query images in the 1-shot and 5-shot testing, as used in (Snell et al., 2017).

From Table. 1, our reconstruction loss plays a crucial role, since it improves performance of GAN + BCE by nearly 9% relative to that of GAN. Importantly, our ablation GAN + BCE already outperforms all related work by a large margin. This suggests that using a simple reconstruction loss improves training of the vanilla GAN. Adding the rotation loss further improves performance of GAN + BCE + ROT by 1%. Finally, the proposed triplet loss in GAN + BCE + ROT + METRIC gives an additional performance gain of 3%, and the state-of-the-art results. Interestingly, in one-shot setting, our full approach GAN + BCE + ROT + METRIC also outperforms the recent fully supervised approach of ProtoNets (Snell et al., 2017) trained on the labeled training set.

Qualitative Results: Fig. 4 illustrates our masking procedure for generating negative images in the triplets for metric learning. In each row, the images are organized from left to right by their estimated distance to the original (unmasked) image in the descending order, where the rightmost image is the closest. From Fig. 4, our metric learning ensures that the image representation captures important object parts, so when such parts are missing in the masked images their distances to the original image are greater than distances of other masked images missing less-important parts.

5 Conclusion

We have addressed unsupervised few-shot object recognition, where all training images are unlabeled and do not share classes with test images. A new GAN-like deep architecture has been

Table 1: Unsupervised few-shot recognition results on Mini-Imagenet and Tiered-Imagenet. We also compare with a recent fully-supervised method trained with ground-truth training labels for few-shot recognition.

	Mini-Imagenet, 5-way		**Tiered-Imagenet, 5-way**	
Unsupervised Methods	1-shot	5-shot	1-shot	5-shot
BiGAN kNN (Donahue et al., 2016)	25.56 ± 1.08	31.10 ± 0.63	-	-
AAL-ProtoNets (Antoniou & Storkey, 2019)	37.67 ± 0.39	40.29 ± 0.68	-	-
UMTRA + AutoAugment (Khodadadeh et al., 2018)	39.93	50.73	-	-
DeepCluster CACTUs -ProtoNets (Hsu et al., 2018)	39.18 ± 0.71	53.36 ± 0.70	-	-
GAN only	34.84 ± 0.68	44.73 ± 0.67	35.57 ± 0.69	49.16 ± 0.70
GAN + BCE	43.51 ± 0.77	57.94 ± 0.76	43.82 ± 0.76	59.22 ± 0.75
GAN + BCE + ROT	44.43 ± 0.78	58.96 ± 0.72	44.80 ± 0.75	61.94 ± 0.75
GAN + BCE + ROT + METRIC	**47.40** ± 0.78	**61.63** ± 0.72	**47.48** ± 0.78	**64.39** ± 0.74
Fully-supervised Method:				
ProtoNets (Snell et al., 2017)	46.56 ± 0.76	62.29 ± 0.71	46.52 ± 0.72	66.15 ± 0.74

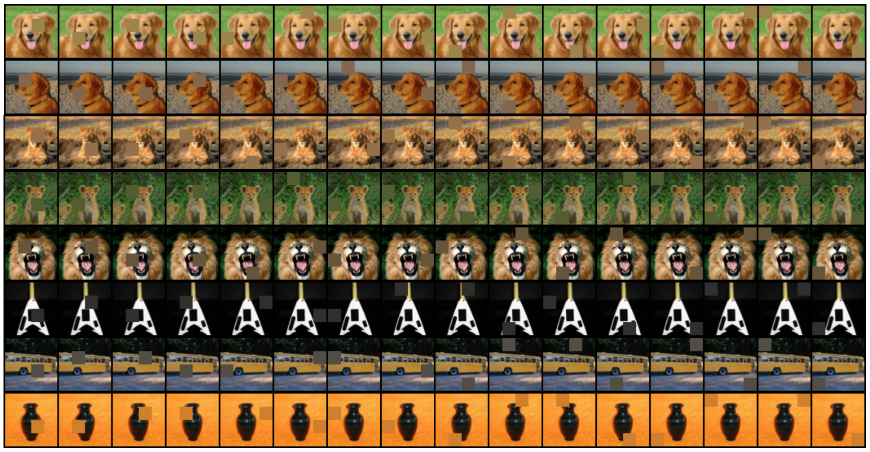

Figure 4: Our image masking with rectangular patches for Mini-Imagenet. In every row, the images are organized from left to right in the descending order by their estimated distance to the original (unmasked) image.

proposed for unsupervised learning of an image representation which respects image similarity in terms of shared latent object parts. We have made two contributions by extending the vanilla GAN with reconstruction loss and by integrating deep metric learning with the standard adversarial and self-supervision learning. Our results demonstrate that our approach generalizes will to unseen classes, outperforming the sate of the art by more than 8% in both 1-shot and 5-shot recognition tasks on the benchmark Mini-Imagenet dataset. We have reported the first results of unsupervised few-shot recognition on the Tiered-Imagenet dataset. Our ablations have evaluated that solely our first contribution leads to superior performance relative to that of closely related approaches, and that the addition of the second contribution further improves our 1-shot and 5-shot recognition by 3%. We also outperform a recent fully-supervised approach to few-shot learning that uses the common episodic training on the same datasets.

References

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.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149, 2018.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

Ting Chen, Mario Lucic, Neil Houlsby, and Sylvain Gelly. On self modulation for generative adversarial networks. *arXiv preprint arXiv:1810.01365*, 2018.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1422–1430, 2015.

Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

Li Fe-Fei et al. A bayesian approach to unsupervised one-shot learning of object categories. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 1134–1141. IEEE, 2003.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1126–1135. JMLR. org, 2017.

Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4367–4375, 2018.

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

Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11719–11727, 2019.

Siavash Khodadadeh, Ladislau Bölöni, and Mubarak Shah. Unsupervised meta-learning for few-shot image and video classification. *arXiv preprint arXiv:1811.11819*, 2018.

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

Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.

Erik G Miller, Nicholas E Matsakis, and Paul A Viola. Learning from one example through shared densities on transforms. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 1, pp. 464–471. IEEE, 2000.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pp. 69–84. Springer, 2016.

Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5898–5906, 2017.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. *NIPS-W*, 2017.

Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. *International Conference on Learning Representation*, 2016.

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.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, Rogerio Feris, Raja Giryes, and Alex Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. In *Advances in Neural Information Processing Systems*, pp. 2845–2855, 2018.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pp. 4077–4087, 2017.

Qianru Sun, Xinzhe Li, Yaoyao Liu, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. *arXiv preprint arXiv:1906.00562*, 2019a.

Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 403–412, 2019b.

Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2018.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pp. 3630–3638, 2016.

Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7278–7286, 2018.

Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. f-vaegan-d2: A feature generating framework for any-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10275–10284, 2019.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pp. 649–666. Springer, 2016.

Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1058–1067, 2017.