Low-shot visual object recognition

Bharath Hariharan  
Facebook AI Research  
bharathh@fb.com

Ross Girshick  
Facebook AI Research  
rbg@fb.com

Abstract

Low-shot visual learning—the ability to recognize novel object categories from very few examples—is a hallmark of human visual intelligence. Existing machine learning approaches fail to generalize in the same way. To make progress on this foundational problem, we present a novel protocol to evaluate low-shot learning on complex images where the learner is permitted to first build a feature representation. Then, we propose and evaluate representation regularization techniques that improve the effectiveness of convolutional networks at the task of low-shot learning, leading to a 2x reduction in the amount of training data required at equal accuracy rates on the challenging ImageNet dataset.

1 Introduction

In recent times, error rates on benchmarks such as ImageNet [6] have been halved, and then halved again. These gains come from deep convolutional networks (convnets) that learn rich feature representations [20]. It is now commonly accepted that if an application has an a priori fixed set of visual concepts and thousands of examples per concept, the best way to build an object recognition system is to train a deep convnet. But what if these assumptions are not satisfied and the network must learn novel categories from very few examples?

The ability to perform low-shot learning—learning novel concepts from very few examples—is a hallmark of the human visual system. We are able to do this not only for natural object categories such as different kinds of animals, but also for synthetic objects that are unlike anything we’ve seen before [34]. This ability likely stems from how our brains represent visual percepts. These representations are “trained” on the vast amount of visual stimuli experienced during development from infancy to adulthood. In this sense they correspond by loose analogy to the visual representations learnt by a convnet. Yet existing machine learning approaches fail to generalize well from few examples [21]. Our goal in this paper is to modify how convnets are trained in order to improve generalization in the low-shot learning of novel categories.

As illustrated in Figure 1, most existing work falls into two camps: (1) one-shot learning, for which the dominant focus has been restricted domains such as handwritten characters or faces [21,38], and (2) transfer learning, for which the image domains are complex, but novel categories have abundant training data [7,29]. Our first contribution is to bridge the gap between these lines of work. Unlike most work on transfer learning, we seek to transfer representational knowledge to novel categories that have limited training data, from one to tens of examples. Unlike most work on one-shot learning, we focus on complex datasets with diverse categories and large intra-class variation.

Specifically, we propose an evaluation protocol based on ImageNet-1k, a challenging dataset for general category object recognition. In the representation learning phase of the protocol, the learner is permitted to tune its feature representation on a set of base classes. This feature representation is then frozen, and in the low-shot learning phase, the learner trains a classifier for an additional set of novel classes with only a few examples per class. We measure the learner’s accuracy for varying amounts of novel-category training data.
Figure 1: Prior work on one-shot learning typically assumes simplistic and restricted domains such as handwritten characters. In contrast, the transfer learning literature uses complex domains, but typically assumes the availability of extensive training data for novel categories. Our paper bridges this gap and focuses on low-shot learning in realistic and complex domains.

On this testbed, we propose and evaluate novel methods of regularizing learnt feature representations. We find that a variety of convnet feature regularization methods greatly improve low-shot learning and in practice decrease the number of samples needed to match baseline performance by 2x. We view this work as a step towards low-shot visual object recognition, hoping that it spurs further research on this practical and challenging problem.

2 Related work

One-shot learning: One class of approaches to one-shot learning uses generative models of appearance. The benefit of such approaches is that they can tap into a global [10] or a supercategory-level [33] prior. For restricted domains such as hand-written characters [24, 21], there exist powerful generative models that compose characters from a dictionary of parts [41] or strokes [22]. Such generative models have shown promise in restricted domains or simpler datasets without much intra-class variation, such as Caltech 101 [10]. However, the unrestricted visual world is far too challenging even for state-of-the-art generative models. Jia et al. [18] present a promising alternative using Bayesian reasoning to infer an object category from a few examples; however, in their setting examples from all categories are “seen” during training.

Among discriminative approaches, early work attempted to use a single image of the novel class to adapt classifiers from similar base classes [3, 28], but only used simple hand-crafted features. More recent work uses metric learning approaches such as the triplet loss [38, 35, 11] or siamese networks [19, 14] to automatically learn feature representations where objects of the same class are closer together. Such approaches have shown benefits in face identification [38], where the novel “classes” are quite similar to the base classes. On more challenging benchmarks such as ImageNet [6], these methods perform worse than simple classification baselines [31], and it is unclear if they can benefit low-shot learning.

Zero-shot learning: Zero-shot recognition uses textual or attribute-level descriptions of object classes to learn classifiers. While this is a different problem than ours, the motivation is the same: to reduce the amount of data required to learn classifiers. One line of work uses hand-designed attribute descriptions that are provided to the system for the novel categories [32, 23, 9]. Another class of approaches embeds images into word embedding spaces learnt using large text corpora, so that classifiers for novel concepts can be obtained simply from the word embedding of the concept [12, 37, 27, 42]. A final class of approaches attempts to directly regress to image classifiers from textual descriptions [8, 25] or from prototypical images of the category [17]. While the results from these approaches are encouraging, the assumption that no training images are available for the novel category might be extreme. Furthermore, it is often easier to obtain a few images of an object class, than a detailed textual description containing enough information to uniquely identify the category.

Transfer learning: The ability to learn novel classes quickly is one of the main motivations for multitask or transfer learning. Thrun’s classic paper convincingly argues that “learning the n-th task should be easier than learning the first,” with ease referring to sample complexity [59]. However,
recent transfer learning research has mostly focussed on the scenario where a lot of training data for novel classes is available. In such situations, the efficacy of pre-trained convnets for extracting features is well known [7, 29, 36]. There is also some analysis on what aspects of ImageNet training aid this transfer [1, 2]. Taigman et al. [38] find that low-dimensional feature representations transfer better on faces, and Galanti et al. [13] provide some theoretical justification for this finding. This hints at a link between the complexity of the feature representation and its generalizability, a link which we also observe in this paper. There have also been novel losses proposed explicitly to aid transfer, such as the multiverse loss of Littwin and Wolf [26]. This paper also proposes novel losses designed specifically for low-shot learning.

3 A protocol for evaluating low-shot learning

Our protocol for evaluating low-shot learning employs a learner, two training phases, and one testing phase. The learner is assumed to be composed of a feature extractor and a multi-class classifier. The protocol is agnostic to the specific form of each component.

During representation learning (training phase one), the learner receives a fixed set of base categories \( C_{\text{base}} \), and a dataset \( D_{\text{base}} \) containing a large number of examples for each category in \( C_{\text{base}} \). The learner uses \( D_{\text{base}} \) to set the parameters of its feature extractor. After this phase, the feature extractor is fixed and may not be altered. In the second phase, which we call low-shot learning, the learner is provided with an additional set of novel categories \( C_{\text{novel}} \). For each novel category, the learner has access to only \( n \) positive examples, where \( n \in \{1, 2, 5, 10, 20\} \). For the base categories, the learner still has access to \( D_{\text{base}} \). The learner may then use these examples and its fixed feature extractor to set the parameters of its multi-class classifier. This setup mimics a realistic scenario where after training the feature representation, we want to train the classifier to recognize novel categories for which we have only a few examples.

In the testing phase, the learnt model predicts labels from the combined label space \( C_{\text{base}} \cup C_{\text{novel}} \) on a set of previously unseen test images. To measure the variability in low-shot learning accuracy, we repeat the low-shot learning and testing phases for 5 trials, each time with a random draw of examples for the novel classes (but keeping the feature representation fixed). We report the mean accuracy, detailed later, and the standard deviation over these trials.

In this paper, we use a convnet as the learner’s feature extractor, and multi-class logistic regression as the classifier. After representation learning, the base-category classifier is removed from the convnet and the rest of the convnet is frozen and used as a feature extractor. See Section 5 for more implementation details.

4 Learning representations for low-shot learning

Our goal is to modify the representation learning phase to yield better generalization from low-shot learning. We first describe a proposal that encodes the goal of low-shot learning in a loss that can be minimized during representation learning. Then, we draw connections to several alternatives.

4.1 Squared gradient magnitude loss (SGM)

During representation learning, we use the base examples \( D_{\text{base}} = \{(x_i, y_i) \mid i = 1, \ldots, N\} \) to train a convnet. The convnet is composed of two parts: the feature representation \( \phi(\cdot) \), and the linear classifier over the base classes with weight matrix \( W = [w_1, \ldots, w_{|C_{\text{base}}|}] \). The model is trained with stochastic gradient descent (SGD) on the objective:

\[
W^*, \phi^* = \arg\min_{W, \phi} \sum_{i=1}^{N} L(W, \phi(x_i), y_i),
\]

in which \( L(W, \phi(x_i), y_i) = -\log p_{y_i}(x_i) \) is the log loss of the true class’s predicted probability, with the probability of class \( k \) for example \( i \) under the model computed by the standard softmax:

\[
p_{ik} = \exp(w_k \cdot \phi(x_i)) / \sum_j \exp(w_j \cdot \phi(x_i)).
\]

After representation learning, we will freeze the featurization function \( \phi^* \) and then train linear classifier weights \( W \) to classify novel classes given a few examples. For the moment, let us ignore
the fact that these are novel classes, and instead assume that we are simply going to retrain the weight matrix on the base classes, and this time with only a few examples per base class. In other words, suppose we want to train the new weight matrix $\hat{W}$ using a very small dataset $S \subset D_{base}$. We would do this by again minimizing $\sum_{(x,y) \in S} L(W, \phi^*(x), y)$ with respect to $W$, while keeping $\phi^*$ fixed.

In the limit of low-shot learning, $S$ will contain a single example $(x, y)$, and we would get $\hat{W}$ by minimizing the loss on that single instance: $\hat{W} = \arg \min_W L(W, \phi^*(x), y)$.

Intuitively, our goal is that $\hat{W}$ should be the same as $W^*$. This condition implies that the optimal solution given all examples is the same as the optimal solution given only one example, i.e. that low-shot learning (or one-shot learning, in this case) works well. Alternatively stated, we want $W^*$ to be the \textit{global minimizer} of $L(W, \phi^*(x), y)$, for any example $(x, y)$. Note that this is distinct from, and stricter than, wanting low loss on that example. The loss can be quite low even if $W^*$ is far from the optimum of $L(W, \phi^*(x), y)$, and so while $W^*$ might generalize well, $\hat{W}$ might not if it is far from $W^*$. Since the log loss $L(W, \phi^*(x), y)$ is convex in $W$, the global optimum is characterized by having zero gradient. Alternatively, if the gradient of $L(W, \phi^*(x), y)$ at $W^*$ is high, then that implies that $\hat{W}$ is far away from the optimum. Therefore, we want to make sure that $\|\nabla_W L(W^*, \phi^*(x), y)\|$ is low for each example $(x, y)$ during representation learning.

This argument suggests that we should add a gradient magnitude term to the objective in Equation (1):

$$W^*, \phi^* = \arg \min_{W, \phi} \sum_{i=1}^{N} L(W, \phi(x_i), y_i) + \lambda \|\nabla_W L(W, \phi(x_i), y_i)\|^2. \tag{2}$$

It is easy to show that $\nabla_W L(W, \phi(x_i), y_i)$ has a simple form:

$$\nabla_W L(W, \phi(x_i), y_i) = [\nabla_{\phi}, L(W, \phi(x_i), y_i), ... , \nabla_{\phi_{i_{base}}} L(W, \phi(x_i), y_i)] \tag{3}$$

$$\nabla_{\phi_k} L(W, \phi(x_i), y_i) = (p_{ik} - \mathbb{I}(y_i == k))\phi(x_i) \tag{4}$$

where $\mathbb{I}(\cdot)$ returns 1 if its argument is true and 0 otherwise, and $p_{ik}$ is the probability assigned by the classifier to the $k$-th class, as defined above. Thus, our convnet training objective takes the form:

$$W^*, \phi^* = \arg \min_{W, \phi} \sum_{i=1}^{N} L(W, \phi(x_i), y_i) + \lambda \alpha_i \|\phi(x_i)\|^2, \tag{5}$$

where $\alpha_i = \sum_k (p_{ik} - \mathbb{I}(y_i == k))^2$. The second term in Equation (5) penalizes the squared $L_2$ norm of the feature representation of $x_i$ with an example-dependent weight $\alpha_i$. This weight is higher for examples that are misclassified, and nears zero when the classification is perfect. This makes intuitive sense: high norm examples that are misclassified might be outliers, and in a low-shot learning scenario, such an outlier can pull the learnt weight vectors far away from the right solution. We refer to the squared gradient magnitude loss as SGM.

\textbf{Batch SGM.} SGM is derived under the assumption of $n = 1$ examples per class during low-shot learning. However, we may reasonably expect to have a small number of examples that is greater than 1 and indeed we evaluate over the range of $n \in \{1, 2, 5, 10, 20\}$. We therefore consider an extension of SGM that we call “batch SGM.” Here, rather than penalizing the squared gradient magnitude of a single example, we penalize the squared gradient magnitude of the loss averaged over all examples in each SGD mini-batch $B$, yielding the loss term: $\lambda \|\nabla_W \left(1/|B| \sum_{(x,y) \in B} L(W, \phi^*(x), y)\right)\|^2$.

Note that because this loss is defined on mini-batches the number of examples per class in each mini-batch is a random variable. Thus batch SGM optimizes for an expected loss over a distribution of possible low-shot values $n$.

\subsection{4.2 Feature regularization-based alternatives}

One can ask if the weights $\alpha_i$ in Equation (5) are necessary. We can therefore consider a simplification of Equation (5) in which we use a simple squared $L_2$ norm:

$$W^*, \phi^* = \arg \min_{W, \phi} \sum_{i=1}^{N} L(W, \phi(x_i), y_i) + \lambda \|\phi(x_i)\|^2. \tag{6}$$
While $L_2$ regularization is a common technique, note that here we are regularizing the feature representation, as opposed to regularizing the weight vector.

We can also consider other ways of regularizing the representation, such as an $L_1$ regularization:

$$W^*, \phi^* = \arg\min_{W, \phi} \sum_{i=1}^{N} L(W, \phi(x_i), y_i) + \lambda \|\phi(x_i)\|_1.$$  (7)

While regularizing the feature vector norm has been a staple of unsupervised learning approaches to prevent degenerate solutions [30], to the best of our knowledge it hasn’t been considered in supervised classification.

We also evaluate other forms of feature regularization that have been proposed in the literature. The first of these is dropout [16], which were used in earlier convnet architectures [20], but has been eschewed by recent architectures such as ResNets [15]. Another form of feature regularization involves minimizing the correlation between the features [4, 5]. As a final baseline, we also consider the multiverse loss [26] which was shown to improve transfer learning performance.

### 4.3 Metric-learning based approaches

Distance metric learning is a common approach in the one-shot learning literature. The intuition being that the distance metric can generalize to novel classes. There are many metric learning approaches. We train a convnet with the triplet loss as a representative method. The triplet loss takes as input a triplet of examples $(x, x_+, x_-)$, where $x$ and $x_+$ belong to the same category while $x_-$ doesn’t:

$$L_{\text{triplet}}(\phi(x), \phi(x_+), \phi(x_-)) = \max(0, \|\phi(x_+) - \phi(x)\| - \|\phi(x_-) - \phi(x)\| + \gamma).$$  (8)

Here, $\gamma$ is a margin and intuitively the loss is zero if $x_-$ is at least $\gamma$ farther away from $x$ than $x_+$ is. Otherwise the loss increases the closer $x_-$ is to $x$ or the farther $x_+$ is from $x$.

### 5 Experiments and discussion

#### 5.1 Network architecture and training details

We use a small ten-layer ResNet architecture [15] (details in supplementary material). When trained on all 1000 categories of ImageNet, it gives a validation top-5 error rate of 16.7% (center crop), making it similar to AlexNet [20]. We use this architecture because it’s relatively fast to train (2 days on 4 GPUs) and resembles state-of-the-art architectures. Note that ResNet architectures, as described in [15], do not use dropout.

For all methods, except the triplet embedding, the networks are trained from scratch for 90 epochs on the base classes. The learning rate starts at 0.1 and is divided by 10 every 30 epochs. The weight decay is fixed at 0.0001. For the triplet embedding, we first pretrain the network using a softmax classifier and log loss for 90 epochs, and then train the network further using the triplet loss and starting with a learning rate of 0.001. We stop training when the loss stops decreasing (55 epochs). This schedule is used because, as described in [31], triplet networks train slowly from scratch.

For methods that introduce a new loss, there is a hyperparameter that controls how much we weigh the new loss. Dropout also has a similar hyperparameter that governs what fraction is dropped out at each iteration. We set these hyperparameters by cross-validation as described below.

#### 5.2 Training with class imbalance

The evaluation protocol creates a heavily class-imbalanced classification problem during low-shot learning: the base classes may have thousands of examples, while each novel class may have only a few examples. We use two simple strategies to mitigate this issue. One, we oversample the novel classes when training the classifier by sampling uniformly over classes and then uniformly within each chosen class. Two, we $L_2$ regularize the multi-class logistic classifier’s weights.

---

1The supplementary material will soon be available at http://tinyurl.com/lowshot-zip
2A random initialization, but with the same random number generator seed for all experiments.
We observe that:

Among domestic animals, we have 36 base and 85 novel categories. We find that the weight of the classifier’s $L_2$ regularization term has a large impact and needs to be cross-validated. To do so, we set aside 5 examples per novel class, and half of $D_{\text{base}}$ as a validation set. Using the second half of $D_{\text{base}}$, and with $n \in \{1, 2, 5, 10, 20\}$ examples per class from the novel classes, we train multi-class logistic classifiers with different values of the $L_2$ regularization weight. We then select the best hyperparameter value according to the average top-5 accuracy on the validation set (averaged over the different values of $n$ and all base and novel classes). This strategy implements a naive cross-validation protocol that doesn’t favor any particular value of $n$ or the novel classes over base classes. We use top-5 accuracy because it is more stable than top-1 for small $n$.

5.3 Low-shot learning on ImageNet-1k

Setup. We conduct the low-shot evaluation using classes and images from the ImageNet-1k challenge dataset. We use ImageNet because it has a wide array of classes with significant intra-class variation. We divide the 1000 categories into 389 base and 611 novel categories, which are listed in the supplementary material. We use all 389 base classes for representation learning in all experiments. The categories are diverse, and some of the novel classes are visually far away from any of the base classes. While it is important to be able to generalize in such cases, it is also useful to look at accuracy when the novel classes are nearer to base classes. To evaluate this scenario, we use the ImageNet hierarchy to restrict attention to the “domestic animals” subtree in some experiments. In this case, during the low-shot learning phase we only take base and novel classes from the domestic animals subtree, and train a $K_{\text{animal}}$-way classifier, where $K_{\text{animal}} = |C_{\text{base animal}}| + |C_{\text{novel animal}}|$. Among domestic animals, we have 36 base and 85 novel categories.

Performance is measured by top-1 and top-5 accuracy on the test set for each value of $n$ (we use the ImageNet-1k validation set for testing). We report two sets of numbers: average accuracy on the test samples from the novel classes and on the samples from the base classes. While our focus is on the novel classes, we nevertheless need to ensure that good performance on novel classes doesn’t come at the cost of lower accuracy on the base classes.

Results. We plot a subset of the methods \cite{9} in Figure 2 and show the full set of numbers in Tables 1 and 2. The plots show the mean top-5 accuracy, averaged over 5 low-shot learning trials, for the novel and base classes of the full ImageNet-1k tree and the domestic animals subtree. The standard deviations are low (generally less than 0.5%, see Tables 1 and 2) and are too small to display clearly as error bars. Top-1 accuracy and numerical values are in the supplementary material.

We observe that:

- Batch SGM, SGM, $L_2$ and the baseline perform similarly on the base classes. However, there are large differences in accuracy on the novel classes. Further, these differences are larger for small $n$, and slowly even out as $n$ increases. This indicates that performance on base classes cannot be extrapolated to novel classes, especially if the novel classes have very few training examples.

- The baseline model trained with softmax performs the worst. For $n = 1$ or 2, it is almost 10 points worse than the best variant. For domestic animals, this gap persists even for $n = 20$.

- Batch SGM, SGM, and $L_2$ require 2x fewer samples than the baseline to match its accuracy. For $n \leq 10$, all three approaches provide similar and consistent gains over the baseline. For domestic animals this gain is larger and is sustained even for $n = 20$.

- Dropout, $L_1$, the DeCov loss \cite{5} and the multiverse loss \cite{26} also provide gains over the baseline for small values of $n$, but are outperformed by the SGM variants and the $L_2$ loss. This suggests that other forms of feature regularization can work well too. Empirically, except for the multiverse loss, all these methods tend to reduce feature norm, suggesting that implicit $L_2$ feature regularization might explain some of these gains.

- The triplet network is somewhat better than the baseline, but it is outperformed significantly by the feature regularization approaches. Note that more sophisticated variants of the triplet loss may improve performance \cite{31}. However, it seems clear that feature regularization alternatives are not only competitive, but also simpler.

\footnote{The subset reduces clutter, making the plots more readable. We omit results for Batch SGM and $L_1$ because Batch SGM performs similarly to SGM and $L_2$, while $L_1$ performs worse (similar to multiverse and triplet).}
We also evaluate the image rankings produced by the baseline and SGM. To do so, we sampled 20 novel classes at random and then selected three classes with notable differences in the top four ranked images. The rankings are displayed in Figure 4 together with the seed images that were used for low-shot learning (\(n = 2\), in this case). In the first set, SGM has higher semantic consistency than the baseline, which confuses cows for camels in two of the four images. In the second set, the baseline appears to rely too heavily on color, causing it to rank an image of a lobster highly. In the last set, again SGM maintains higher semantic consistency than the baseline.

### Qualitative results
Figure 5 shows T-SNE embeddings [40] of a fixed set of images drawn from five base and five novel classes. We show embeddings for three methods: baseline, SGM, and \(L_2\). Compared to the baseline, SGM and \(L_2\) appear to place the novel classes into tighter clusters with better separation.

We also evaluate the image rankings produced by the baseline and SGM. To do so, we sampled 20 novel classes at random and then selected three classes with notable differences in the top four ranked images. The rankings are displayed in Figure 4 together with the seed images that were used for low-shot learning (\(n = 2\), in this case). In the first set, SGM has higher semantic consistency than the baseline, which confuses cows for camels in two of the four images. In the second set, the baseline appears to rely too heavily on color, causing it to rank an image of a lobster highly. In the last set, again SGM maintains higher semantic consistency than the baseline.

### Table 1: Top-5 accuracy on ImageNet-1k val on the root hierarchy.

| \(n\) | Base classes | Batch SGM | L2 | Triplet | SGM | Dropout | L1 | Multiverse | DeCov |
|------|--------------|----------|----|--------|-----|---------|----|------------|-------|
| 1    | 4.32 ± 0.32  | 18.78 ± 0.52 | 17.86 ± 0.69 | 8.52 ± 0.37 | 18.80 ± 0.59 | 12.17 ± 0.33 | 8.43 ± 0.27 | 8.67 ± 0.30 | 9.07 ± 0.32 |
| 2    | 15.31 ± 0.44 | 33.85 ± 0.68 | 33.32 ± 0.78 | 22.79 ± 0.64 | 34.07 ± 0.65 | 26.48 ± 0.59 | 22.28 ± 0.35 | 22.44 ± 0.63 | 26.45 ± 0.27 |
| 5    | 37.61 ± 0.39 | 48.61 ± 0.33 | 49.57 ± 0.23 | 48.94 ± 0.30 | 46.12 ± 0.41 | 44.11 ± 0.45 | 43.08 ± 0.41 | 48.24 ± 0.54 |
| 10   | 51.45 ± 0.25 | 54.98 ± 0.20 | 56.55 ± 0.15 | 52.85 ± 0.31 | 55.46 ± 0.35 | 56.19 ± 0.32 | 56.13 ± 0.23 | 54.54 ± 0.51 | 56.78 ± 0.41 |
| 20   | 61.07 ± 0.19 | 58.29 ± 0.21 | 60.08 ± 0.34 | 58.35 ± 0.26 | 58.94 ± 0.24 | 62.04 ± 0.34 | 63.36 ± 0.20 | 62.12 ± 0.44 | 61.42 ± 0.26 |

| \(n\) | Novel classes (zoom \(n = 1, 2\)) |
|------|-------------------------------|
| 1    | 87.81 ± 0.07 | 86.32 ± 0.15 | 86.96 ± 0.04 | 84.39 ± 0.04 | 86.61 ± 0.10 | 86.53 ± 0.04 | 87.75 ± 0.08 | 88.33 ± 0.09 | 87.53 ± 0.10 |
| 2    | 87.21 ± 0.10 | 84.16 ± 0.21 | 84.58 ± 0.13 | 82.42 ± 0.11 | 84.37 ± 0.12 | 84.68 ± 0.16 | 86.57 ± 0.05 | 87.30 ± 0.08 | 85.62 ± 0.09 |
| 5    | 84.89 ± 0.12 | 82.31 ± 0.11 | 82.21 ± 0.21 | 77.97 ± 0.14 | 82.47 ± 0.09 | 81.00 ± 0.14 | 83.35 ± 0.09 | 84.54 ± 0.18 | 81.83 ± 0.11 |
| 10   | 82.33 ± 0.03 | 82.02 ± 0.10 | 81.80 ± 0.17 | 76.20 ± 0.12 | 82.13 ± 0.11 | 79.20 ± 0.12 | 81.05 ± 0.10 | 82.01 ± 0.14 | 80.78 ± 0.12 |
| 20   | 80.56 ± 0.14 | 81.96 ± 0.15 | 81.74 ± 0.19 | 75.59 ± 0.11 | 82.10 ± 0.11 | 78.48 ± 0.18 | 79.94 ± 0.17 | 80.43 ± 0.15 | 80.61 ± 0.16 |

Figure 2: Top-5 accuracy on ImageNet-1k val on base and novel categories for different convnet training methods. The top-performing feature regularization methods reduce the number of training samples needed to match the baseline accuracy by 2x.
Figure 3: T-SNE visualizations of the feature representations learnt by the baseline (top), SGM (middle) and $L_2$ regularization (bottom) for a set of five base and five novel classes.

Figure 4: Examples of the ranking produced by the baseline and by the SGM loss. Each set represents a novel category learnt with a few examples ("seeds"). The top row of each set shows the top ranked images for this category according to the baseline and the bottom row is the top ranked images when using the representation trained with SGM. See text for a discussion.
We thank Piotr Dollar and Larry Zitnick for the many insightful discussions.

Acknowledgements

We thank Piotr Dollar and Larry Zitnick for the many insightful discussions.

References

[1] P. Agrawal, R. Girshick, and J. Malik. Analyzing the performance of multilayer neural networks for object recognition. In ECCV, 2014.

[2] H. Azizpour, A. Sharif Razavian, J. Sullivan, A. Maki, and S. Carlsson. From generic to specific deep representations for visual recognition. In CVPR, 2015.

[3] E. Bart and S. Ullman. Cross-generalization: Learning novel classes from a single example by feature replacement. In CVPR, 2005.

[4] B. Cheung, J. A. Livezey, A. K. Bansal, and B. A. Olshausen. Discovering hidden factors of variation in deep networks. In ICLR, 2015.

[5] M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick, and D. Batra. Reducing overfitting in deep networks by decorrelating representations. In ICLR, 2016.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In CVPR, 2009.

[7] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In ICML, 2014.

[8] M. Elhoseiny, B. Saleh, and A. Elgammal. Write a classifier: Zero-shot learning using purely textual descriptions. In ICCV, 2013.

[9] A. Farhadi, I. Endres, and D. Hoiem. Attribute-centric recognition for cross-category generalization. In CVPR, 2010.

[10] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. TPAMI, 2006.

[11] M. Fink. Object classification from a single example utilizing class relevance metrics. NIPS, 2005.

[12] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In NIPS, 2013.

[13] T. Galanti, L. Wolf, and T. Hazan. A theoretical framework for deep transfer learning. Information and Inference, 2016.

6 Conclusion

Existing research on one-shot learning typically considers narrow domains, such as handwritten characters or faces. In contrast, mainstream object recognition focuses on comparatively less restricted domains. The goal of this paper is two-fold: (1) to push research on low-shot learning towards unrestricted image sets, such as ImageNet, and (2) to present feature regularization techniques that significantly improve the low-shot generalization abilities of convnet features. Our novel feature regularization method, called squared gradient magnitude (SGM), is derived by encoding the end-goal of low-shot learning as a soft constraint during convnet training. We show that SGM is closely related to \( L_2 \) feature regularization. We experimentally compare these methods and find that they reduce the number of samples required to reach baseline accuracy by 2x.

Table 2: Top-5 accuracy on ImageNet-1k val on the domestic animal subtree.

| \( n \) | Baseline | Batch SGM | L2 | Triplet SGM | Dropout | L1 | Multiverse | DeCov |
|---|---|---|---|---|---|---|---|---|
| Novel classes |
| 1 | 14.85 ± 0.90 | 28.19 ± 1.93 | 27.48 ± 1.76 | 12.90 ± 0.96 | 28.70 ± 2.24 | 18.25 ± 1.12 | 25.26 ± 2.12 | 15.20 ± 0.49 | 15.46 ± 1.35 |
| 2 | 32.79 ± 1.64 | 45.89 ± 2.49 | 45.38 ± 2.59 | 29.62 ± 1.53 | 45.96 ± 2.18 | 36.13 ± 1.65 | 41.72 ± 0.95 | 32.14 ± 1.02 | 35.54 ± 1.18 |
| 5 | 53.77 ± 1.24 | 63.17 ± 1.39 | 62.53 ± 1.34 | 52.29 ± 1.21 | 63.59 ± 1.81 | 57.17 ± 1.42 | 58.82 ± 1.14 | 54.78 ± 1.10 | 59.12 ± 1.33 |
| 10 | 62.78 ± 0.48 | 70.23 ± 0.81 | 69.56 ± 0.46 | 63.59 ± 0.84 | 70.44 ± 0.89 | 67.35 ± 0.96 | 65.56 ± 0.65 | 65.96 ± 0.42 | 69.12 ± 0.55 |
| 20 | 68.00 ± 0.64 | 74.31 ± 0.34 | 73.90 ± 0.46 | 69.92 ± 0.63 | 74.17 ± 0.09 | 74.19 ± 0.56 | 69.69 ± 1.11 | 73.46 ± 0.68 | 74.17 ± 0.62 |
| Base classes |
| 1 | 91.67 ± 0.28 | 94.07 ± 0.11 | 93.97 ± 0.29 | 95.31 ± 0.27 | 94.18 ± 0.43 | 95.37 ± 0.07 | 95.58 ± 0.63 | 96.18 ± 0.15 | 95.92 ± 0.40 |
| 2 | 86.89 ± 0.35 | 91.01 ± 0.33 | 90.96 ± 0.43 | 93.37 ± 0.17 | 91.17 ± 0.61 | 93.57 ± 0.17 | 95.22 ± 0.39 | 95.13 ± 0.18 | 94.34 ± 0.39 |
| 5 | 81.49 ± 0.71 | 88.31 ± 0.58 | 88.10 ± 0.44 | 88.36 ± 0.28 | 88.48 ± 0.53 | 89.91 ± 0.35 | 82.41 ± 0.53 | 92.60 ± 0.41 | 90.40 ± 0.44 |
| 10 | 80.32 ± 0.76 | 87.39 ± 0.19 | 87.21 ± 0.51 | 85.76 ± 0.24 | 87.64 ± 0.30 | 87.40 ± 0.54 | 82.32 ± 0.62 | 90.03 ± 0.26 | 88.44 ± 0.45 |
| 20 | 80.11 ± 0.88 | 87.40 ± 0.35 | 87.07 ± 0.44 | 84.86 ± 0.40 | 87.40 ± 0.16 | 85.56 ± 0.45 | 82.44 ± 0.50 | 87.97 ± 0.42 | 87.56 ± 0.27 |
[14] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.
[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR (To appear)*, 2016.
[16] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.
[17] S. Jetley, B. Romera-Paredes, S. Jayasumana, and P. Torr. Prototypical priors: From improving classification to zero-shot learning. In *BMVC*, 2015.
[18] Y. Jia and T. Darrell. Latent task adaptation with large-scale hierarchies. In *ICCV*, 2013.
[19] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, 2015.
[20] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012.
[21] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 2015.
[22] B. M. Lake, R. R. Salakhutdinov, and J. Tenenbaum. One-shot learning by inverting a compositional causal process. In *NIPS*, 2013.
[23] C. H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *TPAMI*, 2014.
[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 1998.
[25] J. Lei Ba, K. Swersky, S. Fidler, and R. salakhutdinov. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *ICCV*, 2015.
[26] E. Littwin and L. Wolf. The multiverse loss for robust transfer learning. *arXiv preprint arXiv:1511.09033*, 2015.
[27] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-shot learning by convex combination of semantic embeddings. In *ICLR*, 2014.
[28] A. Opelt, A. Pinz, and A. Zisserman. Incremental learning of object detectors using a visual shape alphabet. In *CVPR*, 2006.
[29] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014.
[30] M. Ranzato. *Unsupervised learning of feature hierarchies*. PhD thesis, New York University, 2009.
[31] O. Rippel, M. Paluri, P. Dollar, and L. Bourdev. Metric learning with adaptive density discrimination. In *ICLR*, 2016.
[32] B. Romera-Paredes and P. Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2015.
[33] R. Salakhutdinov, J. Tenenbaum, and A. Torralba. One-shot learning with a hierarchical nonparametric bayesian model. *Unsupervised and Transfer Learning Challenges in Machine Learning*, 2012.
[34] L. A. Schmidt. *Meaning and compositionalty as statistical induction of categories and constraints*. PhD thesis, Citeseer, 2009.
[35] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.
[36] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In *CVPR Workshops*, 2014.
[37] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, 2013.
[38] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Web-scale training for face identification. In *CVPR*, 2015.
[39] S. Thrun. Is learning the n-th thing any easier than learning the first? *NIPS*, 1996.
[40] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008.
[41] A. Wong and A. L. Yuille. One shot learning via compositions of meaningful patches. In *ICCV*, 2015.
[42] Z. Zhang and V. Saligrama. Zero-shot learning via semantic similarity embedding. In *ICCV*, 2015.