Learning Compositional Representations for Few-Shot Recognition

Pavel Tokmakov
Carnegie Mellon University
ptokmako@cs.cmu.edu

Yu-Xiong Wang
Carnegie Mellon University
yuxiongw@cs.cmu.edu

Martial Hebert
Carnegie Mellon University
hebert@cs.cmu.edu

Abstract

One of the key limitations of modern deep learning based approaches lies in the amount of data required to train them. Humans, on the other hand, can learn to recognize novel categories from just a few examples. Instrumental to this rapid learning ability is the compositional structure of concept representations in the human brain — something that deep learning models are lacking. In this work we make a step towards bridging this gap between human and machine learning by introducing a simple regularization technique that allows the learned representation to be decomposable into parts. We evaluate the proposed approach on three datasets: CUB-200-2011, SUN397, and ImageNet, and demonstrate that our compositional representations require fewer examples to learn classifiers for novel categories, outperforming state-of-the-art few-shot learning approaches by a significant margin.

1. Introduction

Consider the images representing four categories from the CUB-200-2011 dataset [40] in Figure 1. Given a representation learned using the first three categories, shown in red, can a classifier for the fourth category shown in green be learned from just a few, or even a single example - a problem known as few-shot learning [38, 21, 18, 12]? Clearly, this depends on the properties of the representation. According to cognitive science one property that is crucial for solving this problem is compositionality. Human representations of concepts are decomposable into parts [6, 17], such as the ones shown in the top right corners of the images in Figure 1, allowing for classifiers to be rapidly learned for novel concepts through combination of known primitives [13] (see the example of the novel bird category - all of its discriminative attributes have already been observed in the first three categories). These ideas have been highly influential in computer vision, with some of the first models for visual concepts being built as compositions of parts and relations [26, 27, 42].

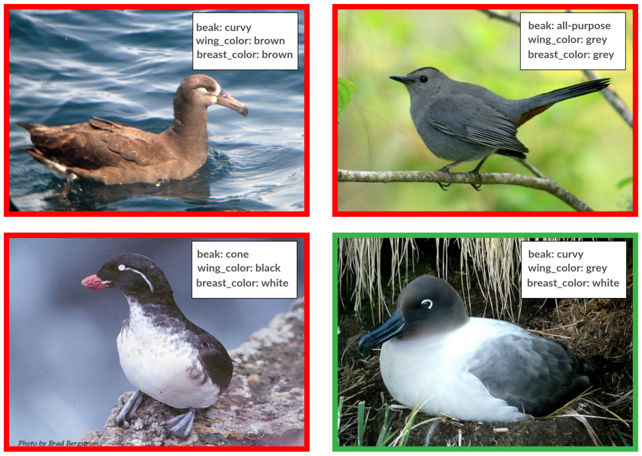

Figure 1. Images from four categories of the CUB-200-2011 datasets, together with some of their attribute annotations. We propose to learn image representations that are decomposable over the attributes and thus can learn new categories from few examples.

However, state-of-the-art methods for virtually all visual recognition tasks are based on deep learning [24, 20]. The parameters of deep neural networks are optimized for the end task with gradient-based methods, resulting in representations that are not easily interpretable. There has been a lot of effort for qualitative interpretation of these representations [45, 46]. Very recently a quantitative approach for evaluating the compositionality of deep representations has been proposed [3]. The authors posit that a feature encoding of an image is compositional if it can be represented as a sum of the encodings of attributes describing the image and design an algorithm to quantify this property. They also observe that representations that are more compositional exhibit a better generalization behavior. In this work we turn the compositionality measure proposed in [3] into a constraint for training neural networks, forcing learned image representations to be decomposable into parts.

Our method for learning compositional representations for visual recognition takes as input a dataset of images together with their class labels and category-level attribute annotations. The attributes can be both purely visual, such as object parts - `beak_shape`, or scene elements - `grass`, and more abstract, such as `openness` of a scene. To enforce the learned representation to be decomposable over these attributes we turn the compositionality measure of [3] into a regularization term in a loss function. In particular given an image with its corresponding attribute annotations, like the ones shown in Figure 1, we jointly learn a CNN for the image embedding and a linear layer for the attribute embedding. We then constrain the image representation to be equal to the sum of the attribute representations. Intuitively, applied together with a classification loss, this regularization forces the optimization to chose a representation that is more compositional over the attributes out of the space of all possible discriminative representations.

This constraint, however, implies that exhaustive attribute annotations are available. Such an assumption is not realistic for most of the image domains. To address this issue, we propose a relaxed version of the compositionality regularizer. We evaluate our approach in a few-shot recognition setting on three datasets of different sizes and domains: CUB-200-2011 [40] for fine grained recognition, SUN397 for scene classification [43] and ImageNet [9] for object classification. We demonstrate a significant improvement in generalization performance on all three datasets. In particular, our model learned with the proposed regularization achieves a 6.9% top-5 accuracy improvement over the baseline in the most challenging 1-shot scenario on the CUB-200-2011 dataset. Overall, we demonstrate state-of-the-art results on all three datasets in a variety of settings.

One obvious limitation of the proposed approach is that it requires additional annotations. One might ask, how expensive it is to collect the attribute labels, and, more importantly, how to even define the vocabulary of attributes for an arbitrary dataset. To illustrate that collecting category-level attributes is in fact relatively easy even for large-scale datasets, we label 159 attributes for a subset of the ImageNet categories defined in [15]. A crucial detail is that the attributes have to be labeled on the category, not on the image level, which allowed one of the authors to collect the annotations in just three days. A representation learned with these attributes achieves state-of-the-art results in a few-shot evaluation setting. We are planning to release our collected attribute annotations together with the code and trained models. More details on the annotations process are provided in Section 3.4.

Our contributions are three-fold. (1) We propose the first approach for learning deep compositional representations in Section 3. Our method takes images together with their attribute annotations as input and applies a soft regularizer to enforce the image representation to be decomposable over the attributes. (2) We illustrate the simplicity of collecting attribute annotations on a subset of the ImageNet dataset in Section 3.4. (3) We provide a comprehensive analysis of the learned representation in the context of few-shot learning on three datasets. The evaluation in Section 6.1 demonstrates that the proposed approach results in a representation that generalizes significantly better and requires fewer examples to learn novel categories.

2. Related Work

Few-shot learning is a classical problem of recognition with only a few training examples [38]. Lake *et al.* [21] proposed to explicitly encode compositionality and causality properties with bayesian probabilistic programs. Learning then boils down to constructing programs that best explain the observations and can be done efficiently with a single example per category. This approach is limited however by the fact that the programs have to be manually defined for each new domain.

State-of-the-art methods for few-shot learning can be categorized into the ones based on metric learning [18, 39, 36, 44] — training a network to predict whether two images belong to the same category, and the ones built around the idea of meta-learning [12, 33] — training with a loss that explicitly enforces easy adaptation of the weights to new categories with only a few examples. Separately from these approaches, some authors propose to learn to generate additional examples for unseen categories [41, 15]. Recently it has been shown that it is crucial to use cosine similarity as a distance measure to achieve top results in few shot learning evaluation [14]. Even more recently the authors of [1] demonstrated that a simple baseline approach — a linear layer learned on top of a frozen CNN, achieves state-of-the-art results on two few shot learning benchmarks. The key to the success of their baseline is using cosine classification function and applying standard data augmentation techniques during few-shot training. Here we confirm their observation about the surprising efficiency of this baseline in a more realistic setting and demonstrate that learning a classifier on top of the compositional feature representation results in a significant improvement in performance.

Compositional representations have been extensively studied in the cognitive science literature [6, 17, 13] with Biderman's Recognition-By-Components theory being especially influential in computer vision. One especially attractive property of compositional representations is that they allow learning novel concepts from a few or even a single example by composing known primitives. Lake et al. [22] argue that compositionality is one of the key building blocks of human intelligence that is missing in the state-of-the-art AI systems. Although early computer vision models have been inherently compositional [26, 27, 42], building upon feature hierarchies [11, 47] and part-based models [30, 10], modern deep learning systems [24, 20, 16] do not explicitly model concepts as combinations of parts.

Analysis of internal representations learned by deep networks [45, 35, 25, 46, 19] has shown that some of the neurons in the hidden layers do encode object and scene parts.

However, all these works observe that the discovered compositional structure is limited and qualitative analysis of network activations is highly subjective. Very recently an approach for quantitative evaluation of compositionality of learned representations has been proposed in [3]. We build on top of the formalism proposed in that work, but instead of using it to measure the properties of a learned model turn it into a training objective.

Among works that explicitly address compositionality in deep learning models, Misra *et al.* [29] propose to train a network that predicts classifiers for novel concepts by composing existing classifiers for the parts. For instance, their model can obtain a classifier for the category `large elephant` by combining independently learned classifiers for categories `elephant` and `large` without seeing a single image with a label `large elephant`. In contrast, we propose to train a single model that internally decomposes concepts into parts and show results in a few-shot setting. In [37] the authors address the notion of spatial compositionality, proposing to constrain network representations of objects in an image to be independent from each other and from the background. They then demonstrate that networks trained with this constraint generalize better to the test distribution. Our work is similar in that we too propose to enforce decomposition of network representation into parts with the goal of increasing its generalization abilities. Our approach, however, does not require spatial, or even image-level supervision and thus can handle abstract attributes, as well as be readily applied to large scale datasets.

Learning with attributes has been studied for a variety of applications. Most notably, zero-shot learning methods use category-level attributes to recognize novel classes without seeing any training examples [4, 5, 8, 23]. To this end they propose to learn models that take attributes as input and predict image classifiers, allowing them to recognize never before seen classes as long as they can be described by the known attribute vocabulary. Very recently it has ben shown [2] that such attribute-based classifiers also require less data for training. In contrast, our method uses attributes to learn compositional image representations that require fewer training examples to recognize novel concepts. Crucially, unlike the method described above, our approach does not require attribute annotations for novel classes.

Another context in which attributes have been used is that of active [31, 7] and semi-supervised learning [34]. In [31] the authors use attribute classifiers to mine hard negative images for a category based on user feedback. In [7] a method that explicitly constructs classifiers by combing discriminative attributes provided by the user into DNF formulas is proposed. Our method is offline and does not require user interactions. In [34] the attributes are used to explicitly provide constraints when learning from a small number of labeled and a large number of unlabeled images. Our approach uses attributes to constraint a learned deep image representation, resulting in these constraints being implicitly encoded by the network.

3. Our Approach

3.1. Problem Formulation

We consider the task of few-shot image classification. Formally, we have a set of base categories $\mathcal{C}_{base}$ and a corresponding dataset $S_{base} = \{(x_i, y_i)\}, x_i \in \mathcal{X}, y_i \in \mathcal{C}_{base}$ which contains a large number of examples per category. We also have a set of unseen novel categories $\mathcal{C}_{novel}$ and a corresponding dataset $S_{novel} = \{(x_i, y_i)\}, x_i \in \mathcal{X}, y_i \in \mathcal{C}_{novel}$ which consists of only n examples per category, where n could be as few as one. We learn a representation model f_θ parametrized by θ on S_{base} that can be used for the downstream classification task on S_{novel}.

While there might exist many possible representations that can be learned and achieve similar generalization performance on the base categories, we argue that the one that is decomposable into shared parts will be able to generalize better to novel categories from fewer examples. Consider again the example in Figure 1. Intuitively, a model that has internally learned to recognize the attributes `beak:curved`, `wing_color:grey`, and `breast_color:white` is able to obtain a classifier of the never-before-seen bird species simply by composition. But how can this intuitive notion of compositionality be formulated in the space of deep representation models?

Inspired by [3], on the base dataset S_{base}, we augment the category labels $y_i \in \mathcal{C}_{base}$ of the examples x_i, with information about the structure of the examples in the form of *derivations* $D(x_i)$, defined over as set of *primitives* $\mathcal{D}_0$. That is, $D(x_i)$ is a subset of $\mathcal{D}_0$. In practice, these primitives can be seen as parts, or, more broadly, attributes capturing the compositional structure of the examples. Derivations are then simply sets of attribute labels. For instance, for the CUB-200-2011 dataset the set of primitives consists of items such as `beak:curved`, `beak:nidle`, *etc.*, and a derivation for the image in Figure 1a is then $\{$`beak:curved`, `wing_color:brown`, ...$\}$.

We now leverage derivations to learn a compositional representation on the base categories. Note that for the novel categories, we have only access to the category labels *without* any derivations. We will first explain measure of compositionality in representations and then introduce it as constraints to learn deep compositional representations.

3.2. Measure of Compositionality

We make use of the framework for reasoning about compositional properties of black-box image representations in [3]. A representation f is compositional over $\mathcal{D}_0$ if each

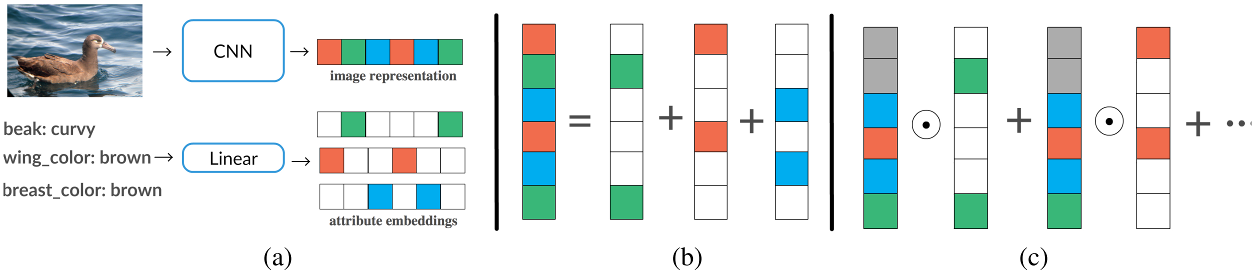

Figure 2. Overview of the proposed compositional regularization. The goal is to learn an image representation that is decomposable into parts by utilizing attribute annotations. First an image is encoded with a CNN and its attributes with a linear layer (a). We then propose two forms of regularizations: TRE shown in (b) and Soft TRE shown in (c). The former is a hard constraint forcing the image representation to be fully described by the attributes. The latter is a relaxed version that allows for a part of the representation to encode other information about the images (shown in gray).

$f(x)$ is determined by $D(x)$. For a fully compositional f there exists such a function $\hat{f}$ with parameters η that satisfies the following equality:

$$f(x_i) = \hat{f}_\eta(d_{i,1}) * \hat{f}_\eta(d_{i,2}) * ...\hat{f}_\eta(d_{i,k}), \qquad (1)$$

where $*$ is a composition operator and $D(x_i) = \{d_{i,j}\}_{j=1}^k$ is the derivation of x_i. In practice, $*$ is set to be a sum and $\hat{f}_\eta$ is implemented as a linear embedding layer [3]. With this choice of operators, Eq. (1) is fully differentiable over η and thus can be optimized with gradient descent:

$$\eta^* = \arg\min_\eta \sum_i \sigma(f(x_i), \hat{f}_\eta(D_i)), \qquad (2)$$

where σ is a differentiable distance function, such as Euclidean or Cosine similarity, and $\hat{f}_\eta(D_i) = \sum_j \hat{f}_\eta(d_{i,j})$. In other words, we aim to search for a representation $\hat{f}_\eta$ that allows an explicitly compositional model to approximate the true f as closely as possible. Intuitively, the closer this approximation is, the more compositional f is. A Tree Reconstruction Error (TRE) is then proposed in [3] to quantify the compositionality of f:

$$TRE(\mathcal{X}) = \frac{1}{n} \sum_i \sigma(f(x_i), \hat{f}_{\eta^*}(D_i)), \qquad (3)$$

which is evaluated on the validation set. A lower TRE score indicates higher compositionality.

3.3. Compositionality Regularization

Motivated by the observation made in [3] that the representations with lower TRE score (higher compositionality) exhibit better generalization behavior, we propose to convert this measure into a regularization approach. The overview of our approach is shown in Figure 2. We first observe that the objective function $\sum_i \sigma(f_\theta(x_i), \hat{f}_\eta(D_i)$ in Eq. (2) is differentiable not only with respect to η but also with respect to the parameters θ of f. This allows us to jointly optimize both f_θ and $\hat{f}_\eta$.

Next we observe that the derivations in [3] are defined on the instance level. While this fine-grained supervision is necessary for an accurate measure, it is expensive to obtain. Our key insight is that instances in any given category share the same compositional structure. Indeed, all seagulls have curved beaks and short necks, so we can significantly reduce the annotation effort by redefining derivations as $D(x_i) = D(y_i)$. One objection might be that the beak is not visible in all the images of seagulls. While this is true, we argue that such labeling noise can be ignored in practice, which is verified empirically in Section 6.1.

Hard constraints: Based on these observations, we propose a Tree Reconstruction Error Loss:

$$L_{TRE}(\theta, \eta) = \sum_i \sigma(f_\theta(x_i), \hat{f}_\eta(D(y_i))). \qquad (4)$$

It can be applied as a regularization term together with a classification loss L_{cls}, such as softmax. Intuitively, it puts a constraint on the gradient-based optimization of parameters θ, forcing it to choose out of all the representations that solve the classification problem equally well the one that is also compositional with respect to the predefined vocabulary of primitives $\mathcal{D}_0$. A visualization of L_{TRE} is presented in Figure 2b. Overall we use the following loss for training:

$$L(\theta, \eta) = L_{cls}(\theta) + \lambda L_{TRE}(\theta, \eta), \qquad (5)$$

where λ is a hyper-parameter that balances the importance of the two objectives.

Soft constraints: One crucial assumption made in Eq. (1) is that the derivations D are exhaustive. For that equation to hold, D has to capture all the aspects of the image that are important for the downstream classification task. However, even in such a narrow domain as that of CUB-200-2011, exhaustive attribute annotations are extremely expensive to obtain. In fact, it is practically impossible for larger scale datasets such as SUN [43] or ImageNet [9]. To mitigate this issue, we propose a soft version of the loss in Eq. (4) that allows for partial attribute supervision. By maximizing the similarity between each attribute's

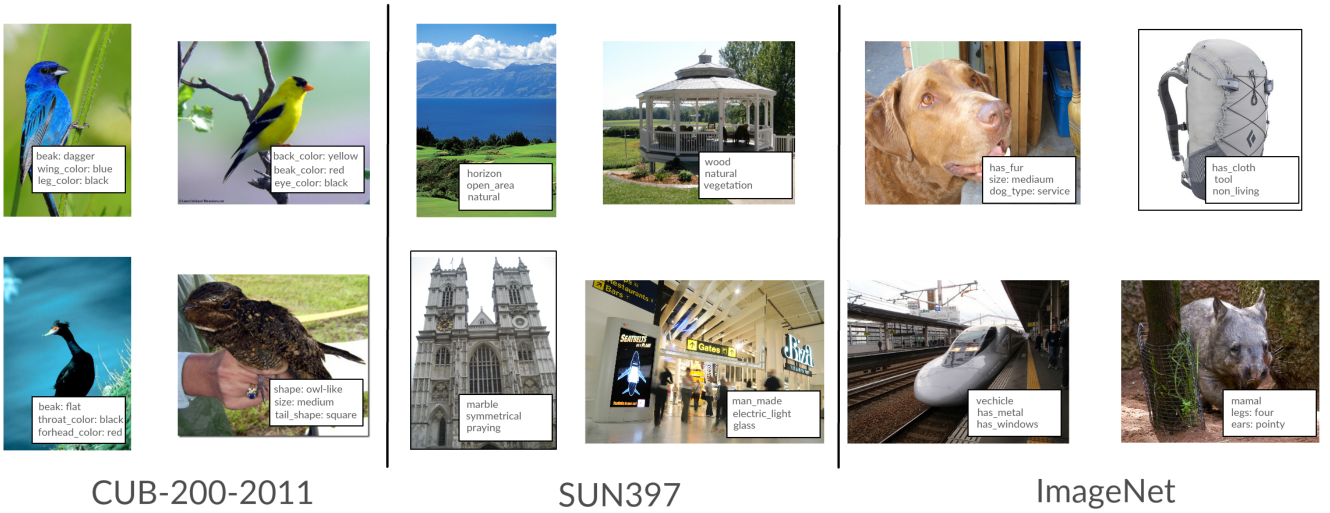

Figure 3. Examples of categories from three datasets used in the paper together with samples of attribute annotations.

embedding and the image embedding individually, we obtain a Soft Tree Reconstruction Error Loss:

$$L_{STRE}(\theta, \eta) = \sum_{i,j} \sigma'(f_\theta(x_i), \hat{f}_\eta(d_{i,j})), \qquad (6)$$

where σ' is a dot product operation (see Figure 2c for a visualization). It is easy to see that this formulation is equivalent to multi-label classification. In contrast to the hard variant in Eq. (4), it allows for a part of the image encoding f to represent the information not captured by the attribute annotations. More formally, Eq. (4) enforces the exact equality $f_\theta(x_i) = \sum_j \hat{f}_\eta(d_{i,j})$, whereas Eq. (6) enforces a softer constraint $f_\theta(x_i) = \sum_j \hat{f}_\eta(d_{i,j}) + w$, where w accounts for a part of the image representation not described by the attributes (show in grey in Figure 2c). In Section 4.3 we empirically verify that this relaxation is critical for obtaining top performance.

3.4. Collecting Attribute Annotations for ImageNet

We use three dataset for experimental evaluation: CUB-200-2011 [40], SUN397 [43] and ImageNet [9]. Samples of images from different categories of the three datasets together with their attribute annotations are shown in Figure 3. As can be seen from the figure, our method handles concrete visual attributes like material and color as well as abstract attributes, such as openness or symmetry. For the first two datasets attribute annotations are publicly available but for ImageNet we collect them ourselves. Below we describe key steps in collecting these annotations.

We heavily rely on the Wordnet [28] hierarchy both to define the vocabulary of attributes and to collect them. Firstly, we define attributes on each level of the hierarchy: every object has `size` and `material`, most of the mammals have `legs` and `eyes`, etc. This allows us do obtain a vocabulary that is both broad, intersecting boundaries of categories, and specific enough, capturing discriminative properties. Secondly, we also rely on the hierarchical properties of the attributes to simplify annotation process. In particular, the annotator is first asked about generic properties of the category, like whether it is `living`, and then all the properties specific to `non-living` categories are set to a negative value automatically. This pruning is applied on every level of the hierarchy, allowing a single annotator to collect attribute labels for 386 categories in the base split of [15] in just 3 days.

4. Experiments

4.1. Datasets and Evaluation

We use three datasets for experimental analysis: CUB-200-2011, SUN397 and ImageNet. For the first two datasets we employ attribute annotations collected externally. For ImageNet no prior attribute annotations exist, so we collect them ourselves. Below we describe each of the datasets together with their evaluation protocols in more detail.

CUB-200-2011 is a dataset for fine-grained classification [40]. It contains 11,788 images of birds labeled with 200 categories corresponding to bird species. The dataset is evenly split into training and test subsets. In addition, the authors have collected annotations for 307 attributes, corresponding to the appearance of the birds' parts, such as shape of the beak or color of the forehead. These attribute annotations have been collected on the image level via crowd sourcing. We *aggregate them on the category level* by labeling a category as having a certain attribute if at least half of the images in the category are labeled with it. We further filter out rare attributes by only keeping the ones that are labeled for at least five categories, resulting in 130 attributes used in training. For few-shot evaluation, we randomly split the 200 categories into 100 base and 100 novel categories.

SUN397 is a subset of the SUN dataset for scene recognition, which contains the 397 most well sampled categories, totaling to 108,754 images [43]. Patterson *et al.* [32] have collected discriminative attributes for these scene categories. In particular, each annotator has been shown four images representative of four random categories and then asked to name attributes that distinguish them from the other two. This resulted in a vocabulary of 106 attributes that are both discriminative and shared across scene classes. Examples include both abstract attributes, such as 'manmade', and purely visual ones, such as 'grass'. Similar to CUB, we aggregate these image-level labels for categories by labeling a category as having an attribute if half of the labeled images in the category have this attribute, and filter out the infrequent categories resulting in 89 attributes used for training. For few-shot evaluation, we randomly split the scene categories into 197 base and 200 novel categories.

ImageNet is an object-centric dataset [9] that contains 1,200,000 images labeled with 1,000 categories. The categories are sampled from the Wordnet [28] hierarchy and constitute a diverse vocabulary of concepts ranging from animals to music instruments. Defining a vocabulary of attributes for such a dataset is non-trivial and has not been done previously. We described our approach for collecting the attributes in more detail in Section 3.4. For few-shot evaluation, we use the split proposed in [15, 41].

4.2. Implementation Details

Following [15, 41], we use a ResNet-10 [16] architecture in most of the experiments, but also show results on deeper variants in Section 4.3. We add a linear layer without a nonlinearity at the end of all the networks to aid in learning a cosine classifier. The networks are first pre-trained on the base categories using mini-batch SGD, as in [15, 41, 1]. The learning rate is set to 0.1, momentum to 0.9 and weight decay to 0.0001. The batch size and learning rate schedule depend on the dataset size. In particular, for ImageNet and SUN397, we use the setting proposed in [15, 41] with a batch size of 256 and 90 training epochs. The learning rate is decreased by a factor of 10 every 30 epochs. For CUB-200-2011, which is a much smaller dataset, we use a batch size of 16 and train for 170 epochs. The learning rate is first decreased by a factor of 10 after 130 epochs, and then again after 20 more epochs. This schedule is selected on the validation set. We use both linear and cosine classifiers proposed in [14, 1] in the experiments. All the models are trained with a sofmax cross-entropy loss as L_{cls} in Eq. (5).

We observe that the proposed TRE loss slows down convergence when training from scratch. To mitigate this issue, we first pre-train a network with the standard classification loss and then fine-tune it with the TRE regularization for the same number of epochs using the same optimization parameters. For a fair comparison, baseline models are fine-tuned in the same way. We set the hyper-parameter λ in Eq. (5) for each dataset individually, using the validation set. For ImageNet and SUN397 λ is set to 8, and for CUB-200-2011 to 10. The attribute annotations are sparse, with around 10% of them being labeled as positive for any given image on average. Due to this highly imbalanced distribution of training labels, all the attribute classifiers learn to predict the negative labels. To address it, we randomly sample a subset of the negative attributes for every example in every batch to balance the number of positive and negative examples.

In few-shot training, we use the baseline proposed in [1] as our base model. In particular, we learn either a linear or Cosine classifier on top of the frozen CNN representation. Differently from [1], we learn the classifier jointly on novel and base categories. We use mini-batch SGD with a batch size of 1,000 and a learning rate of 0.1, but find that training is robust to these hyper-parameters. What is important is the number of training iterations. This number depends on the dataset and the classifier. On ImageNet we train for 100 iterations for both Cosine and linear classifiers. On SUN397 we train for 200 iterations for linear and 100 for Cosine classifier. On CUB we train for 100 iterations for linear and 40 for Cosine classifier. To select these values, we split the base categories in half, using one half as validation, and train until the top-5 performance on the validation categories stops increasing. We use the same setting to select the optimal hyper-parameters for other methods. Overall, we follow the evaluation protocol proposed in [41].

4.3. Analysis of Compositional Representations

In this section, we analyze whether the compositionallity constraints proposed in Section 3 lead to learning representations that are able to recognize novel categories from a few examples. Most of the analysis is performed on the CUB dataset [40] due to its small size. Following [14, 1], we use a Cosine classifier in the most of the experiments due to its superior performance. A qualitative analysis of the learned representations using the NetworkDissection framework of Zhou et al. [46] is also provided in the supplementary material.

Comparison between hard and soft compositional constraints: We begin our analysis by comparing the two proposed variants of compositionallity regularizations: the regular TRE in Eq. (4) and the Soft TRE in Eq. (6). Figure 4 shows the top-5 performance of the baseline model with a Cosine classifier and compares with the two variants of our compositional model on the novel categories of CUB. We perform the evaluation in 1-, 2-, and 5-shot scenarios. First we notice that the variant of the regularization based on the hard sum constraint (shown in orange) decreases the performance over the baseline. This is not surprising, since, as we mentioned in Section 3, this constraint assumes exhaustive attribute annotations which are

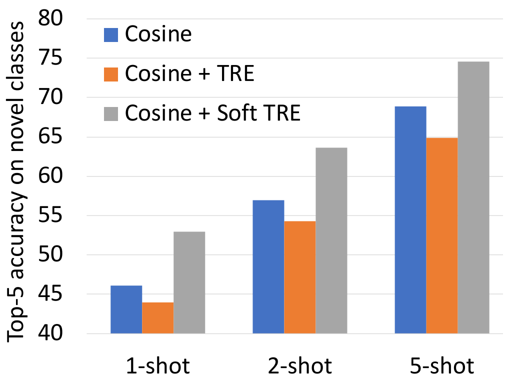

Figure 4. Comparison of the two variants of our compositionallity regularizations to a baseline on the novel categories of the CUB dataset. The y-axis indicates top-5 accuracy in a 100-way evaluation. Our soft TRE regularization achieves the best performance.

not available in these experiments. By contrast, our proposed soft constraint operationalized with attribute classifiers allows the representation to capture important information that is not described in the attribute annotation. The variant with Soft TRE regularization (shown in gray) thus improves the performance by 6.9% over the baseline in the most challenging 1-shot scenario. This confirms our hypothesis that enforcing the learned representation to be decomposable over category-level attributes allows it to generalize to novel categories with fewer examples. We use the soft variant of our approach in the remainder of the paper. Finally, the improvement of our compositional model over the baseline decreases slightly, as it sees more examples from the novel categories. This is because, as the training regime gets closer to the standard data-rich scenario, additional regularization methods become redundant. Nevertheless, our variant with soft regularization still improves over the baseline by 5.7% in a 5-shot scenario.

Ablation studies: We further analyze the compositional representation learned with the soft constraint ('Soft TRE') through extensive ablations and report the results in Table 1.

Evaluation in the challenging joint label space of base and novel classes: We notice that the observation about the positive effect of the compositionallity constraints on the generalization performance of the learned representation made above for the novel categories holds for the $novel + base$ setting (right part 'All' of the table, rows 1 and 2). In particular, our approach improves over the baseline by 4% in the 1-shot and by 4.7% in the 5-shot setting.

Cosine vs. linear classifiers: The linear classifier (denoted as 'Linear w/ comp') performs significantly worse than the Cosine variant, especially in the $novel + base$ setting. A similar behavior was observed in [14, 1] and attributed to that the Cosine classifier explicitly reduces intra-class variation among features during training by unit-normalizing the vectors before dot product operation.

	Novel			All		
	1-shot	2-shot	5-shot	1-shot	2-shot	5-shot
Cos	46.1	57.0	68.9	58.2	63.3	69.8
Cos w/ comp	53.0	63.6	**74.6**	62.2	68.1	**74.5**
Linear w/ comp	47.0	58.8	72.8	55.5	63.6	71.8
Cos + data aug	47.7	58.0	69.4	58.7	64.0	70.1
Cos w/ comp + data aug	**53.3**	**64.2**	74.3	**62.6**	**68.4**	74.4
Linear w/ comp + data aug	49.5	60.5	72.7	57.5	64.6	71.6

Table 1. Analysis of our approach: top-5 accuracy on the novel and all (*i.e.*, $novel + base$) categories of the CUB dataset. 'Cos': the baseline with a Cosine classifier, 'Cos w/ comp': our compositional representation ('Soft TRE') with a Cosine classifier, 'Linear w/ comp': our compositional representation ('Soft TRE') with a linear classifier. The variants trained with data augmentation are marked with '+ data aug'. **See ablations for detailed discussions.**

	Novel			All		
	1-shot	2-shot	5-shot	1-shot	2-shot	5-shot
Cos	35.4	45.6	56.4	52.1	56.7	61.9
Cos w/ comp	41.6	52.5	64.7	54.2	59.9	66.0
Linear w/ comp	40.6	51.9	63.5	48.7	56.2	65.1
Cos + data aug	39.9	49.7	59.7	54.2	58.5	63.5
Cos w/ comp + data aug	**44.7**	**55.5**	**65.9**	**56.0**	**61.3**	**66.9**
Linear w/ comp + data aug	39.8	49.6	59.5	48.3	53.8	60.3

Table 2. Analysis of our approach: top-5 accuracy on the novel and all (*i.e.*, $novel + base$) categories of the SUN dataset. 'Cos': the baseline with a Cosine classifier, 'Cos w/ comp': our compositional representation ('Soft TRE') with a Cosine classifier, 'Linear w/ comp': our compositional representation ('Soft TRE') with a linear classifier. The variants trained with data augmentation are marked with '+ data aug'. **See ablations for detailed discussions.**

Effect of data augmentation: Another important observation made in [1] is that, for a fair comparison, standard data augmentation techniques (*e.g.*, random cropping and flipping) need to be applied when performing few-shot learning. We report the results with data augmentation in the lower part of the table. The most important observation here is that, although all the approaches benefit from data augmentation, the improvements for the compositional model with the Cosine classifier are marginal. This further confirms our hypothesis that compositional representations demonstrate better generalization behavior, making some of the techniques designed to improve the generalization performance of standard deep learning models less important.

Larger-scale evaluation: To validate our previous observations, we now report results on a much larger SUN397 dataset [43]. Table 2 summarizes the 200- and 397-way evaluation in $novel$ and $novel + base$ setting, respectively. Overall, similar conclusions can be drawn here. One noticeable difference is that data augmentation has a more significant effect on the performance of the compositional model, but it is still less pronounced compared to the baseline.

Do we sacrifice the performance on base for novel classes? Figure 5 evaluates the accuracy of the baseline Cosine classifier (shown in blue) and our compositional representations (shown in gray) on the validation set of the base categories of CUB and SUN. The compositional representations improve the performance on the base categories as

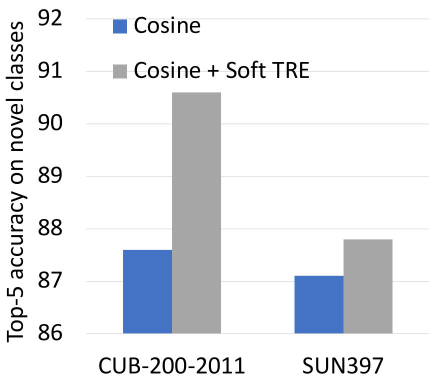

Figure 5. Comparison of our compositionallity regularization to a baseline on the base categories of the CUB dataset. The y-axis indicates top-5 accuracy on the corresponding validation set. Our approach improves the performance on the base categories as well.

well, although the improvement is significantly lower than that on the novel categories (*e.g.*, in the 1-shot scenario: 3.0% compared with 6.9% on CUB , and only 0.7% compared with 6.2% on SUN). Hence, while compositional representations are especially important when learning to recognize novel categories from a few examples, they do provide improvements even in the standard training regime.

Effect of the network depth: finally, we study the generalizability of the proposed compositionallity regularization to deeper network architectures. We conduct these experiments on the SUN dataset due to its large size and high quality of the attribute annotations. In Table 3 we compare the ResNet10 model with cosine classifier to ResNet18 and ResNet34. First of all, we notice that the improvements with respect to the baseline due to compositionallity regularization are diminishing as the network depth increases. Moreover, the shallow 'ResNet10, Cos w/ comp' model outperforms the deeper variants. We analyze this behavior and observe that the deeper models are able to learn attribute classifiers without significantly modifying their representation. This can be explained by the fact that the feature space of the last layer of the deep networks has a higher representation power. We thus propose to adapt our regularization by applying it not only to the last, but also to the intermediate layers of the network. In practice, we apply it to the outputs of all the ResNet blocks starting from the block 9. This new variant, which we denote 'Cos w/ deep comp", achieves improvements over the baseline for ResNet18 and ResNet34 comparable to those of 'Cos w/ comp' for ResNet10, which confirms that our proposed approach is indeed applicable to deeper networks.

	Novel			All		
	1-shot	2-shot	5-shot	1-shot	2-shot	5-shot
ResNet10, Cos	35.4	45.6	56.4	52.1	56.7	61.9
ResNet10, Cos w/ comp	41.6	52.5	64.7	54.2	59.9	66.0
ResNet18, Cos	37.7	47.5	58.8	53.7	58.0	63.3
ResNet18, Cos w/ comp	39.5	49.6	60.9	54.4	58.9	64.3
ResNet18, Cos w/ deep comp	43.6	54.4	65.9	56.7	62.2	68.1
ResNet34, Cos	38.5	48.8	60.2	54.3	58.8	64.4
ResNet34, Cos w/ comp	38.8	49.1	60.7	54.9	59.4	64.8
ResNet34, Cos w/ deep comp	**44.0**	**55.2**	**66.8**	**56.9**	**62.5**	**68.4**

Table 3. Evaluation of deeper architectures: top-5 accuracy on the novel and all (*i.e.*, $novel + base$) categories of the SUN dataset. 'Cos': the baseline with a Cosine classifier, 'Cos w/ comp': our compositional representation ('Soft TRE') with a Cosine classifier, 'Cos w/ deep comp': our compositional representation ('Soft TRE') with regularization applied to intermediate layers of the network. **See ablations for detailed discussions.**

4.4. Comparison to the State-of-the-Art

We now compare our compositional representations with the Cosine classifiers (denoted as 'Cos w/ comp') to the state-of-the-art few-shot methods based on meta-learning. We use the ResNet10 architecture as a backbone for all the methods. We evaluate on 3 datasets: CUB-200-1011, SUN397, and ImageNet. For CUB and SUN which have publicly available, well annotated attributes, Tables 4 and 5 show that our approach easily outperforms all the baselines across the board even without data augmentation. In particular, our full method provides around 5 to 7 point improvement on CUB and 4 to 6 point improvement on SUN for the novel classes in the most challenging 1-, 2-shot scenarios, and achieves similar improvements in the joint label space.

Table 6 summarizes the comparison on ImageNet for which we collected attribute annotations ourselves. Here we compare to the state-of-the-art methods on this dataset reported in [41], including the approaches that generate additional training examples . These results verify the effectiveness of our proposed approach in Section 3.4 of annotating attributes on the category level. The collected annotations might be noisy or less discriminative, compared with the crowd sourced annotation in [40, 32]. However, our compositional representation with a simple Cosine classifier still achieves the best performance in 1-, 2-, and 5-shot scenarios, and is only outperformed in the 10-shot scenario by Prototypical Matching Networks.

5. Conclusion

In this work we have proposed a simple attribute-based regularization approach that allows to learn compositional image representations. We validated the use of our approach in the task of learning from few examples, obtaining the state-of-the-art results on three dataset, and demonstrating that compositional representations help learn classifiers in the small sample size regime. In addition, attribute classifiers used to train our model can be used to enhance its interpretability. Compositionality is one of the key properties of human cognition that is missing in the modern deep learning methods, and we believe that our work is a precursor to a more in-depth study on this topic.

	Novel				All			
	1-shot	2-shot	5-shot	10-shot	1-shot	2-shot	5-shot	10-shot
Prototypical networks [36]	43.2	54.3	67.8	72.9	55.6	59.1	64.1	65.8
Matching Networks [39]	48.5	57.3	69.2	74.5	50.6	55.8	62.6	65.4
Relational networks [44]	39.5	54.1	67.1	72.7	51.9	57.4	63.1	65.3
Cos w/ comp (Ours)	53.0	63.6	**74.6**	**78.6**	62.2	68.1	**74.5**	**77.0**
Cos w/ comp + data aug (Ours)	**53.3**	**64.2**	74.3	78.5	**62.6**	**68.4**	74.4	76.7

Table 4. Comparison to the state-of-the-art approaches: top-5 accuracy on the novel and all (*i.e.*, $novel + base$) categories of the CUB dataset. Our approach consistently achieves the best performance.

	Novel				All			
	1-shot	2-shot	5-shot	10-shot	1-shot	2-shot	5-shot	10-shot
Prototypical networks [36]	37.1	49.2	63.1	70.0	51.3	59.0	66.4	69.3
Matching Networks [39]	41.0	48.9	60.4	67.6	50.3	54.0	60.2	64.4
Relational networks [44]	35.1	49.0	63.7	70.3	51.0	58.6	66.5	69.1
Cos w/ comp (Ours)	41.6	52.5	64.7	70.5	54.2	59.9	66.0	69.1
Cos w/ comp + data aug (Ours)	**44.7**	**55.5**	**65.9**	**71.5**	**56.0**	**61.3**	**66.9**	**70.0**

Table 5. Comparison to the state-of-the-art approaches: top-5 accuracy on the novel and all (*i.e.*, $novel + base$) categories of the SUN dataset. Our approach consistently achieves the best performance.

	Novel				All			
	1-shot	2-shot	5-shot	10-shot	1-shot	2-shot	5-shot	10-shot
Prototypical Matching Network w/ G [41]	45.8	57.8	69.0	**74.3**	57.6	64.7	**71.9**	**75.2**
Prototypical Matching Network [41]	43.3	55.7	68.4	74.0	55.8	63.1	71.1	75.0
Prototypical networks w/ G [41]	45.0	55.9	67.3	73.0	56.9	63.2	70.6	74.5
Prototypical networks [36]	39.3	54.4	66.3	71.2	49.5	61.0	69.7	72.9
Matching Networks [39]	43.6	54.0	66.0	72.5	54.4	61.0	69.0	73.7
Cos w/ comp (Ours)	47.0	58.0	68.4	72.9	55.6	63.8	71.2	74.5
Cos w/ comp + data aug (Ours)	**49.0**	**59.9**	**69.3**	73.4	**57.9**	**65.1**	71.7	74.8

Table 6. Comparison to the state-of-the-art approaches: top-5 accuracy on the novel and all (*i.e.*, $novel + base$) categories of the ImageNet dataset. Even with *noisy or less discriminative* attributes we collected, our approach achieves the best performance in 1-, 2-, and 5-shot scenarios. In addition, our approach can be potentially combined with the data generation approach [41] for further improvement.

References

[1] A closer look at few-shot classification. ICLR 2019 preprint: https://openreview.net/pdf?id=HkxLXnAcFQ. 2, 6, 7

[2] Efficient lifelong learning with A-GEM. ICLR 2019 preprint: https://openreview.net/pdf?id=Hkf2_sC5FX. 3

[3] Measuring compositionality in representation learning. ICLR 2019 preprint: https://openreview.net/pdf?id=HJz05o0qK7. 1, 2, 3, 4

[4] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for attribute-based classification. In *CVPR*, 2013. 3

[5] J. Almazán, A. Gordo, A. Fornés, and E. Valveny. Word spotting and recognition with embedded attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(12):2552–2566, 2014. 3

[6] I. Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987. 1, 2

[7] S. Dasgupta, S. Sabato, N. Roberts, and A. Dey. Learning from discriminative feature feedback. In *NIPS*, 2018. 3

[8] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-scale object classification using label relation graphs. In *ECCV*, 2014. 3

[9] 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. 2, 4, 5, 6

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. 2

[11] S. Fidler and A. Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. In *CVPR*, 2007. 2

[12] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *NIPS*, 2017. 1, 2

[13] J. A. Fodor. *The language of thought*, volume 5. Harvard University Press, 1975. 1, 2

[14] S. Gidaris and N. Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 2, 6, 7

[15] B. Hariharan and R. B. Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *ICCV*, 2017. 2, 5, 6

[16] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 2, 6

[17] D. D. Hoffman and W. A. Richards. Parts of recognition. *Cognition*, 18(1-3):65–96, 1984. 1, 2

[18] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, 2015. 1, 2

[19] V. Krishnan and D. Ramanan. Tinkering under the hood: Interactive zero-shot learning with net surgery. *arXiv preprint arXiv:1612.04901*, 2016. 2

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 2

[21] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. 1, 2

[22] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, 2017. 2

[23] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009. 3

[24] Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995. 1, 2

[25] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In *CVPR*, 2015. 2

[26] D. Marr. Vision: A computational investigation into the human representation and processing of visual information. mit press. *Cambridge, Massachusetts*, 1982. 1, 2

[27] D. Marr and H. K. Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proc. R. Soc. Lond. B*, 200(1140):269–294, 1978. 1, 2

[28] G. A. Miller. WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 5, 6

[29] I. Misra, A. Gupta, and M. Hebert. From red wine to red tomato: Composition with context. In *CVPR*, 2017. 3

[30] P. Ott and M. Everingham. Shared parts for deformable part-based models. In *CVPR*, 2011. 2

[31] A. Parkash and D. Parikh. Attributes for classifier feedback. In *ECCV*, 2012. 3

[32] G. Patterson, C. Xu, H. Su, and J. Hays. The SUN attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1-2):59–81, 2014. 6, 8

[33] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 2

[34] A. Shrivastava, S. Singh, and A. Gupta. Constrained semi-supervised learning using attributes and comparative attributes. In *ECCV*, 2012. 3

[35] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR Workshop*, 2014. 2

[36] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017. 2, 9, 11

[37] A. Stone, H.-Y. Wang, M. Stark, Y. Liu, D. S. Phoenix, and D. George. Teaching compositionality to CNNs. In *CVPR*, 2017. 3

[38] S. Thrun. Is learning the n-th thing any easier than learning the first? In *NIPS*, 1996. 1, 2

[39] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *NIPS*, 2016. 2, 9, 11

[40] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD birds-200-2011 dataset. 2011. 1, 2, 5, 6, 8

[41] Y.-X. Wang, R. Girshick, M. Hebert, and B. Hariharan. Low-shot learning from imaginary data. In *CVPR*, 2018. 2, 6, 8, 9

[42] P. H. Winston. Learning structural descriptions from examples. 1970. 1, 2

[43] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 2, 4, 5, 6, 7

[44] F. S. Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018. 2, 9

[45] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 1, 2

[46] B. Zhou, D. Bau, A. Oliva, and A. Torralba. Interpreting deep visual representations via network dissection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 1, 2, 6, 11, 13

[47] L. L. Zhu, Y. Chen, A. Torralba, W. Freeman, and A. Yuille. Part and appearance sharing: Recursive compositional models for multi-view multi-object detection. 2010. 2

6. Supplementary Material

This supplementary material provides additional experimental results and details that are not included in the main paper due to limited space. We explore the effect of plugging our compositional representations into existing few-shot learning methods, such as Prototypical Networks [36] and Matching Networks [39] in Section 6.1. We then provide an analysis of the learned representation using Network Dissection [46] in Section 6.2. Finally, we visualize the attributes used in our experiments on ImageNet together with their hierarchical structure in Sections 6.3.

6.1. Incorporation of Compositional Representations into Existing Few-Shot Learning Methods

In the main paper, we have demonstrated that a simple Cosine classifier learned on top of a frozen CNN, which was trained with our compositionality regularization, leads to the state-of-the-art results on three datasets, outperforming more complex existing few-shot classification models, such as Protoypical Networks [36] and Matching Network [39]. It is natural to ask whether training these models with our compositional representations would lead to superior results. To answer this question, we train the CNN backbone on the base categories with a linear classifier and the compositionality regularization. On top of the compositional feature, we learn these few-shot models as described in the main paper. We report the results on the novel categories of the CUB-200-2011 dataset in Table 7.

	Novel		
	1-shot	2-shot	5-shot
PN	43.2	54.3	67.8
PN w/ comp	42.6	54.7	68.1
MN	48.5	57.3	69.2
MN w/ comp	50.4	59.3	70.8
Linear w/ comp	47.0	58.8	72.8
Cos w/ comp	**53.0**	**63.6**	**74.6**

Table 7. Incorporating our compositional representations into existing few-shot classification models : top-5 accuracy on the novel categories of the CUB dataset. ‘PN’: Prototypical Networks, ‘PN w/ comp’: Prototypical Networks with our compositional representation, ‘MN’: Matching Networks, ‘MN w/ comp’: Matching Networks with our compositional representation, ‘Linear w/ comp’: our compositional representation with a linear classifier, ‘Cos w/ comp’: our compositional representation with a Cosine classifier.

We observe that using compositional representations indeed leads to an improved performance for both Prototypical Networks and Matching Networks in almost all the settings. The improvements for Prototypical Networks are marginal. The effect of compositional representations for Matching Networks is more pronounced, allowing them to outperform the linear classifier in 1- and-2-shot evaluation setting. However, our Cosine classifier remains superior to the few-shot learning methods. These experiments not only confirm the surprising effectiveness of the Cosine classifier observed in the main paper, but also show that the proposed compositional representations can generalize to other scenarios and classification models.

6.2. Analysis and Visualization of Representations

We now qualitatively and quantitatively analyze the learned representations using Network Dissection: a framework for studying the interpretability of CNNs proposed by Zhou *et al.* [46]. They first collect a large dataset of images with pixel-level annotations, where the set of labels spans a diverse vocabulary of concepts from low-level (*i.e.*, textures) to high-level (*i.e.*, object or scene categories) concepts. They then probe each unit in a pretrained CNN by treating it as a classifier for each of these concepts. If a unit achieves a score higher than a threshold for one of the concepts, it is assumed to capture the concept. The number of internal units that capture some interpretable concepts is then used as a measure of the interpretability of the network.

We compute this measure for the last layer of our networks (before the classification layer) for both the baseline Cosine classifier and the Cosine classifier with our compositionality regularization trained on SUN397. We observe that the baseline has 169 interpretable units out of 512, capturing 92 unique concepts. For our proposed compositional model, the number of interpretable units increases to 326 and the number of unique concepts increases to 109. Clearly, the proposed regularization results in learning a much more interpretable representation. To further analyze its properties, we present the distribution of the interpretable units for the baseline in Figure 6 and that for the proposed model in Figure 7, grouped by the concept type. We observe that our improvement in novel concepts mainly comes from the scene categories. This is expected, since SUN397 is a scene classification dataset.

Another interesting observation is that most of the new interpretable units seem to be duplicates of the units that already existed in the baseline model. This is due to a limitation of the Network Dissection approach. Although the vocabulary of concepts which this evaluation can identify is relatively broad, it is still limited. Several different real-world concepts thus end up being mapped to a single label in the vocabulary. To illustrate this observation and further analyze our approach, we visualize the maximally activating images for several units that are mapped by Network Dissection to the category `house` in Figure 8. The figure also shows attention maps of the units within each image. The first two units, which are shared by the baseline and the proposed model, seem to capture the general concepts of a `wooden house` and a `stone house`. However, the other three units, which are only found in the model trained with the compositionality regularization, seem to capture parts of the house, such as roof, window, and porch (see attention maps). This observation further validates that the proposed approach leads to learning representations that capture the compositional structure of the concepts.

6.3. ImageNet Attributes

In Figure 9, we visualize the hierarchical structure of the attributes which we defined for the 389 base categories in the subset of ImageNet used in our experiments. Each node (including non-leaf nodes) represents a binary attribute and edges capture the parent-child relationships between the attributes. These relationships are used in the annotation process to prune irrelevant attributes (such as number of wheels for a living thing) and thus save the annotator's time. Note that our annotated attributes might not be the perfect set of attributes for ImageNet. Nevertheless, even with these imperfect attributes, our compositionality regularization approach allowed us to achieve the state-of-the-art result.

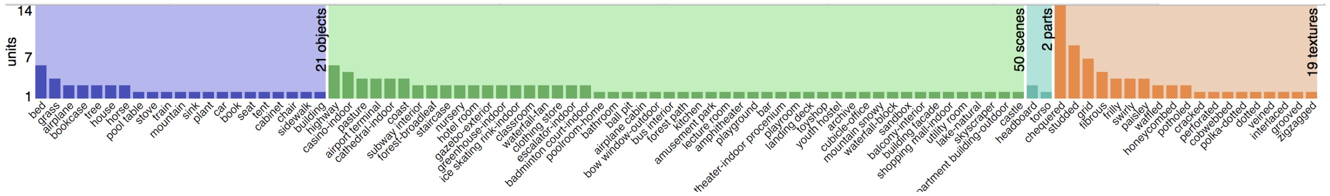

Figure 6. Distributions of interpretable units in the last layer of the baseline model trained with a Cosine classifier on SUN397 according to Network Dissection [46]. The units are grouped by the type of the concepts they represent (*i.e.*, object, scene, part, or texture). Overall, this layer has **169** interpretable units, capturing **92** unique concepts.

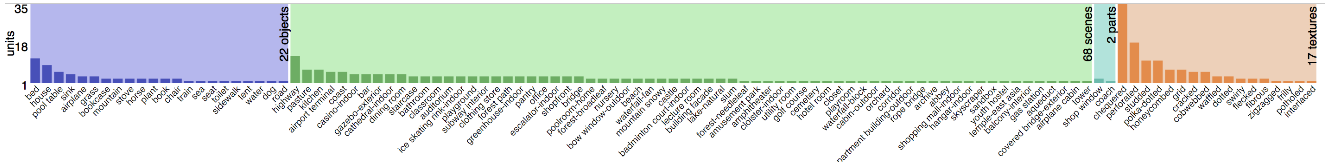

Figure 7. Distributions of interpretable units in the last layer of the model trained with a Cosine classifier and our compositionality regularization on SUN397 according to Network Dissection [46]. The units are grouped by the type of the concepts they represent (*i.e.*, object, scene, part, or texture). Overall, this layer has **326** interpretable units, capturing **109** unique concepts.

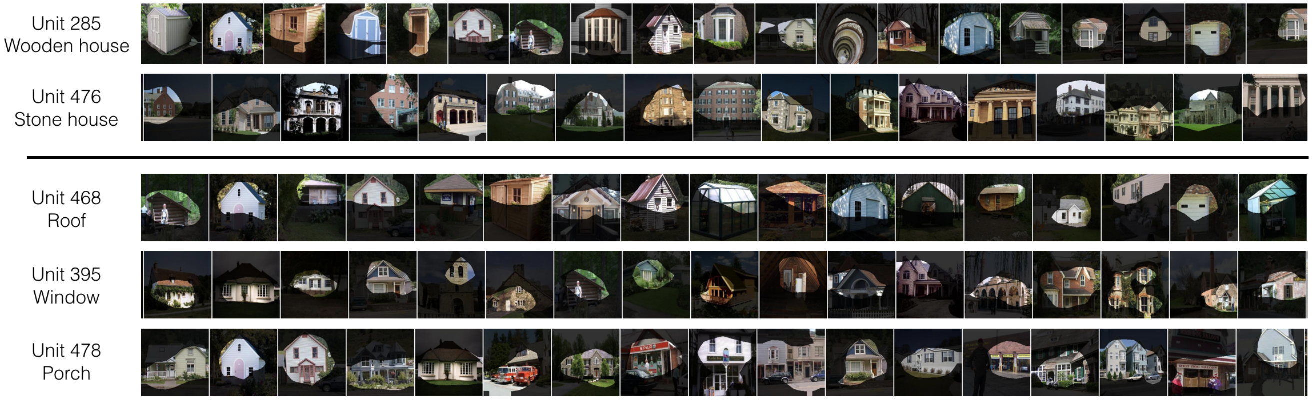

Figure 8. Top activating images for several units in the last layer of the network that are mapped to the concept `house` by Network Dissection, together with the units' attention maps. The first two units are found both in the baseline model and in the model trained with our compositionality regularization, and capture generic concepts: `wooden house` and `stone house`. The next three units are only found in the proposed model and capture parts of the house, such as `roof`, `window`, and `porch` (see attention maps).

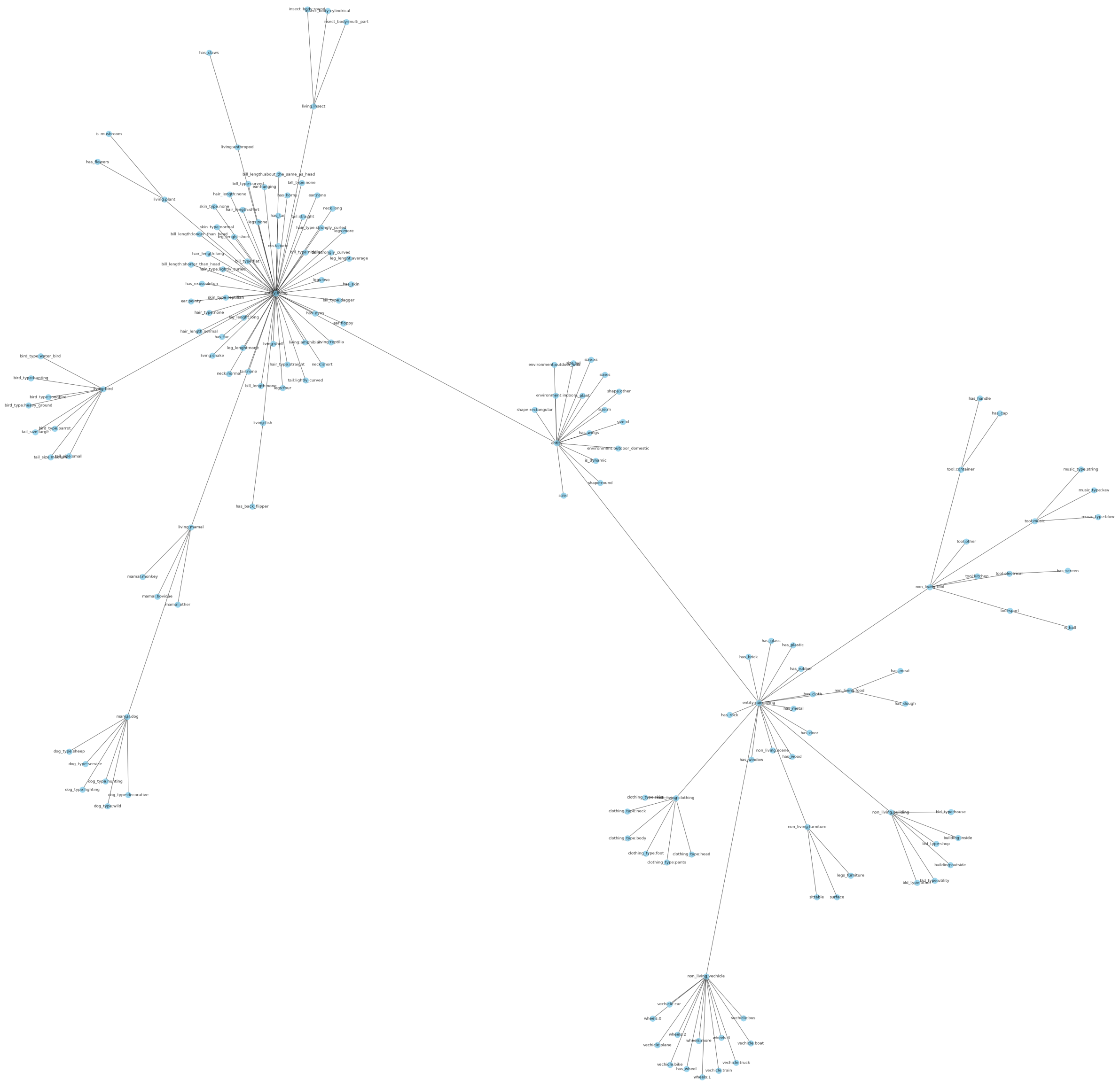

Figure 9. Attributes used in our ImageNet experiments together with their hierarchical structure. Each node represents a binary attribute and edges capture the parent-child relationships between the attributes.