Weakly-supervised Spatially Grounded Concept Learner for Few-Shot Learning

*Gaurav Bhatt[1]
gauravbhatt.cs.iitr@gmail.com
*Deepayan Das[2]
deepayan137@gmail.com
Leonid Sigal[1]
leonid.sigal77@gmail.com
Vineeth N Balasubramanian[2]
vineethnb@cse.iith.ac.in

[1] The University of British Columbia
Vancouver, Canada
[2] Indian Institute of Technology
Hyderabad
India

Abstract

One of the fundamental properties of an intelligent learning system is its ability to decompose a complex problem into smaller reusable concepts and use those concepts to adapt to new tasks. This core construct has inspired several concept-based few-shot learning approaches. However, most existing methods lack explicit semantics or require strong supervision to impose semantic structure over their concept representations. In this work, we propose a weakly-supervised and visually grounded concept learner (VGCoL), which enforces semantic structure over the learned spatial representations. The core of VGCoL is its reusable block that learns semantic concept prototypes and grounds them in an image by associating the cell features (obtained as the output of the convolution over the image) with these concept prototypes using an attention mechanism. To ensure the learned prototypes are semantic and disentangled, we introduce a regularization that aligns these prototypes with weights of the image-level concept/attribute classifiers and induces orthogonality. We illustrate that this hierarchical and semantic representation results in state-of-the-art few-shot classification performance on multiple datasets, resulting in improvements of 3–4% on CUB, SUN, and AWA2 datasets. Further, we illustrate that we can learn meaningful, interpretable, spatially coherent, and grounded concept representations despite weak class-level concept supervision.

1 Introduction

Deep learning, manifested by CNN-based [12, 16, 32] and Transformer-based [3, 20] architectures, has led to vastly improved, and in certain cases, human-level, recognition performance in circumstances where large-scale and fully-annotated data is available (*e.g.*, ImageNet [16]). However, the performance of such approaches in scenarios where only limited data exists, is considerably more modest, with most architectures overfitting and lacking the ability to generalize. [1] In a limited data regime, few-shot learning has surfaced as

[1] ***equal contribution**

the proxy for measuring the efficiency and generalization of various learning approaches and paradigms. Meta-learning [8, 22, 23, 28, 33] in particular, has emerged as a general paradigm for *learning how to learn* from related tasks. Meta-learning approaches leverage many tasks, each with a relatively small amount of data, to accumulate prior knowledge and learn adaptive strategies for transferring this knowledge to novel tasks. However, most such approaches [33, 43] lack any form of representation semantics and rely on traditional CNN-based architectures and semantics-agnostic adoptive strategies to tune representations and classifiers.

Several approaches have tried to leverage the high-level ideas, starting from early work in zero-shot learning that used class-attribute concepts [18]. More recently, semantics have been used in the form of attributes that act as a bridge between the seen and unseen classes [10, 13, 31, 37]. Most of such approaches [37, 40] assume annotation of concepts per category class, which makes them scalable – little additional annotation effort is needed. However, this also results in models that are unable to spatially localize the concepts in an image, lacking both precisions of concept definitions and visual interoperability one would like to have. To address this, [1] required strong supervision where each image instance is annotated with localized rectangular (part) concepts. Similarly, [21, 36] utilize the parts localization information to learn the correspondence between the visual features and semantics. While this results in a compositional model with grounded concepts, this comes at a price of much costlier annotation. Further, some concepts may not be appropriately grounded to a rectangular region of the bounding box, either because they map out a highly irregularly shaped region or simply have no visually observable component (*e.g.*, concepts such as “calm", “friendly", *etc.*).

To solve these challenges, we propose an interpretable spatially grounded concept learner, which only requires weakly supervised annotations at a class (or, optionally, image) level. In doing so, we develop an end-to-end framework to learn structured, reusable, semantic, and concept-based representations from limited data and leverage them to recognize novel object categories. We model concepts by learning semantic vector prototypes. We *ground* these concepts by computing the similarity of each image region, encoded by a cell feature vector from a CNN backbone, to each concept prototype. We combine these grounded concept representations (akin to attention maps) with original image features in a hierarchical and implicitly compositional manner to arrive at a classification prediction. To ensure that the learned prototypes are semantic, we introduce a regularization that both (1) aligns them with the weights of an image-level attribute classifier and (2) ensures that the concept representations are disentangled, by inducing orthogonality. We evaluate the proposed method in few-shot learning tasks on three benchmark datasets: (CUB200-2011 [39]), outdoor/indoor scene classification (SUN [26]), animal categorization (AWA2 [40]). Through extensive experiments, we demonstrate that the proposed method not only achieves SoTA few-shot classification performance but learns semantically interpretable and spatially grounded concept representations.

2 Related Work

Concept Learning. From the human cognition perspective, concept-learning is considered the building block for human intelligence that allows us to learn and reason about new concepts [4, 7, 14, 17, 25, 37]. In the past, researchers have used concept learning in computer vision by introducing feature hierarchies [7] and part-based learning [25]. There have been

some recent efforts that focus on building deep learning models which are compositional [1, 10, 37]. Researchers have also used part-based dictionaries to learn concept representations [25, 34]. [19] used part-based dictionaries to cluster the DCNNs features. In [15] a generative dictionary-based model was proposed. More recently, [10] uses part-based dictionaries along with an attention network to learn concept representations. While most existing methods encode the semantics, these representations are not spatially grounded, making them less interpretable.

Interpretable Representations. Deep convolution networks use convolution filters to learn implicit feature representations of the data [16, 42]. Visualizations of CNN features have revealed that deep networks learn a hierarchy of representations starting from the local features, such as edges, to global features such as the collection of objects in a scene [2, 16, 42]. Alternatively, explicit methods perform clustering over the part-based representations and model the spatial configuration of these parts [6, 29, 45]. Researchers have explored the constellation model family for learning expressive representations [6, 43, 45]. These models use clustering to model the spatial configuration among the cell features. ConstellationNet [43] has recently shown the benefit of combining implicit and explicit features with the constellation model family to achieve interpretable representations for few-shot learning tasks. Part-based learning models, such as the ConstellationNet [43] and CORL [11], learn interpretable spatial data features; however, they fail to encode the semantic structure as their representations focus only on the spatial locations. Consequently, the learned part-based representations are less meaningful and sometimes visually lack semantics, making the predictions less reliable for new/unseen tasks.

Few-shot Learning. In few-shot learning (FSL), researchers have leveraged attribute-based embedding as semantic prior to bridging the gap between the seen and unseen classes [24, 27, 30, 31, 37, 40, 44]. These attribute-embeddings are class-specific rather than instance-specific, making them easy to attain, thereby applicable to a wide variety of FSL datasets and tasks. In contrast, the recently proposed COMET [1] builds an interpretable model for few-shot learning using instance-specific bounding box information to define the concepts, which are then used to learn the concept prototypes. COMET's strong performance depends on the bounding box information, which is a form of strong supervision. Annotating the entire dataset with concept (part) bounding boxes are expensive, cumbersome, and requires domain expertise.

3 Problem Formulation

In few-shot classification, the aim is to take a model, trained on a dataset of samples from seen classes $\mathcal{D}^{seen}$ with abundant annotated data, and transfer / adopt this model to classify a set of samples from a disjoint set of unseen/novel classes $\mathcal{D}^{novel}$ with limited labeled data. We assume that we also have access to semantic information for each class. This information comes in the form of class-specific attributes that are available as a form of weak[2] supervision. Formally, let $\mathcal{D}^{seen} = \{(\mathbf{x}, y, \mathbf{s})\}$, where $\mathbf{x} \in X$ corresponds to an image, $y \in Y^{seen}$ corresponds to the label among the set of seen classes, and $\mathbf{s} \in S$ is a vector of semantic attributes. We follow the work of [40] and use the standard attributes that are available for all the few-shot learning datasets. Similarly, with slight abuse of notation, $\mathcal{D}^{novel} = \{(\mathbf{x}, y, \mathbf{s})\}$,

[2] "Weak" refers to lack of spatial or image-level annotations.

where the only difference is that $y \in Y^{novel}$, *i.e.*, comes from a set of novel/unseen classes.

During training, we learn a feature extractor $\mathcal{F}_\theta$ that learns the representation from the $\mathcal{D}^{seen}$ by minimizing the cross-entropy loss over the seen classes:

$$\mathcal{L}_{train} = E_{(\mathbf{x},y,\mathbf{s})\sim\mathcal{D}^{seen}} \mathcal{L}_{ce}(\mathcal{F}_\theta(\mathbf{x}), y). \tag{1}$$

We use semantic attributes $\mathbf{s}$ to regularize the compositional structure of the feature extractor $\mathcal{F}_\theta(\mathbf{x})$, to ensure generalization and interpretability.

For inference, we use the standard M-way, N-shot classification by forming *tasks* ($\mathcal{T}$), each comprising of *support set* ($\mathcal{S}$) and *query set* ($\mathcal{Q}$), constructed from $\mathcal{D}^{novel}$. Specifically, a support set consists of $M \times N$ images; N random images from each of M classes randomly chosen from Y^{novel}. The query set consists of a disjoint set of images, to be classified, from the same M classes. Following the setup of [33], we predict the class label $\hat{y}$ for $\mathbf{x}^q \in Q$ using nearest prototype $\mathbf{c}_k$ [3]:

$$\hat{y} = \arg\max_m d(\mathcal{F}_\theta(\mathbf{x}^q), c_m);\ \mathbf{c}_m = \frac{1}{N} \sum_{(\mathbf{x},y,\mathbf{s})\in\mathcal{S}, y=m} \mathcal{F}_\theta(\mathbf{x}). \tag{2}$$

Importantly, in training, our method uses semantic information ($\mathbf{s}$) for aligning the prototypes with the weights of a semantic decoder. However, semantic information is not needed at the test time to classify novel classes.

Notations. We use $\mathbf{f}$ to denote the extracted feature tensor from the convolution layer. The prototype matrix is defined as $\mathbf{P}_s$, which is used to compute the similarity matrix $\mathbf{M}^k$ for k^{th} prototype. The attention score matrix is defined as $\mathbf{A}$, while the weights of the semantic decoder are defined with trainable matrix $\mathbf{W}^s$.

4 Methodology

VGCoL architecture consists of a hierarchy of blocks, each of which has an identical structure with two key components: interaction of spatial features with learned concept prototypes obtained using semantic concept attention (SemCon-Attn) module and alignment of the concept prototypes with the weights of a learned semantic decoder (image-level concept/attribute classifier). We also introduce an orthogonal regularization that ensures the concept prototypes are disentangled. The overview of VGCoL is given in Figure 1. Given an input image $\mathbf{x}$, we extract patches (or visual features) using a convolution layer with a fixed *kernel size* and *stride*: $\mathbf{f} = \texttt{Conv}(\mathbf{x})$. This gives us visual features $\mathbf{f} \in R^{H\times W\times C}$ where H, W and C refer to the height, width and channels respectively. Next, we take columns of $\mathbf{f}$ as cell features, resulting in $H \times W$ cell feature $\mathbf{f}_{i,j} \in R^C$. These cell features encode the local information at each spatial location. Our objective is to encourage these local features to encode visual concepts, like color and texture so that when faced with novel categories they can learn to identify these visual concepts and generalize well with very few examples.

Semantic Concept Attention (SemCon-Attn). We define semantic/concept prototypes $\mathbf{P}_s = \{\mathbf{p}_k \in R^C\}_{k=1}^K$, where $\mathbf{p}_k$ denotes the prototype for the semantic concept k. Note that the dimension of each semantic prototype is equal to the image feature channels, which is C. For each semantic prototype $\mathbf{p}_k$, we compute a similarity map $\mathbf{M}^k \in R^{H\times W}$ where each element in the similarity map is computed by the dot-product between the cell feature at (i,j), $\mathbf{f}_{i,j}$,

[3] We note that the prototype $\mathbf{c}_k$ refers to class representations.

and the semantic prototype $\mathbf{p}_k$, *i.e.*, $\mathbf{M}^k_{i,j} = \mathbf{f}_{i,j} \cdot \mathbf{p}_k$. The SemCon-Attn module is introduced to improve the interaction between the concept prototypes and the visual features. Once all the similarity maps are generated, we compute the attention score over each similarity matrix as:

$$a^k_{i,j} = \texttt{Softmax}\left(\frac{1}{\sqrt{C}}\mathbf{M}^k_{i,j}\right). \tag{3}$$

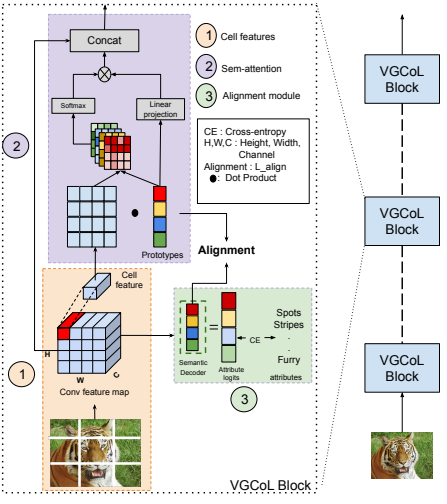

Figure 1: **Architecture of VGCoL.** We use multiple blocks of VGCoL in a deep pipeline. Our proposed method learns semantic concepts aligned with the spatial regions and is visually grounded.

The attention score map $\mathbf{A}^k = \{a^k_{i,j}\}$ gives higher weight to spatial regions containing features consistent with corresponding concept k. However, this score map is unable to model spatial prior over the concept locations or to capture spatial interactions between concepts. To address this we introduce a linear layer that computes a linear weighted sum of the concept prototypes: $\bar{\mathbf{P}}_s = \texttt{Linear}([\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_K])$, where resulting $\bar{\mathbf{P}}_s \in R^{H\times W\times K}$. We then compute a Hadamard product between the attention score matrix $\mathbf{A} = \{\mathbf{A}^k\}_{k=1}^K$ and the weighted sum of concept prototypes and concatenate ($\oplus$) the result with the original feature maps. As a result, the output of each VGCoL block is computed as:

$$VGCoL_{out} = \mathbf{f} \oplus \left(\mathbf{A} \odot \sigma\left(\bar{\mathbf{P}}_s\right)\right). \tag{4}$$

where, σ is the activation function.

Our pipeline consists of multiple blocks of VGCoL stacked over each other, giving a richer compositional representation by exploring the hierarchy of visual information (from fine-grained to coarse-grained representations). To learn semantic concepts at each level, we use different parameters for prototypes per block; that is, the semantic concepts for block l are given by $\mathbf{P}^l = \{\mathbf{p}^l_k \in R^C\}_{k=1}^K$.

We further pass the output of the VGCoL block which is of dimension $H \times W \times (C+K)$ through a 1×1 convolution to restore the original number of channels C, followed by batch normalization and ReLU. We perform standard classification (see Eq. (2)) on the output of the final VGCoL block. In essence, the sequence of VGCoL blocks define $\mathcal{F}_\theta$ in equations (1) and (2). In other words, the output of the final VGCoL block forms the image representation and is averaged to obtain prototype representations for each class $\mathbf{c}_k$ during the meta-testing.

Aligning semantic prototypes. Cell-features and SemCon-Attn module associates spatial regions with concept prototypes; however, it fails to make the prototypes semantic. It fails to associate a concept prototype with a unique (nameable) attribute. To induce semantics into the concept prototypes we introduce a semantic decoder which is defined as a simple neural net sharing the same feature backbone: $D_{\mathbf{W}^s} = \texttt{Softmax}(\texttt{Linear}(\texttt{AvgPool}(\mathbf{f}); \mathbf{W}^s))$ in

the final VGCoL block (see Figure 1). This semantic decoder outputs logits equivalent to the number of attributes present for a particular dataset. Given the image $\mathbf{x}$ as input, the semantic decoder (D_{W^s}) computes the softmax distribution over the concepts:

$$p(\mathbf{s}_k = 1|\mathbf{x}) = \frac{\exp\left(\mathbf{W}^s_{[k,:]} \cdot \texttt{AvgPool}(\mathbf{f})\right)}{\sum_k \exp\left(\mathbf{W}^s_{[k,:]} \cdot \texttt{AvgPool}(\mathbf{f})\right)}, \tag{5}$$

where $\mathbf{W}^s$ signify the trainable parameters of the attribute classifier, and the k^{th} row of the $\mathbf{W}^s \in R^{K \times C}$ is associated with the k^{th} concept. We optimize the semantic decoder using cross-entropy loss which we refer to as semantic loss or $\mathcal{L}_{sem}$ in our paper. The semantic loss ensures that each row of the parameter matrix $\mathbf{W}^s$ is associated with a particular semantic concept such as *stripes* or *spots*. To encode this semantic information into our prototypes we align the weights of the semantic decoder with the corresponding semantic prototypes using an L_1 loss which we refer to as alignment loss or $\mathcal{L}_{align}$:

$$\mathcal{L}_{align} = \sum_{l \in L} \sum_{k \in K} ||\mathbf{W}^s_{[k,:]} - \mathbf{p}^l_k||_1. \tag{6}$$

We align the concept prototypes with the weights of the semantic decoder at each level of the hierarchy, which results in the hierarchical visual grounding of concepts.

Decorrelating semantic concepts. Visual attributes frequently co-occur since they are highly correlated. It becomes difficult to prevent the entanglement of concepts at times due to their high frequency of occurrence together. To encourage the disentanglement of visually grounded concepts we introduce an orthogonality constraint. The constraint is inspired by [37] and takes the form $\mathcal{L}_{ortho} = |\mathbf{W^s} \cdot (\mathbf{W^s})^T - \mathbf{I}|$, where $\mathbf{I}$ is the identity matrix.

Loss objective. The network is jointly trained to optimize all losses:

$$Loss = \mathcal{L}_{cls} + \alpha \mathcal{L}_{sem} + \beta L_{align} + \lambda \mathcal{L}_{ortho}, \tag{7}$$

where α, β, and λ are the hyper-parameters giving relative weighting of terms. Note, $\mathcal{L}_{cls} = \mathcal{L}_{train}$ from Eq. (1). The hyperparameters defined in Eq. (7) are the weights given to each loss term, namely, semantic loss, alignment loss, and orthogonal loss.

5 Experiments

Datasets. We evaluate the performance of the proposed model on three benchmark datasets for few-shot learning: Caltech-UCSD-Birds (CUB) [39], Scene classification with attributes (SUN) [26], Animals with Attributes 2 (AWA2) [40]. CUB consists of 11,788 images from 200 bird classes with each class further having 312 attributes corresponding to different body parts of the birds. However, the attributes for CUB are instance specific and noisy [14, 37]. We use the 112 attributes as mentioned in [14] in our experiments. SUN [26] consists of 14,340 images from 717 scene classes while AWA2 contains 37,322 images divided into 50 classes. These datasets have 102 and 85 attributes, respectively. The semantic vectors are manually annotated for each class and are provided in the official repository for the CUB, SUN, and AWA2 datasets. Please note that the semantic vectors are class-specific rather than sample-specific. That is, for all samples in a given class there is a single semantic vector. We follow the standard few-shot splits provided in [40].

Implementation Details. We show the effectiveness of our method on two widely used backbone architectures for few-shot learning, namely Conv-4 and Resnet-12. Conv-4 contains 4 blocks, with each block consisting of a 3×3 convolutional layer, a batch normalization layer, and a ReLU followed by a max-pooling layer. Further, each of the convolutional layers has 64 filters. The ResNet-12 network has 4 residual blocks, with each block in turn consisting of three convolutional blocks. Each convolutional block contains a 3×3 convolutional layer, a batch normalization layer, ReLU, and max-pooling. The filter sizes are set to 64, 128, 256, and 512 respectively for each residual block. The number of neurons in the final layer of the semantic decoder is equivalent to the number of attributes for each corresponding dataset.

5.1 Few-shot classification

We now compare our proposed method to a number of state-of-the-art few-shot methods COMET [1], ProtoNets [33], RelationNets [35], CompoNets [37], MatchingNets [38], and ConstellationNet [43]. This includes approaches that leverage interpretable representations, meta-learning, and prototype-based learning. Our method, which learns compositional representations in the form of semantic prototypes, achieves better 1-shot and 5-shot performance in nearly all settings (Table 1). This suggests that our method can inject interpretability into the network without loss in performance. With a deeper network (ResNet-12), the performance surpasses all existing methods, suggesting that semantic information captured by VGCoL is helpful for few-shot classification.

On the CUB dataset, the VGCoL's performance is comparable to COMET [1] which uses strong supervision in the form of bounding boxes for each image sample. This suggests that semantic attention and alignment help VGCoL (which is weakly supervised) achieve similar performance to COMET (strongly supervised). COMET fails to utilize non-visual concepts (such as activity, and behavior), while VGCoL can associate and ground all concepts to visual space, resulting in better generalization. Further, COMET's applicability is limited to datasets where all part annotations are available, which is why COMET cannot work on SUN or AWA2. In contrast, VGCoL is applicable in most practical scenarios with weak supervision of class attributes.

Further, we compare VGCoL with several attribute-based few-shot methods: [9, 24, 30, 31, 41, 42] (see middle block of Table 3). Most of these methods use a pretrained ResNet-101 backbone, so we also report the performance of VGVoL with this backbone.

Fine-tuned VGCoL. We also present a strategy to fine-tune VGCoL on the support set before the meta-testing phase. During fine-tuning, we freeze all the layers of VGCoL except the semantic decoder and minimize the sum of semantic loss and alignment loss. This gives VGCoL some prior knowledge of the semantic attributes on the support set of novel classes. We show the results of fine-tuning in Table 1. We can see an improvement in performance on all datasets, surpassing the performance of all existing methods by 3–4% on CUB, SUN, and AWA2.

5.2 Ablation study

In Table 2 we present an ablation study to understand the importance of each component of our model. We show performance with Conv-4 and ResNet-12 backbone. Here, the $\mathcal{L}_{fsl}$ is the setup that uses only the cell features and SemCon-Attn for training on $\mathcal{D}^{novel}$ classes.

Method	CUB		SUN		AWA2	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ProtoNets [33]	43.4	67.8	37.1	63.1	41.9 ± 0.8	54.86 ± 0.7
MatchingNets [38]	48.5	69.2	41.0	60.4	-	-
RelationNets [35]	39.5	67.1	35.1	63.7	-	-
COMET [1]	67.9 ± 0.9	85.3 ± 0.5	-	-	-	-
CompoNets [37]	53.6	74.6	45.9	67.1	-	-
ConstellationNet - Conv-4	67.8 ± 0.9	85.7 ± 0.6	49.7 ± 0.8	68.2 ± 0.7	44.4 ± 0.7	60.0 ± 0.6
ConstellationNet - ResNet-12	70.1 ± 0.8	86.3 ± 0.5	50.3 ± 0.8	70.1 ± 0.7	47.3 ± 0.7	63.3 ± 0.6
Ours - Conv-4	66.7 ± 0.5	83.1 ± 0.6	52.5 ± 0.8	69.1 ± 0.7	45.7 ± 0.7	61.5 ± 0.6
Ours - ResNet-12	70.5 ± 0.3	87.3 ± 0.5	54.6 ± 0.7	71.2 ± 0.6	47.5 ± 0.6	65.9 ± 0.6
Ours - Conv-4 finetune	66.8 ± 0.9	83.2 ± 0.6	54.4 ± 0.8	71.5 ± 0.7	46.6 ± 0.3	62.1 ± 0.7
Ours - ResNet-12 finetune	$\mathbf{73.8 \pm 0.8}$	$\mathbf{90.0 \pm 0.3}$	$\mathbf{57.9 \pm 0.7}$	$\mathbf{75.6 \pm 0.7}$	$\mathbf{50.1 \pm 0.9}$	$\mathbf{70.0 \pm 0.9}$

Table 1: Comparison with existing approaches on the task of few-shot learning. Here, we evaluate the performance of the proposed model on three benchmark datasets for FSL.

Model	backbone	1-shot	5-shot
$\mathcal{L}_{fsl}$	Conv-4	62.9 ± 0.9	81.6 ± 0.6
$\mathcal{L}_{fsl}+\mathcal{L}_{sem}+\mathcal{L}_{align}$	Conv-4	63.1 ± 0.9	81.9 ± 0.6
$\mathcal{L}_{fsl}+\mathcal{L}_{sem}+\mathcal{L}_{align}+\mathcal{L}_{ortho}$	Conv-4	66.7 ± 0.5	83.1 ± 0.6
$\mathcal{L}_{fsl}$	ResNet-12	68.7 ± 0.8	86.4 ± 0.6
$\mathcal{L}_{fsl}+\mathcal{L}_{sem}+\mathcal{L}_{align}$	ResNet-12	70.0 ± 0.9	87.0 ± 0.5
$\mathcal{L}_{fsl}+\mathcal{L}_{sem}+\mathcal{L}_{align}+\mathcal{L}_{ortho}$	ResNet-12	70.5 ± 0.3	87.3 ± 0.5

Table 2: **Ablations.** Ablation study of each component of the VGCoL on the CUB dataset.

Method	Backbone	1-shot	5-shot
ProtoNets [33]	ResNet-12	43.4	67.8
COMET [1]	ResNet-12	67.9 ± 0.9	85.3 ± 0.5
CompoNets [37]	ResNet-12	53.6	74.6
ConstellationNet [38]	Conv-4	61.2 ± 0.9	81.0 ± 0.6
ConstellationNet [38]	ResNet-12	67.8 ± 0.8	85.3 ± 0.5
CADA VAE [28]	ResNet-101 pretrained	55.2	63.0
DRAGON [39]	ResNet-101 pretrained	55.3	63.5
Analogy [20]	ResNet-10	56.5	78.0
Imprinted [24]	ResNet-12	48.5	80.0
f-VAEGAN-d2 [40]	ResNet-101 pretrained	76.1	83.4
APN+f-VAEGAN-D2 [40]	ResNet-101 pretrained	77.8	84.8
TF-VAEGAN [40]	ResNet-101 pretrained	75.6	83.5
APN+TF-VAEGAN [40]	ResNet-101 pretrained	77.1	85.2
Ours - VGCoL	ResNet-101	75.1 ± 0.82	88.9 ± 0.4
Ours - VGCoL	ResNet-101 pretrained	$\mathbf{89.3 \pm 0.6}$	$\mathbf{95.7 \pm 0.2}$

Table 3: Comparison with existing approaches with varying backbones on CUB dataset.

Next, we add the $\mathcal{L}_{sem}$ and the $\mathcal{L}_{align}$ where we align the prototypes with weights matrix $\mathbf{W}^s$. The addition of a semantic alignment module improves both 1-shot and 5-shot performance. With the addition of orthogonal loss, the performance for the ResNet-12 backbone does not change much, but Conv-4 performance improves. The $\mathcal{L}_{align}$ and $\mathcal{L}_{ortho}$ are responsible for making learned concepts semantic. With the addition of $\mathcal{L}_{align}$ and $\mathcal{L}_{ortho}$ VGCoL learns visually coherent semantic concepts as we will discuss next.

5.3 Visualizing concept activation maps

We present the visualization of activation maps corresponding to multiple concept prototypes present in an image. The activation maps are computed by upsampling the similarity matrix $\mathbf{M}$ corresponding to a certain concept prototype to the original image size using bilinear interpolation. The region with the highest excitation localizes that particular concept. As shown in Figure 2 (a), the VGCoL learns to focus on semantically relevant regions around certain prototypes. The proposed method learns to ground visually discernible traits such as *furry*, *stripes*, *hooves*, and the semantic meaning of visually indiscernible traits such as *swims* and *walks*. Here, VGCoL learns to map the *swims* concept to water and the *walks* concept to the ground. This demonstrates the effectiveness of the proposed approach in visually grounding semantic concepts without any supervision about attribute localization.

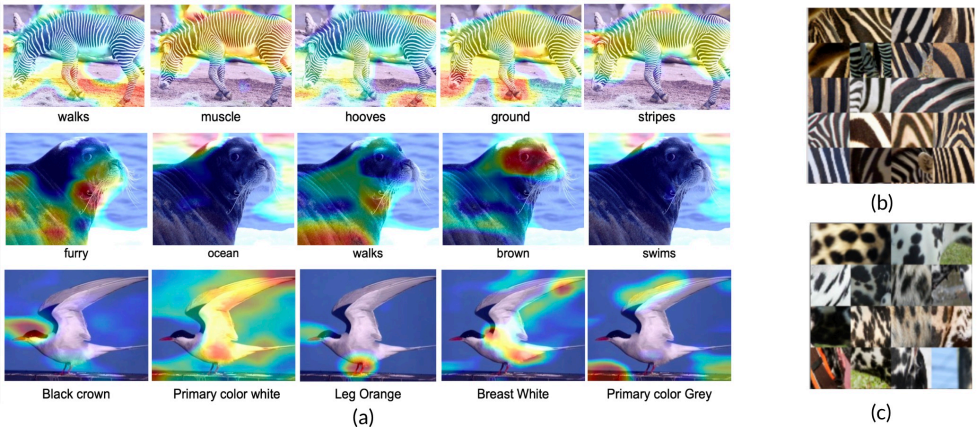

Figure 2: **(a)** Visualizing the similarity matrix **M**. Three samples and 5 concepts are illustrated. Red corresponds to strong grounding of the concept. **(b) and (c)** shows extracted patches around the concepts *stripes* and *spots*, respectively.

Figure 3: **Zero-shot segmentation results on novel/unseen classes.** Here, the middle row shows the union of maximum activation (heat-maps) which we get by aggregating top occurring concepts. The bottom row is the segmented masks.

To demonstrate the visual semantics achieved by VGCoL, we visualize the patches around the prototypes obtained from the last module of our method on the AWA2 dataset, as shown in Figure 2(b,c). It clearly shows that the proposed method is able to correctly distinguish between concepts *stripes* and *spots* which are spatially similar, but semantically different. During training, each attribute gets associated with a particular concept prototype which makes VGCoL prototypes *identifiable* and easy to interpret.

5.4 Zero-shot segmentation

Here we present an interesting downstream task of zero-shot segmentation using VGCoL. We use our pre-trained VGCoL method to extract activation maps corresponding to each

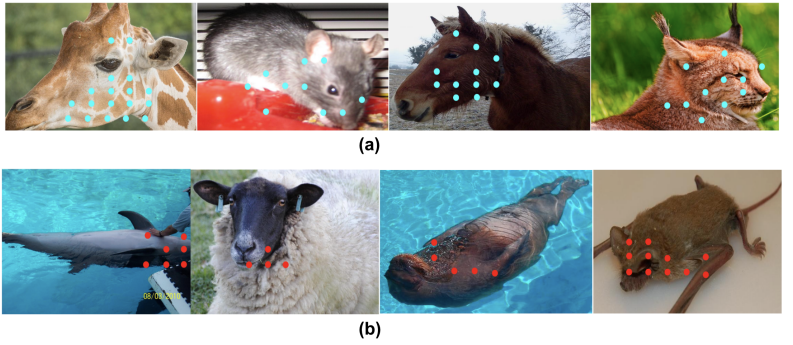

Figure 4: **Failure cases of VGCoL on the AWA2 dataset.** Top-to-bottom: (a) shows the visual grounding of imbalanced *quadra-pedal* concept; (b) demonstrates grounding of visually indiscernible and imbalanced *new-world* concept.

attribute for given novel/unseen class images. We filter out those activations for whom the value is less than a pre-defined threshold. Next, we combine these concept-specific activation maps by taking an average. The resultant activation map has regions of high activation corresponding to different parts, as shown in Figure 3. Finally, we generate an approximate segmentation mask around the given animal/bird by setting another threshold value which helps us capture the regions with high excitation. This gives us a zero-shot segmentation that uses concept knowledge to segment unseen animal/bird categories. We show the qualitative results in Figure 3.

5.5 Failure Cases

We observe that imbalance among concepts is a challenge for VGCoL as our method relies on class-specific attribute information. For instance, the concept *quadra-pedal* (meaning walks on four legs) is present for most of the animal classes in the AWA2 dataset. The VGCoL model has to visually ground this attribute even if the legs are not visible in the image (as shown in Figure 4(a)). This causes the VGCoL model to wrongly localize the spatial region for an imbalanced attribute such as *quadra-pedal*. Another failure case arises from some visually indiscernible attributes such as *new-world* which is difficult to semantically align with a spatial region (shown in Figure 4(b)).

6 Conclusions

This work presents an end-to-end weakly supervised and visually grounded concept learner for few-shot learning. The proposed method improves few-shot performance across benchmark datasets and generates semantically coherent prototype representations. This, in turn, makes the VGCoL predictions interpretable and thus reliable for generalization over the novel/unseen classes. We show the effectiveness of the proposed method for semantic concept visual grounding and the potential for a zero-shot object segmentation task.

References

[1] Kaidi Cao, Maria Brbic, and Jure Leskovec. Concept learners for few-shot learning. In *International Conference on Learning Representations (ICLR)*, 2021.

[2] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnett, Jonathan Su, and Cynthia Rudin. This looks like that: deep learning for interpretable image recognition. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*, 2021.

[4] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3723–3731, 2019.

[5] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 28(4):594–611, 2006.

[6] Robert Fergus, Pietro Perona, and Andrew Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[7] Sanja Fidler and Ales Leonardis. Towards scalable representations of object categories: Learning a hierarchy of parts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[8] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, pages 1126–1135, 2017.

[9] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE international conference on computer vision*, 2017.

[10] Ju He, Adam Kortylewski, and Alan Yuille. Compas: Representation learning with compositional part sharing for few-shot classification. *arXiv preprint arXiv:2101.11878*, 2021.

[11] Ju He, Adam Kortylewski, and Alan Yuille. Corl: Compositional representation learning for few-shot classification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3890–3899, 2023.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] Yao-Hung Hubert Tsai, Liang-Kang Huang, and Ruslan Salakhutdinov. Learning robust visual-semantic embeddings. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3571–3580, 2017.

[14] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning (ICML)*, pages 5338–5348. PMLR, 2020.

[15] Adam Kortylewski, Qing Liu, Huiyu Wang, Zhishuai Zhang, and Alan Yuille. Combining compositional models and deep networks for robust object classification under occlusion. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1333–1341, 2020.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems (NeurIPS)*, 25:1097–1105, 2012.

[17] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017.

[18] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[19] Renjie Liao, Alexander Schwing, Richard S Zemel, and Raquel Urtasun. Learning deep parsimonious representations. In *International Conference on Neural Information Processing Systems (NeurIPS)*, pages 5083–5091, 2016.

[20] Ze Liu, Yutong Lin, Yue Cao1, Han Hu1, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[21] Changsheng Lu and Piotr Koniusz. Few-shot keypoint detection with uncertainty learning for unseen species. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19416–19426, 2022.

[22] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. *International Conference on Learning Representations (ICLR)*, 2018.

[23] Tsendsuren Munkhdalai, Xingdi Yuan, Soroush Mehri, and Adam Trischler. Rapid adaptation with conditionally shifted neurons. In *International Conference on Machine Learning (ICML)*, pages 3664–3673, 2018.

[24] Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *European Conference on Computer Vision*. Springer, 2020.

[25] Patrick Ott and Mark Everingham. Shared parts for deformable part-based models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1513–1520, 2011.

[26] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2751–2758, 2012.

[27] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[28] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *International Conference on Learning Representations (ICLR)*, 2019.

[29] Ruslan Salakhutdinov, Joshua B Tenenbaum, and Antonio Torralba. Learning with hierarchical-deep models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(8):1958–1971, 2012.

[30] Dvir Samuel, Yuval Atzmon, and Gal Chechik. From generalized zero-shot learning to long-tail with class descriptors. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 286–295, 2021.

[31] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8247–8255, 2019.

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

[33] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[34] Yihong Sun, Adam Kortylewski, and Alan Yuille. Weakly-supervised amodal instance segmentation with compositional priors. *arXiv preprint arXiv:2010.13175*, 2020.

[35] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1199–1208, 2018.

[36] Luming Tang, Davis Wertheimer, and Bharath Hariharan. Revisiting pose-normalization for fine-grained few-shot recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14352–14361, 2020.

[37] Pavel Tokmakov, Yu-Xiong Wang, and Martial Hebert. Learning compositional representations for few-shot recognition. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6372–6381, 2019.

[38] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems (NeurIPS)*, 29:3630–3638, 2016.

[39] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

[40] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence (TPAMI*, 2018.

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

[42] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1492–1500, 2017.

[43] Weijian Xu, Yifan xu, Huaijin Wang, and Zhuowen Tu. Attentional constellation nets for few-shot learning. In *International Conference on Learning Representations (ICLR)*, 2021.

[44] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for any-shot learning. *IJCV*, pages 1–19, 2022.

[45] Song-Chun Zhu and David Mumford. *A stochastic grammar of images*. Now Publishers Inc, 2007.