Prototypical Matching and Open Set Rejection for Zero-Shot Semantic Segmentation

Hui Zhang* Henghui Ding*†
Nanyang Technological University, Singapore

Abstract

The DCNN methods in addressing semantic segmentation demand vast amount of pixel-wise annotated training samples. In this work, we present zero-shot semantic segmentation, which aims to identify not only the seen classes contained in training but also the novel classes that have never been seen. We adopt a stringent inductive setting in which only the instances of seen classes are accessible during training. We propose an open-aware prototypical matching approach to accomplish the segmentation. The prototypical way extracts the visual representations by a set of prototypes, making it convenient and flexible to add new unseen classes. A prototype projection is trained to map the semantic representations towards prototypes based on seen instances, and will generate prototypes for unseen classes. Moreover, an open-set rejection is utilized to detect objects that do not belong to any seen classes, which greatly reduces the misclassification of unseen objects into seen classes due to the lack of seen training instances. We apply the framework on two segmentation datasets, Pascal VOC 2012 and Pascal Context, and achieve impressively state-of-the-art performance.

1. Introduction

In semantic segmentation [8, 13, 14, 12] which aims to classify each pixel in a given image, great challenges are induced by the demand for massive training samples with pixel-wise annotations. In the field of image recognition facing the same dilemma, zero-shot learning (ZSL) [32, 41, 18] is proposed, where the classification model is trained to accommodate unseen objects using knowledge learnt from seen classes. Similarly, zero-shot segmentation (ZSS) [47, 5, 20, 31, 36, 26, 21] is also proposed in semantic segmentation. The goal of ZSS is to generate segmentation mask for objects of both seen (with annotated instances) and unseen categories (that have never been seen in training), as shown in Figure 1.

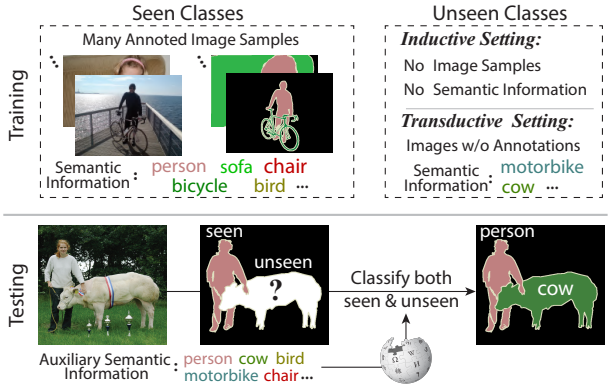

Figure 1. Zero-shot Semantic Segmentation (ZSS). The purpose of ZSS is to transfer knowledge learnt from seen classes to unseen classes (*i.e.*, never-seen in training). In testing, ZSS classifies each pixel to one of the seen classes or newly-added unseen classes.

The preliminary ZSL setting does not need to distinguish between the seen and unseen classes, which is unrealistic and contradicts the real-world conditions of recognition. A practical generalized zero-shot learning (GZSL) [42] is then proposed since image samples from seen and unseen classes often appear together and it is important to recognize both groups simultaneously. Zero-shot segmentation (ZSS) is naturally an analogue of GZSL, since the given image for segmentation already contains diverse categories. "ZSS" in this paper stands for the generalized case.

In ZSS, an important source of information is the semantic representation - semantic information encoded by high-dimensional vectors. The semantic information can include automatically-extracted word vectors, manually-defined attribute vectors, context-based embedding, or their combinations. Each class (either seen or unseen) has its own semantic representation. The ways of utilizing unseen information separates ZSS into two settings: inductive setting and transductive setting (see Figure 1). In inductive training, the visual features and semantic representations of only the seen classes are available; while in transductive training, one can

*Equal contribution.
†Henghui Ding is the corresponding author.

access the semantic representations (and sometimes images without annotations) of unseen classes, apart from the visual features and semantic representations of seen classes. Although several methods (*e.g.*, ZS3 [5], CaGNet [20], and CSRL [33]) are developed under transductive learning, this setting is indeed impractical because it violates the unseen assumption and significantly reduces challenges. Nevertheless, both settings have reached a consensus that the ground truth of unseen classes should never be present or utilized during training. Therefore, misuse of ground truth of unseen classes in training the classifier should be prevented.

In this work, we obey a strict inductive setting, in which only the information (*i.e.*, semantic representations, visual features and ground truth) of seen classes are available during training. In ZSS, to transfer knowledge from seen to unseen classes, a mapping function from semantic space to visual feature space is expected. For instance, ZS3 [5] trained such a generator on seen classes and used it to produce fake visual features for unseen classes. These fake features are then used to fine-tune the classifier (trained on the seen classes in advance). However, it is controversial that their classifier training uses pairs of fake feature and corresponding label, which requires practically inaccessible information (*e.g.*, the ground-truth of unseen pixels, the number and the attributes of unseen classes). Besides, the trained model can no longer handle newly-added unseen classes, showing a fixed capacity. In this work, to break this limitation, we employ a prototypical way instead of the convolutional classifier way. We extract high-level visual representations by training prototypes that correspond to classes one-to-one. Segmentation is performed to each pixel by finding the closest prototype to its own features. The mapping between semantic information and visual features is thereby bridged by the prototype vector. A lightweight projection network is proposed to learn the mapping from semantic information to prototypes. Any new unseen class can be flexibly added during testing, by projecting its semantic description to a prototype and adding the new prototype to existing ones.

Bias problem also imposes a significant challenge to ZSS. There exists a natural bias towards the seen classes when the model trained merely on seen classes is expected to segment both seen and unseen classes. Misclassifications easily take place when objects in unseen classes are similar to a seen class, to any extent or in any form. Previous ZSS works pay little attention to this issue, resulting in inaccurately predicting the unseen class as the seen class. In this work, we propose an open-set rejection (OSR) module as a detector to identify which group (seen or unseen) the test sample belongs to. Concretely, the OSR classifies an object directly to a certain seen class, or a general "unknown" class. If an object is predicted as the "unknown", a certain unseen class label that has the closet prototype will be assigned to it. Therefore, during testing, the possible classes into which unseen objects can be classified is constrained.

The main contributions of this works are concluded from four aspects:

- We clarify the inductive and transductive setting of ZSS and perform ZSS in challenging inductive setting.
- We employ prototypical matching in ZSS, to bridging the semantic and visual information, and make it flexible to add new unseen categories during testing.
- We introduce the open-set rejection into ZSS for the first time, effectively mitigating the bias problem and enhancing the parsing performance.
- We achieve new state-of-the-art performance on ZSS.

2. Related Work

2.1. Zero-Shot Segmentation

The term zero-shot semantic segmentation first appears in SRPNet [47], which achieves knowledge transfer from trained classes to novel classes utilizing the similarity between different categories. This method is obviously biased towards the seen classes, however limited manifestation is built up in their test setting. Almost simultaneously, ZS3Net [5] proposes to generate pixel-wise fake features for unseen classes to fine-tune the classifier in a semantic segmentation network. However, they improperly use the ground truth of unseen objects since they has to specify the label of pixels belonging to unseen classes during the fine-tuning. Gu et al. [20] make some improvements by a contextual module, aiming to generate diverse and context-aware features from semantic information. Li et al. [33] propose a Consistent Structural Relation Learning (CSRL) approach to harness the similarity of category-level relations and to learn a better visual feature generator. As inspired by few-shot segmentation, Kato et al. [31] propose a two-branch (segmentation branch and conditioning branch) architecture for ZSS. In their testing, the test set contains only unseen classes, which is unrealistic and significantly reduces the challenge. Different from few-shot segmentation that predicts masks of the same classes in query image and its support annotated image, zero-shot segmentation aims at transferring knowledge from seen classes to unseen classes (non-overlap in between) via the bridge between semantic and visual information. The challenges of ZSS come not only from the domain shift but also the obvious bias to seen classes. Most existing works are devoted to solving the domain transfer problem in ZSS. Specifically, Lv et al. [36] mitigate the problem of strong bias towards seen classes by a transductive approach, where both labeled seen images and unlabeled unseen images are utilized for training. Hu et al. [26] define another challenge that is induced by the noisy and outlying training samples from seen

classes, and address it with Bayesian uncertainty estimation. Another perspective of ZSS is to generate synthetic visual features for unseen classes, as demonstrated by Gu et al. [21]. They generate synthetic unseen features by utilizing category-level semantic representations and pixel-wise contextual information.

2.2. Zero-Shot Learning

The existing ZSL works can be divided into classifier-based methods and instance-based methods. The classifier-based methods again can be divided into correspondence-based [1, 10, 34] and relationship-based [29, 19, 51]. The correspondence-based methods capture a correspondence between visual features and class label embeddings, *i.e.*, they aim at building a general mapping function from semantic embedding to visual space. The relationship-based methods aim to simulate the relationships among classes, so that the relationships observed in the semantic spaces could be directly transferred to the feature space. The instance-based methods [45, 15, 49] are devoted to retrieving some instances (visual features) for unseen classes, although they are not provided in training set. Such instances could be acquired through projection functions, borrowing from seen classes and synthesizing methods. Although the instance-based methods are effective in zero-shot learning, it is hard to be extended to ZSS since synthesizing pixel-level instances for segmentation tasks are much harder than synthesizing image-level instances for recognition tasks.

2.3. Generalized Zero-Shot Learning

The GZSL is firstly proposed by Scheirer et al. [42]. Afterwards, Chao et al. [7] empirically show that the ZSL methods cannot function well under the GZSL setting. Because of ZSL's overfitting on seen categories, there is a strong bias problem of classifying all testing instances of the unseen as the seen class. Calibration techniques [6, 9, 22, 27] are proposed to alleviate this issue, by attempting to achieve a balance between the classification of seen group and unseen group.

Detector-based methods [4, 17] become another branch, which aims to determine whether a testing image belongs to the unseen group. This scheme restricts the number of candidate categories by narrowing down the group (seen or unseen) to which the test sample belongs. For example, Socher et al. [45] believe that compared with the seen categories, the unseen ones may exceed the scope of distribution. The testing instances from unseen group are regarded as outliers of the training (*i.e.*, seen) distribution. Later, auto-encoder-based [4], entropy-based [38] and probabilistic-based [46] detectors are proposed for out-of-distributions (OODs), *i.e.*, the unseen classes. Liu et al. [35] use the temperature scaling [25] and an entropy-based regularizer to make the unseen classes more confident and the seen classes less confident. We follow the way of detector-based methods and design an open set rejection module that can identify whether a pixel belongs to the seen classes.

2.4. Open-Set Learning

Open set learning (OSL) [43] assumes that deficient knowledge exists during the training stage, and aims to recognize samples belonging to known classes and identify unknown samples simultaneously during the testing stage. Most traditional methods are based on support vector machine [43, 44, 28], nearest neighbor [30, 2], sparse representation [50], etc. Recently, deep-learning-based OSL methods [43, 3, 24, 52, 40] greatly advance the state-of-the-art. The straightforward deep-learning-based OSL method is to add a threshold to the close set recognition [24]. However, unknown samples could also obtain high scores because of softmax. To address this issue, Openmax [3] is proposed to redistribute the probability scores produced by softmax and estimates the probability of an input that belongs to an unknown class. Besides, the difficulty of training unknown set arises from the lack of unknown samples. Correspondingly, some works [40, 52] propose to synthesize images of unknown classes for the training of the network. In this work, we share the same spirit with sample synthesis methods. We randomly replace some objects/stuff of known classes in the given image with synthesized unknown objects/stuff. The corresponding annotations in ground truth mask are changed to "unknown".

3. Method

3.1. Problem Formulation

We use $\mathcal{X} = \{X^s, X^u\}$, $\mathcal{A} = \{A^s, A^u\}$, $\mathcal{Y} = \{Y^s, Y^u\}$ to represent the feature space, semantic space, and label space, respectively. The superscript s and u indicate the seen and unseen classes respectively. According to how the information (*i.e.*, X^u and A^u) of unseen classes is utilized, zero-shot semantic segmentation methods can typically be divided into two different settings: inductive setting and transductive setting. Inductive setting can only utilize the information of seen classes in training, while transductive setting can use both the seen information and the unlabeled unseen information. Inductive setting is stricter and more challenging. In detail, for inductive ZSS, its training set can be denoted as $D_{train} = \{(x_i^s, a_i^s, y_i^s)_{i=1}^{N^s} | x_i^s \in X^s, a_i^s \in A^s, y_i^s \in Y^s\}$, where the subscript i indicates the ith sample and N^s is the number of training samples for seen classes. $x_i^s \in \mathbb{R}^K$ is the K-dimensional image* (visual) feature of the ith training sample. $A^s = \{a^{s,1}, \cdots, a^{s,n_{seen}}\}$ indicates the semantic representations of seen classes in the semantic space $\mathcal{A}$, and n_{seen} is the number of seen classes. $Y^s = \{y^{s,1}, \cdots, y^{s,n_{seen}}\}$ represents the label set of the

*Here specifically for segmentation, the "image" is actually a "pixel".

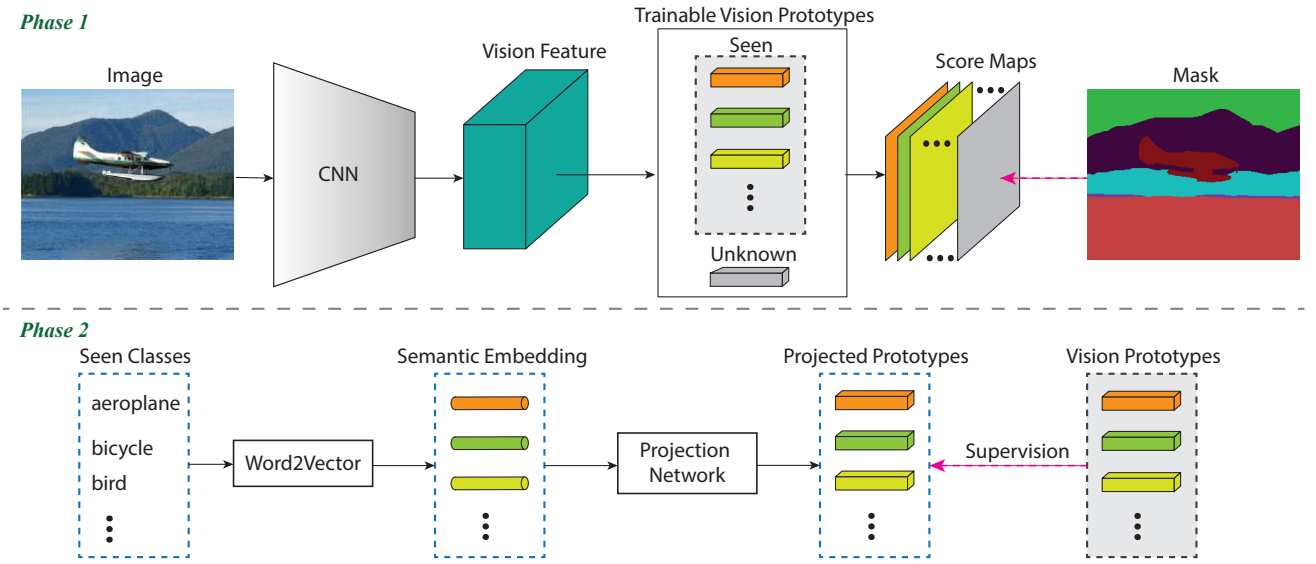

Figure 2. The overall framework for training. In training phase 1, we aim at training the segmentation network with a set of trainable prototypes for seen categories and the "unknown" category. The network conducts pixel-wise classification via calculating the distances between the vision prototypes and features of each pixel. In phase 2, a projection network is trained to bridge the semantic information and the vision prototypes obtained in phase 1.

n_{seen} seen classes in label space $\mathcal{Y}$. While for transductive ZSS, in addition to the information of seen classes, visual features X^u and semantic representations A^u of the unseen classes can be used in training without knowing their corresponding labels. The training set of transductive ZSS is denoted by $D_{train} = \{(x_i^s, a_i^s, y_i^s)_{i=1}^{N^s}, (x_j^u, a_j^u)_{j=1}^{N^u} | x_i^s \in X^s, a_i^s \in A^s, y_i^s \in Y^s, x_j^u \in X^u, a_j^u \in A^u\}$, where N^u is the number of samples for unseen classes.

The label set of unseen classes is represented by $Y^u = \{y^{u,1}, \cdots, y^{u,n_{unseen}}\}$, where n_{unseen} is the number of unseen classes. There is no overlap between the seen and unseen classes, *i.e.*, $Y^s \cap Y^u = \emptyset$. Both inductive and transductive ZSS settings target at learning a model $f_{ZSS} : \mathcal{X} \rightarrow \mathcal{Y}$ to generate pixel-level segmentation mask for each of N_t test samples. The testing samples of zero-shot learning (ZSL) only contain unseen classes, *i.e.*, $D_{test} = \{X^u, A^u, Y^u\}$, while the generalized zero-shot learning (GZSL) contains both seen and unseen classes, *i.e.*, $D_{test} = \{\mathcal{X}, \mathcal{A}, \mathcal{Y}\}$. ZSS is naturally an analogue of GZSL.

3.2. Architecture Overview

In this work, the inductive setting is adopted, where only the images, semantic information and ground truth of seen classes are accessible in training. The overall architecture for training is shown in Figure 2. The training can be divided into two stages. In stage 1, we train an open-aware segmentation network to recognize the seen classes, as well as to identify whether a pixel belongs to the seen classes, by defining an "unknown" category. To make the adding of new classes flexible, the classification of the segmentation network is carried out in a prototypical way, instead of the convolutional classifier way. We are able to obtain a set of prototypes, each for a seen class, at the end of stage 1. In the stage 2, we aim to learn a projection between the semantic and visual information. A projection network is trained to map the semantic representation of each seen class to its corresponding prototype obtained from stage 1. After training, the projection network would be able to generate prototypes for unseen classes, given their semantic embeddings.

Being indifferent to the seen and unseen classes in the inference process always leads to significant misclassifications, because the trained model is naturally biased towards the seen classes. Here, we place the remaining part of the open-set rejection module in inference process, as shown in Figure 3. If the closest prototype to a pixel's visual feature corresponds to one of the seen classes, the label will be output directly. Otherwise, if any pixel is classified as "unknown", it will be compared further with unseen prototypes and the label of the closest prototype will be selected. In this way, unseen classes does not need to compete with seen classes and the bias towards seen class is reduced.

3.3. Training Phase 1: Prototype Extraction

In training phase 1 in Figure 2, we aim to train an open-aware segmentation network and extract the prototypes as high-level visual representations. Before this, we implement pixel synthesis for open-set rejection module. We follow [40, 52] to synthesize images of unknown classes.

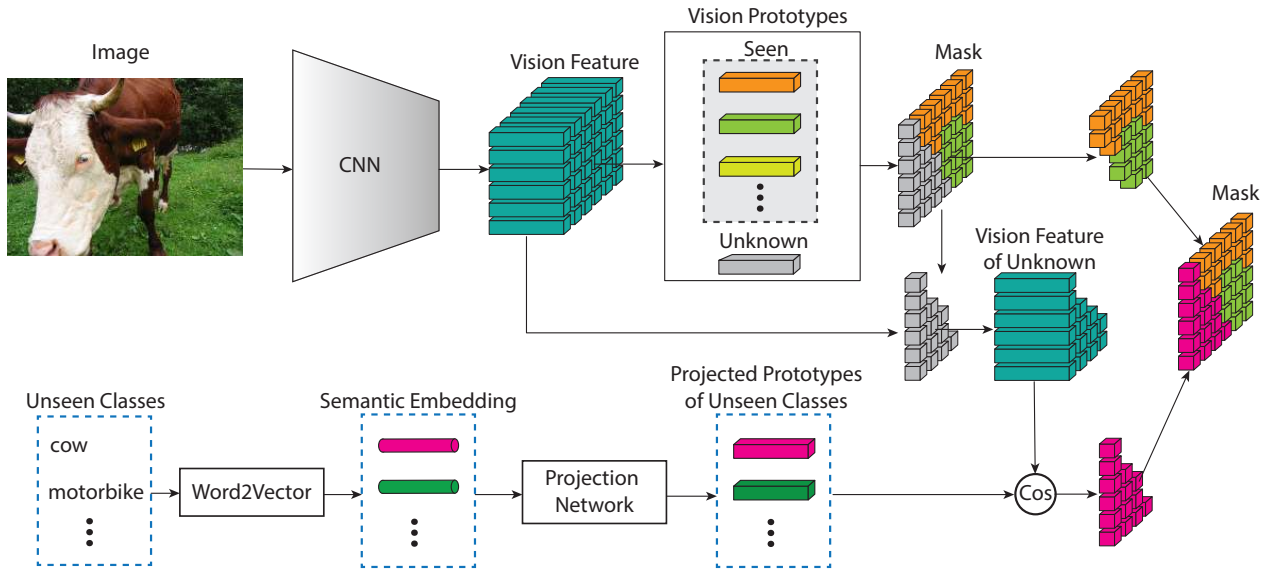

Figure 3. Framework for inference. The projection network maps the semantic embedding of each unseen class to vision prototype. We first conduct open-aware segmentation to assign each pixel to one of the n_{seen} and "unknown" categories. These pixels that are classified as "unknown" are then compared with the projected prototypes of unseen classes and classified as the the one with closest distance.

Different from [40, 52] that generate a whole image, we randomly select pixels/pieces in an image and replace them with synthesized pixel values. The synthetic process is as follows. First, we generate 5k synthetic images outside of the seen class boundaries. Second, for each training image, we randomly select a piece from its ground-truth mask, generate a map that indicates the positions to be replaced. Here, a piece refers to a sub-region that occupies 20% to 100% of the entire region with the same semantic category. We replace the RGB values of the indicated pixels in the training image with the RGB values of the same positions in the synthetic image. Then these pixels/pieces is assigned with a new category, which is collectively referred to as an "unknown" category. The processed images are used to train the segmentation network. It is worth noting here the difference between "unknown" and "unseen". The "unseen" categories are defined in the zero-shot segmentation task and are not accessible during training, but have to be predicted during inference. The "unknown" category is defined in the training phase, and it specifically refers to pixels that do not belong to any seen categories.

For each input image, we convert it into visual features through a backbone network. We adopt prototypical way to classify each pixel. A set of prototypes are randomly initialized as trainable parameters. We calculate the cosine similarity between vision features at each spatial location with the prototypes. Then we apply softmax over the distances to produce a group of probability score maps $\hat{S}$ over semantic classes. Concretely, suppose the set of prototypes $\mathcal{P} = \{p_i | i \in (1, ..., n_{seen} + 1)\}$ and $f^{x,y}$ denote feature vector of position (x, y) in the feature map F. For each p_i we have the score map

$$\hat{S}_i^{(x,y)} = \frac{exp(-\alpha \langle f^{x,y}, p_i \rangle)}{\sum_{p_i \in \mathcal{P}} exp(-\alpha \langle f^{x,y}, p_i \rangle)} \tag{1}$$

where $\langle , \rangle$ represents the computation of cosine similarity. α is an amplification factor. The predicted segmentation mask is then given by

$$\hat{M}^{(x,y)} = \arg\max_i \hat{S}_i^{(x,y)} \tag{2}$$

The supervision loss for training the prototypes are

$$\mathcal{L}_{seg} = -\frac{1}{N} \sum_{x,y} \sum_{p_i \in \mathcal{P}} \mathbb{1}(M^{(x,y)} = i) \log \hat{S}_i^{x,y} \tag{3}$$

where M is the ground truth segmentation mask and N is the total amount of spatial locations. $\mathbb{1}(*)$ is the binary label indicator that outputs 1 when $*$ is true. Optimizing the above loss will train suitable prototypes for each class, including the one for the unknown category.

3.4. Training Phase 2: Semantic-Visual Projection

In the phase 2, we aim to train a projection network that bridges the visual and semantic information. We use the trained vision prototypes of seen classes, *i.e.*, $\mathcal{P}^s = \{p_i | i \in (1, ..., n_{seen})\}$, in training phase 1 as the visual representations. To obtain semantic information, we use the word2vec [37] model which takes the names of seen classes as input and generates embeddings to be the semantic representations. Concretely, we denote the names of seen classes

as $W^s = \{w_i^s | i \in (1, ..., n_{seen})\}$) and the semantic embeddings of seen classes as $A^s = \{a_i^s | i \in (1, ..., n_{seen})\}$, we have $a_i^s = word2vec(w_i^s)$. The word2vec model is trained from a dump of the Wikipedia corpus (about 3 billion words) and produce closer embeddings for words that are closer in context. Thus, the generated semantic embeddings by word2vec is expected to have captured the semantic correlations among the classes.

We stack three linear layers to form a lightweight projection network. Under the L_2 regression loss, the semantic projection network maps the semantic embedding a_i^s of each seen class into the corresponding prototypes p_i obtained from training phase 1, as shown in Figure 2. The projection network learns how to bridge the semantic and visual information in training, and it will generate projected vision prototype for the newly added unseen class in testing. The using of vision prototypes simplifies the projection between visual and semantic information into a linear transformation between two vectors.

3.5. Inference

The inference of our approach is shown in Figure 3. Before segmentation, we first predict the prototypes for the unseen classes through the training outcome at phase 2. We denote the unseen prototypes as $\mathcal{P}^u$, where $\mathcal{P}^u = \left\{p_j^u | j \in (1, ..., n_{unseen})\right\}$. Notably, there is no limit on the number of unseen classes and it is easy to add new prototype by our projection network. Then we conduct the open-aware segmentation to assign each pixel with one of the $n_{seen} + 1$ categories, as in Eq. (1) and Eq. (2). Those pixels classified as "unknown" category are denoted as (x^u, y^u). The (x^u, y^u), *i.e.*, mask of "unknown", is used to index and split the vision features of unknown, as shown in Figure 3. We use the unseen prototype $\mathcal{P}^u$ to designate these "unknown" pixels as one of the unseen categories. The labels in $\hat{M}^{(x^u, y^u)}$ are replaced by

$$\hat{S}_j^{(x^u, y^u)} = \frac{exp(-\alpha \langle f^{x^u, y^u}, p_j^u \rangle)}{\sum_{p_j^u \in \mathcal{P}^u} exp(-\alpha \langle f^{x^u, y^u}, p_j^u \rangle)} \tag{4}$$

$$\hat{M}^{(x^u, y^u)} = \arg\max_j \hat{S}_j^{(x^u, y^u)} \tag{5}$$

Till now, the input image is classified into either seen classes or unseen classes. The unknown class functions as a medium which does not appear in the final prediction, but do participate in the intermediate process. By separating seen and unseen, the open-set rejection helps alleviate the bias problem in (generalized) zero-shot segmentation.

4. Experiments

4.1. Implementation Details

Datasets. We perform experimental evaluation on two datasets: Pascal VOC 2012 [16] and Pascal Context [39]. Pascal VOC 2012 provides segmentation annotations of 21 object classes (including background) for 1464 training and 1449 validation images. Pascal Context contains 4998 training and 5105 validation images of 60 object/stuff classes (including background), and provides dense semantic segmentation annotations for Pascal VOC 2010.

Word embeddings and zero-shot setups. In this work, we use the embeddings generated by word2vec as the semantic representations [11]. The semantic similarity between labels plays an important role in bridging the gap between seen and unseen set. As similar to ZS3 [5], we use zero-shot setups with different number of unseen classes, constructing the 2-, 4-, 6-, 8- and 10-class unseen sets. The detailed splits of Pascal VOC are: 2-cow/motobike, 4-airplane/sofa, 6-cat/tv, 8-train/bottle, 10-chair/potted plant. The detailed splits of Pascal Context are: 2-cow/motorbike, 4-sofa/cat, 6-boat/fence, 8-bird/tvmonitor, 10-keyboard/aeroplane. In different setups, the classes in unseen set increases incrementally, which means, for example, 4-unseen set contains 2-unseen set.

Evaluation metrics. Three standard semantic segmentation matrics, *i.e.*, pixel accuracy (PA), mean accuracy (MA) and mean-intersection-over-union (mIoU) are reported in our experiments. Additionally, in the generalized zero shot segmentation, the search space becomes the union of seen and unseen classes. Following [48, 5], we compute the harmonic mean of seen mIoU and unseen mIoU by

$$\text{hIoU} = \frac{2 \times \text{mIoU}_{\text{seen}} \times \text{mIoU}_{\text{unseen}}}{\text{mIoU}_{\text{seen}} + \text{mIoU}_{\text{unseen}}} \tag{6}$$

where $\text{mIoU}_{\text{seen}}$ and $\text{mIoU}_{\text{unseen}}$ represents the mean IoU of seen classes and unseen classes respectively. $\text{mIoU}_{\text{seen}}$ is commonly much higher than $\text{mIoU}_{\text{unseen}}$ and dominates the overall mIoU. Therefore, we use hIoU that can better demonstrate the overall performance of ZSS.

Backbone and Training details. We adopt ResNet-101 [23] as the backbone to build a DeepLav3+ [8] framework, and train the model by SGD optimize with polynomial learning rate decay. The base learning rate is 7e-3, the weight decay is 5e-4 and the momentum is 0.9. We use the pre-trained model provided by ZS3Net [5], which solely uses seen categories, so to guarantee no information leakage.

4.2. Ablation Studies

1) Trainable Prototypes *vs*. Convolutional Classifier

We test these two different ways, the prototypical way and the convolutional classifier, on supervised network and get the comparable segmentation results of 76.9% *vs*. 76.8% in terms of mIoU. Compared with convolutional classifier, prototypical way is more flexible in adding new category during testing, and simplifies the mapping between semantic and visual information to the projection between semantic embeddings and prototype vectors.

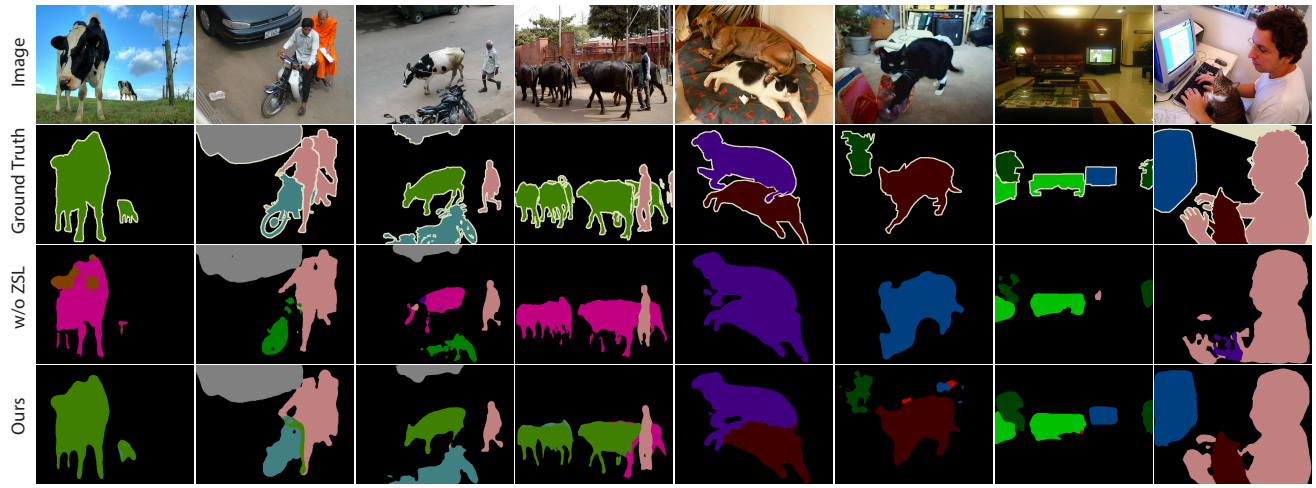

Figure 4. Qualitative results on Pascal VOC. From the first row to the last row are image, ground truth, prediction without ZSL, and our prediction, respectively. Without zero-shot learning, the model incorrectly classify the unseen classes to some seen classes, *e.g.*, recognize the cow as horse in the first column. Our zero-shot segmentation approach can assign correct labels to the unseen classes and generate acceptable masks for these objects.

Table 1. Ablation study of open-set rejection.

Methods	cow/motorbike			cat/tv		
	seen	unseen	hIoU	seen	unseen	hIoU
w/o open-set rejection	66.1	2.8	5.4	69.8	3.2	6.1
w/ open-set rejection	73.8	51.3	60.5	75.4	53.0	62.2
grount-truth rejection	76.7	64.0	69.8	75.6	65.3	70.1

Table 2. Ablation study of projection.

Methods	aeroplane/sofa			bird/boat		
	seen	unseen	hIoU	seen	unseen	hIoU
embedding	76.1	56.7	65.0	74.9	53.9	62.7
projection	76.0	65.2	70.2	75.3	60.2	66.9

2) Open-Set Rejection

In ZSS, the open-set rejection module aims to alleviate the bias problem. To demonstrate the efficiency of the open-set rejection, we design and conduct three experimental settings: (1) Without open-set rejection, we generate the unseen prototypes using the projection network, and directly predict seen and unseen classes simultaneously. We expect the results to be poor, because training images contain only annotated pixels of seen classes, at the testing stage, predictions will be biased significantly to seen classes. (2) Our open-set rejection way, in ZSS and inductive setting. (3) We use ground truth to filter pixels that belong to the unseen classes, and make predictions independently of seen classes. In other words, we only distinguish which one of the unseen classes the pixel belongs to, without distinguishing whether it comes from the seen category. This setting reduces the challenges of ZSS, but it is impractical in actual use and will not be regarded as a meaningful setting of ZSS. However, it imposes an upper limit on how much the overall performance can be improved by solving the bias problem, and estimates the extent to which our open-set rejection module alleviates the bias problem.

The results are shown in Table 1. As observed, when the unseen classes are cow and motorbike that have similar seen classes, *e.g.*, cow-horse and motorbike-bicycle, setting (1) results in poor mIoU as 2.8% on unseen classes, indicating that the network has few classification capability on unseen partitions. Network trained under setting (2) has an mIoU of 73.8% on seen classes and 51.3% on unseen classes and outperforms setting (1) by a very large margin of 55.1% hIoU, which demonstrates the effectiveness of our open set rejection. The setting (3), which is not bothered by the bias problem, has an mIoU of 76.7% on seen classes and 64.0% on unseen classes. This is the maximum gain that the open-set rejection can bring about.

3) Projection Network

The projection network is proposed and aims to transfer the knowledge from seen classes to unseen classes. To quantify the effectiveness of the proposed projection network, we compare the following two settings: (1) When the word embeddings are used directly as the prototypes. (2) Our projection network way. The results are shown in Table 2. For example, when the useen classes are aeroplane and sofa (both do not have similar seen classes), setting (1) has a mIoU of 76.1% on seen classes and 56.7% on unseen classes. The setting (2) has a mIoU of 76.0% on seen classes and 65.2% on unseen classes. The projection way has higher unseen accuracies, which demonstrates its effectiveness. The reason why setting (2) outperforms setting (1)

Table 3. Results on Pascal VOC 2012.

setting	model	Seen	Unseen	Overall	
		mIoU	mIoU	mIoU	hIoU
0	Supervised	-	-	76.9	-
2	ZS3 [5]	72.0	35.4	68.5	47.5
	CSRL [33]	73.4	45.7	70.7	56.3
	Ours	**73.7**	**51.3**	**71.6**	**60.5**
4	ZS3	66.4	23.2	58.2	34.4
	CSRL	69.8	31.7	62.5	43.6
	CaGNet [20]	69.5	40.2	63.2	50.9
	Ours	**75.0**	**44.1**	**69.1**	**55.5**
6	ZS3	47.3	24.2	40.7	32.0
	CSRL	66.2	29.4	55.6	40.7
	Ours	**74.3**	**41.4**	**64.9**	**53.2**
8	ZS3	29.2	22.9	26.8	25.7
	CSRL	62.4	26.9	48.8	37.6
	Ours	**73.8**	**37.6**	**60.0**	**49.8**
10	ZS3	33.9	18.1	26.3	23.6
	CSRL	59.2	21.0	50.0	31.0
	Ours	**72.1**	**33.9**	**53.9**	**46.1**

Table 4. Results on Pascal Context.

setting	model	Seen	Unseen	Overall	
		mIoU	mIoU	mIoU	hIoU
0	Supervised	-	-	42.7	-
2	ZS3+GC [5]	41.5	30.0	41.3	34.8
	CSRL [33]	41.9	27.8	41.4	33.4
	Ours	**41.9**	**51.8**	**42.2**	**46.3**
4	ZS3+GC	39.5	29.1	38.6	33.5
	CSRL	39.8	23.9	38.7	29.9
	Ours	**41.1**	**43.1**	**41.2**	**42.1**
6	ZS3+GC	34.8	21.6	33.5	26.7
	CSRL	35.5	22.0	34.1	27.2
	Ours	**40.9**	**36.4**	**40.5**	**38.5**
8	ZS3+GC	22.8	16.8	22.0	19.3
	CSRL	31.7	18.1	29.9	23.0
	Ours	**40.2**	**27.3**	**38.5**	**32.5**
10	ZS3+GC	24.0	14.1	22.3	17.8
	CSRL	29.4	14.6	27.0	19.5
	CaGNet [20]	24.8	18.5	23.2	21.2
	Ours	**39.8**	**21.3**	**36.7**	**27.7**

is: Data samples of some of the seen and unseen classes are disjoint and unrelated. Directly using word embeddings for classification may prove useful for those unseen classes that has a very close counterpart in seen classes. However, for disjoint unseen classes, a large domain gap exists, and learning the projection function using seen samples is of great help to mitigate the domain gap.

4.3. Results on Benchmarks

Generalized ZSS is a realistic segmentation setting. Instead of only evaluating on the unseen set, we jointly evaluate all classes and report results on seen, unseen, and overall classes, *i.e.*, a pixel can be assigned to one of the seen or one of the unseen classes. The prediction is supposed to be biased towards seen classes because the training images contain only labeled pixels of seen classes. Hence, this is a particularly challenging task. We report in Table 3 and 4 the performance metrics on Pascal VOC 2012 and Pascal Context datasets, according to the three metrics. We focus on unseen class that have a semantically similar seen class, *i.e.*, cow (horse/sheep), motorbike (bicycle), cat (dog), train (bus), chair (dining table). The high accuracies prove that our method effectively alleviate the bias problem and domain-shift problem.

Qualitative Results Figure 4 shows the qualitative results. The sub-images displayed from top to bottom are original images, ground truth, segmentation results without zero-shot learning, and our ZSS results. Most of the shown cases contain more than three classes and are challenging due to the very similar classes in unseen and seen divisions (cow - horse, motorbike - bicycle, cat - dog), the entanglement of different objects, and various scale of the objects. For example, in the 3^{rd} column, which includes the two unseen classes of motorbike and cow, w/o ZSL result predicts the cow as a horse and the motorbike as a bicycle, while our method successfully predicts the unseen classes. Especially in the 5^{th} column, the cat and the dog with similar appearance appear in the same picture, and our method can achieve segmentation well. After open-set rejection, the segmentation present some minor errors between the completely different unseen classes (*e.g.*, cow and motorbike), which looks like irrational, but is actually explainable because we could hardly achieve a 100% segmentation within the rejected group.

5. Conclusion

In this work, we address the challenging zero-shot semantic segmentation (ZSS), where a model is required to conduct pixel-level classification of categories that have been seen or not seen during training. We clarify the inductive/transductive settings of ZSS and adopt the inductive setting. We propose an approach with prototypical matching and open-set rejection to enhance the zero-shot performance. A set of trainable prototypes are employed to extract the visual representations and perform classification. A projection network is trained to map the semantic embeddings to these prototypes, based on seen instances, and generate prototypes for unseen classes. To address the bias problem, an open-set rejection (OSR) module is proposed to identify the pixels that do not belong to seen classes. The OSR help to reduce the misclassification of unseen objects into seen classes. Then the pixels rejected by OSR is classified by projected prototypes of unseen classes. We test the proposed approach on two segmentation datasets and achieve impressively state-of-the-art performance on generalized zero-shot segmentation.

References

[1] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2927–2936, 2015. 3

[2] Abhijit Bendale and Terrance Boult. Towards open world recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1893–1902, 2015. 3

[3] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1563–1572, 2016. 3

[4] Supritam Bhattacharjee, Devraj Mandal, and Soma Biswas. Autoencoder based novelty detection for generalized zero shot learning. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 3646–3650. IEEE, 2019. 3

[5] Maxime Bucher, VU Tuan-Hung, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *Advances in Neural Information Processing Systems*, pages 468–479, 2019. 1, 2, 6, 8

[6] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Classifier and exemplar synthesis for zero-shot learning. *International Journal of Computer Vision*, 128(1):166–201, 2020. 3

[7] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *European Conference on Computer Vision*, pages 52–68. Springer, 2016. 3

[8] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 1, 6

[9] Debasmit Das and CS George Lee. Zero-shot image recognition using relational matching, adaptation and calibration. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019. 3

[10] Berkan Demirel, Ramazan Gokberk Cinbis, and Nazli Ikizler-Cinbis. Attributes2classname: A discriminative model for attribute-based unsupervised zero-shot learning. In *Proceedings of the IEEE international conference on computer vision*, pages 1232–1241, 2017. 3

[11] Henghui Ding, Scott Cohen, Brian Price, and Xudong Jiang. Phraseclick: toward achieving flexible interactive segmentation by phrase and click. In *European Conference on Computer Vision*, pages 417–435. Springer, 2020. 6

[12] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-aware feature propagation for scene segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6819–6829, 2019. 1

[13] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2393–2402, 2018. 1

[14] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic correlation promoted shape-variant context for segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8885–8894, 2019. 1

[15] Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568*, 2014. 3

[16] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015. 6

[17] Rafael Felix, Ben Harwood, Michele Sasdelli, and Gustavo Carneiro. Generalised zero-shot learning with domain classification in a joint semantic and visual space. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE, 2019. 3

[18] Yanwei Fu, Tao Xiang, Yu-Gang Jiang, Xiangyang Xue, Leonid Sigal, and Shaogang Gong. Recent advances in zero-shot recognition: Toward data-efficient understanding of visual content. *IEEE Signal Processing Magazine*, 35(1):112–125, 2018. 1

[19] Chuang Gan, Ming Lin, Yi Yang, Yueting Zhuang, and Alexander G Hauptmann. Exploring semantic inter-class relationships (sir) for zero-shot action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015. 3

[20] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. Context-aware feature generation for zero-shot semantic segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1921–1929, 2020. 1, 2, 8

[21] Zhangxuan Gu, Siyuan Zhou, Li Niu, Zihan Zhao, and Liqing Zhang. From pixel to patch: Synthesize context-aware features for zero-shot semantic segmentation. *arXiv preprint arXiv:2009.12232*, 2020. 1, 3

[22] Yuchen Guo, Guiguang Ding, Jungong Han, Xiaohan Ding, Sicheng Zhao, Zheng Wang, Chenggang Yan, and Qionghai Dai. Dual-view ranking with hardness assessment for zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8360–8367, 2019. 3

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[24] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 3

[25] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[26] Ping Hu, Stan Sclaroff, and Kate Saenko. Uncertainty-aware learning for zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 2

[27] Dat Huynh and Ehsan Elhamifar. Fine-grained generalized zero-shot learning via dense attribute-based attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4483–4493, 2020. 3

[28] Lalit P Jain, Walter J Scheirer, and Terrance E Boult. Multi-class open set recognition using probability of inclusion. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 393–409. Springer, 2014. 3

[29] Mihir Jain, Jan C Van Gemert, Thomas Mensink, and Cees GM Snoek. Objects2action: Classifying and localizing actions without any video example. In *Proceedings of the IEEE international conference on computer vision*, pages 4588–4596, 2015. 3

[30] Pedro R Mendes Júnior, Roberto M de Souza, Rafael de O Werneck, Bernardo V Stein, Daniel V Pazinato, Waldir R de Almeida, Otávio AB Penatti, Ricardo da S Torres, and Anderson Rocha. Nearest neighbors distance ratio open-set classifier. *Machine Learning*, 106(3):359–386, 2017. 3

[31] Naoki Kato, Toshihiko Yamasaki, and Kiyoharu Aizawa. Zero-shot semantic segmentation via variational mapping. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1, 2

[32] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2009. 1

[33] Peike Li, Yunchao Wei, and Yi Yang. Consistent structural relation learning for zero-shot segmentation. *Advances in Neural Information Processing Systems*, 33, 2020. 2, 8

[34] Yan Li, Zhen Jia, Junge Zhang, Kaiqi Huang, and Tieniu Tan. Deep semantic structural constraints for zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 3

[35] Shichen Liu, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Generalized zero-shot learning with deep calibration network. In *Advances in Neural Information Processing Systems*, pages 2005–2015, 2018. 3

[36] Fengmao Lv, Haiyang Liu, Yichen Wang, Jiayi Zhao, and Guowu Yang. Learning unbiased zero-shot semantic segmentation networks via transductive transfer. *IEEE Signal Processing Letters*, 27:1640–1644, 2020. 1, 2

[37] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 5

[38] Shaobo Min, Hantao Yao, Hongtao Xie, Chaoqun Wang, Zheng-Jun Zha, and Yongdong Zhang. Domain-aware visual bias eliminating for generalized zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12664–12673, 2020. 3

[39] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898, 2014. 6

[40] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 613–628, 2018. 3, 4, 5

[41] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International conference on machine learning*, pages 2152–2161. PMLR, 2015. 1

[42] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1757–1772, 2012. 1, 3

[43] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1757–1772, 2012. 3

[44] Walter J Scheirer, Lalit P Jain, and Terrance E Boult. Probability models for open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2317–2324, 2014. 3

[45] Richard Socher, Milind Ganjoo, Hamsa Sridhar, Osbert Bastani, Christopher D Manning, and Andrew Y Ng. Zero-shot learning through cross-modal transfer. *arXiv preprint arXiv:1301.3666*, 2013. 3

[46] Xinsheng Wang, Shanmin Pang, and Jihua Zhu. Domain segmentation and adjustment for generalized zero-shot learning. *arXiv preprint arXiv:2002.00226*, 2020. 3

[47] Yongqin Xian, Subhabrata Choudhury, Yang He, Bernt Schiele, and Zeynep Akata. Semantic projection network for zero-and few-label semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8256–8265, 2019. 1, 2

[48] 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*, 41(9):2251–2265, 2018. 6

[49] Felix X Yu, Liangliang Cao, Rogerio S Feris, John R Smith, and Shih-Fu Chang. Designing category-level attributes for discriminative visual recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 771–778, 2013. 3

[50] He Zhang and Vishal M Patel. Sparse representation-based open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1690–1696, 2016. 3

[51] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE international conference on computer vision*, pages 4166–4174, 2015. 3

[52] Sergey Demyanov Zongyuan Ge and Rahil Garnavi. Generative openmax for multi-class open set classification. In Gabriel Brostow Tae-Kyun Kim, Stefanos Zafeiriou and Krystian Mikolajczyk, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 42.1–42.12. BMVA Press, September 2017. 3, 4, 5