SIGN: Spatial-information Incorporated Generative Network for Generalized Zero-shot Semantic Segmentation

Jiaxin Cheng, Soumyaroop Nandi, Prem Natarajan, Wael Abd-Almageed
USC Information Sciences Institute, Marina del Rey, CA, USA
{chengjia,soumyarn,pnataraj,wamageed}@isi.edu

Abstract

Unlike conventional zero-shot classification, zero-shot semantic segmentation predicts a class label at the pixel level instead of the image level. When solving zero-shot semantic segmentation problems, the need for pixel-level prediction with surrounding context motivates us to incorporate spatial information using positional encoding. We improve standard positional encoding by introducing the concept of Relative Positional Encoding, which integrates spatial information at the feature level and can handle arbitrary image sizes. Furthermore, while self-training is widely used in zero-shot semantic segmentation to generate pseudo-labels, we propose a new knowledge-distillation-inspired self-training strategy, namely Annealed Self-Training, which can automatically assign different importance to pseudo-labels to improve performance. We systematically study the proposed Relative Positional Encoding and Annealed Self-Training in a comprehensive experimental evaluation, and our empirical results confirm the effectiveness of our method on three benchmark datasets.

1. Introduction

Zero-shot learning (ZSL) solves the task of learning in the absence of training data of that task (*e.g*., recognizing unseen classes). It has been widely adopted in classic computer vision problems, such as classification [1, 15, 2, 46, 48, 52, 58, 41, 54, 7, 38, 14, 34, 39, 19, 25, 51] and object detection [42, 3, 16, 11, 43, 4]. The key challenge in ZSL based tasks is to make the underlying model capable of recognizing classes that had not been seen during training. Earlier work focused on learning a joint embedding between seen and unseen classes [1, 15, 2, 46, 48, 52, 58, 41]. In recent works, knowledge and generative-based methods have gained more prominence. Knowledge-based methods [39, 19, 25, 51] use structured knowledge learned in another domain (*e.g*., natural language [36], knowledge graph[32], *etc*.) as constraints [19, 25] to transfer the learned information to unseen categories. Generative methods use attributes [14, 7], natural language [44] or word embeddings [14, 9] as priors to generate synthetic features for unseen categories.

In this work, we investigate the zero-shot semantic segmentation problem, which is less studied than other zero-shot computer vision problems [53, 5, 26, 18]. Generative methods have been widely adopted in zero-shot semantic segmentation problem. Bucher *et al*. [5] proposed ZS3Net, in which they leverage Generative Moment Matching Networks [31] to generate synthetic features for unseen categories by using word2vec [35] embeddings and random noise as prior (Fig. 1(a)). The use of random noise prevents the model from collapsing, which occurs due to lack of feature variety [59, 17]. Li *et al*. [30] extended ZS3Net [5] by adding a structural relation loss which constrains the generated synthetic features to have a similar structure as the semantic word embedding space. Gu *et al*. [18] suggested using a context-aware normal distributed prior when generating synthetic features instead of random noise (Fig. 1(b)).

We propose to incorporate spatial information to improve the performance of the zero-shot semantic segmentation problem. Our motivation arises from the assumption that knowing a pixel's location may help semantic segmentation because it is a 2D prediction task. Incorporating spatial information in computer vision problems has recently attracted the attention of the community. In image classification, [12] slices the input image into nine patches and adds a positional vector for each patch to indicate the patches' location. Other methods leverage [56] the relative position of objects through a space-aware knowledge graph for object detection. Zhang *et al*. [57] counted the co-occurrence of features to learn spatial invariant representation for semantic segmentation. However, to the best of our knowledge, spatial information has not been widely studied in previous zero-shot learning research. In this work, we propose to exploit spatial information by using Positional Encoding [50], as shown in Fig. 1(c). Positional Encoding generates a positional vector that indicates the position of a pixel in the image. Previous work [12] divided the input images into fixed number of patches and appended positional embeddings on the images. However, in our case, dividing the input image into small patches is incompatible because the semantic segmentation problem requires the entire image as input. We mitigate this shortcoming by incorporating spatial information into the image features and propose Relative Positional Encoding to handle varying input image sizes.

In the zero-shot learning problem, unlabeled samples, including unseen classes, are sometimes available. Trained models can annotate unlabeled samples to obtain additional training data [60] and fine-tune the models with pseudo-annotated data. Such a training strategy is called self-training. In zero-shot semantic segmentation, self-training annotates pixel-level pseudo labels [5, 18]. To reduce the number of unreliable pseudo labels, [5] ranks the confidence score (*i.e.* the probability of classes after *softmax* function) of pseudo labels and uses only the most confident 75% pseudo labels. [18] eliminates pseudo labels if the confidence score is below a certain threshold.

This work proposes a knowledge distillation-inspired [22] self-training strategy, namely *Annealed Self-Training* (AST), to generate better pseudo-annotations for self-training. Knowledge distillation is widely used in the form of teacher-student learning [22, 37], where the student network is trained on the soft labels from the teacher network in addition to (one-hot) hard labels since soft labels have more information due to high entropy [22]. In the zero-shot semantic segmentation problem, previous methods [5, 18] set a threshold to eliminate pseudo labels from the low confidence scores, while assigning the same loss weights to the remaining pseudo labels. This is similar to using hard labels in teacher-student learning. However, it is hard to ensure that the threshold can generalize for each sample, and again low confidence pseudo labels may also contain some useful information. To avoid the shortcomings of setting a threshold and assigning the same loss weights to pseudo labels, in AST, we use pseudo-annotations of all unlabeled pixels while re-weighing their importance according to their confidence score. We leverage the annealed *softmax* [22] function to normalize the pseudo labels' weights and control their relative importance by adjusting the *annealed temperature* in the *softmax* function.

We make the following contributions in this paper. Firstly, we introduce the Spatial Information Module to incorporate spatial information in semantic segmentation using a novel Relative Positional Encoding (RPE) scheme. Compared to previous work [12], RPE does not need patch-sliced input and can handle varying image sizes. Secondly, we propose a knowledge distillation-inspired self-training strategy, namely Annealed Self-Training (AST). AST generates pseudo-annotations for unlabeled samples and adjusts their importance during self-training with a tunable *annealed temperature*. Finally, we evaluate the performance

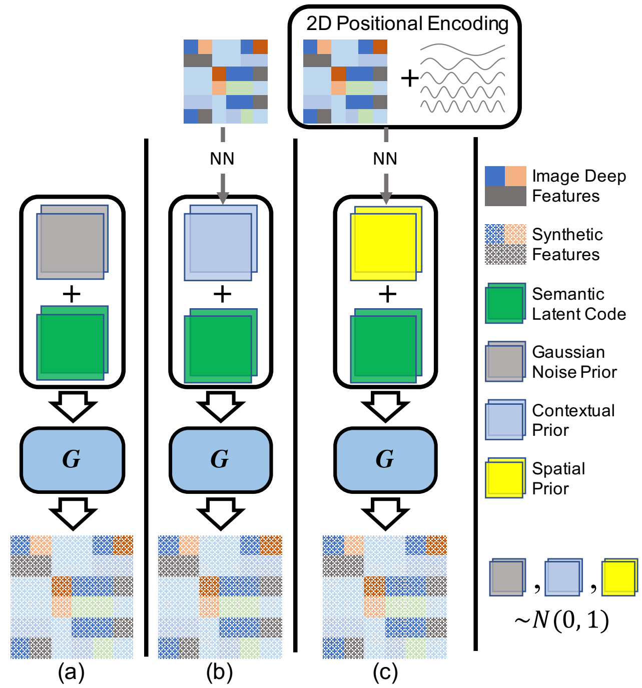

Figure 1. Latent code used in generating synthetic features consists of two parts: i) semantic word embeddings and ii) a normal distributed prior. The normal distributed prior can be: (a) Gaussian noise [5]; (b) Context-aware prior [18]; (c) Our context-aware and space-aware priors.

on three benchmark datasets and conduct extensive ablation experiments to demonstrate the effectiveness of our method.

2. Related Work

Zero-shot Learning Without loss of generality, approaches to zero-shot classification can be categorized into three families — joint embedding, generative and knowledge-based methods. Earlier works focused on linear embedding [1, 15, 2, 46], non-linear embedding [48, 52], and hybrid embedding [58, 41] methods. In the embedding-based methods, the basic idea is to learn encodings for images and attributes (*e.g.*, description) and maximize a linear/non-linear score between matched pairs. Generative [54, 7, 38, 14, 34] and knowledge-based [39, 19, 25, 51] approaches have recently become more popular. Generative method use attributes to create synthetic images with a generative model (*e.g.*, generative adversarial network [17] or a conditional variational autoencoder [49]) and then train a classifier based on seen and synthetic unseen categories. Knowledge-based method often use structured knowledge as constraints [19, 25, 51] of relationships between classes and employ graph networks, *e.g.*, Graph Convolutional Network [29], to generalize learned information from seen categories to unseen ones.

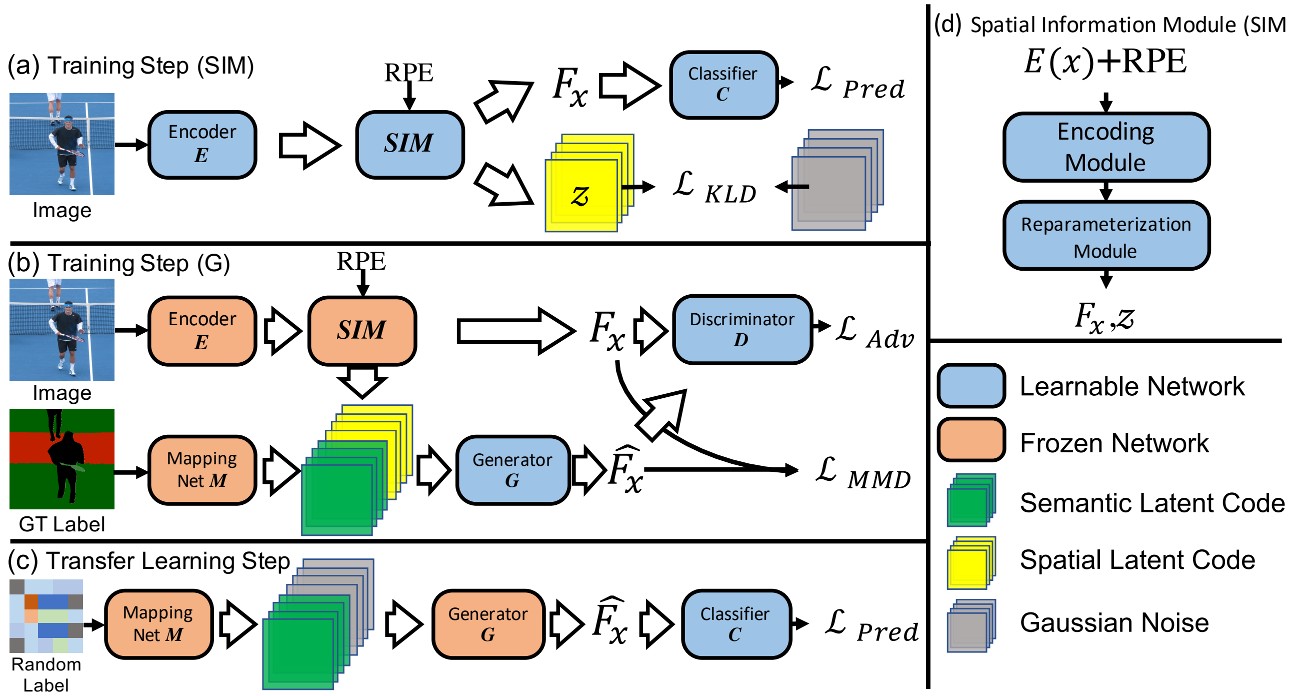

Figure 2. Our model consists of five learnable networks (***E***, ***G***, ***C***, ***D***, ***SIM***) and one unlearnable network (***M***). ***E*** is the feature encoder CNN using Deeplab-v2 architecture. Generator ***G*** generate synthetic features for unseen classes which can deceive discriminator ***D***. Classifier ***C*** is trained on seen categories during Training Step and synthetic unseen categories during Transfer Learning Step. ***SIM*** encodes spatial information into deep features through Relative Positional Encoding. Mapping network ***M*** converts ground truth into semantic latent code.

Zero-shot Semantic Segmentation Bucher *et al.* [5] use Generative Moment Matching Networks [31] to create synthetic features for unseen categories and train a classifier based on the union of the features of seen categories and synthetic features. Xian *et al.* [53] replace the last layer of a classifier with embeddings from word2vec [35] and introduce a calibration mechanism to adjust the class probability on seen categories due to imbalanced confidence on seen and unseen classes. Kato *et al.* [26] used a variational mapping for binary zero-shot semantic segmentation. They create a two-branched – conditioning and segmentation – network. The conditioning branch takes embeddings of unseen classes and maps them to classification layers in the segmentation branch for zero-shot semantic segmentation. Gu *et al.* [18] and Li *et al.* [30] also adopt the idea of generating synthetic features. Gu *et al.* [18] used a context-aware prior to generate features of unseen classes. Li *et al.* [30] added constraints for the generation of unseen class features by exploiting structural relationships between seen and unseen categories.

3. Proposed Framework

3.1. Problem Formulation

In zero-shot learning problems, class labels consist of two parts: seen categories $\mathcal{C}^S$ and unseen categories $\mathcal{C}^U$. Meanwhile, in zero-shot semantic segmentation problems, the training set $\mathcal{D}^S$ is composed of images and labels of seen categories to which image pixels belong. In other words, $\mathcal{D}^S = \{(x, y) | \; \forall i \; y_i \in \mathcal{C}^S\}$, where x is an image and y is its corresponding ground-truth label, y_i is the ground-truth label for pixel i. Other images that include pixels of unseen categories are denoted by $\mathcal{D}^U = \{(x, y) | \; \exists i \; y_i \in \mathcal{C}^U\}$ and are only encountered at the inference time.

3.2. Spatial-information Incorporated Generative Network

Fig. 2 illustrates the proposed Spatial-information Incorporated Generative Network (SIGN). SIGN is composed of one unlearnable mapping network ***M*** and five learnable networks — feature encoder (***E***), generator (***G***), classifier (***C***), discriminator (***D***), and Spatial Information Module (***SIM***). During training, an input image first goes through the feature encoder ***E***. Then, ***SIM*** conducts RPE on image features and produces F_x for classifier ***C***, and a stochastic vector z for generator ***G***. ***G*** synthesizes features from semantic word embeddings and z with a target of image features. To generalize the model on unseen categories, a random label including unseen classes is passed to ***M*** and ***G***, and the synthesized features is used to train ***C***. At inference time, the test image only passes ***E***, ***SIM*** and ***C***.

The model is optimized in three stages — (1) *Training Step (SIM)* updates the main feature encoder ***E***, the Spatial Information Module ***SIM***, and the classifier ***C*** in a standard semantic segmentation fashion. (2) *Training Step (G)* trains a generator $\boldsymbol{G}$ that produces synthetic features with a target of real image features. (3) *Transfer Learning Step* uses synthesized features to fine-tune classifier $\boldsymbol{C}$, which enables $\boldsymbol{C}$ to recognize unseen categories.

The Training Step (SIM) objective is to fine-tune the upstream feature encoder $\boldsymbol{E}$ and train $\boldsymbol{SIM}$. $\boldsymbol{E}$ is a semantic segmentation backbone network, which extracts image features. The choice of the backbone is architecture-agnostic, and any CNN-based network can be used (*e.g.*, DeepLab [8], UNet [47], FCN [33]). The Spatial Information Module ($\boldsymbol{SIM}$) takes image feature inputs $\boldsymbol{E}(x)$, and (1) incorporates spatial information into image features through positional encoding and (2) produces a space-aware stochastic latent representation z, as shown in Eq. (1), where $\oplus$ denotes concatenation operation and PE stands for positional encoding vector.

$$F_x,\ z = \boldsymbol{SIM}(\boldsymbol{E}(x) \oplus PE) \tag{1}$$

We use KL divergence to force z to converge to a normal distribution[18] to ensure stochasticity, as shown in Eq. (2).

$$\mathcal{L}_{KLD} = \mathrm{KL}(z \| \mathcal{N}(0,1)) \tag{2}$$

We use the standard categorical cross-entropy loss to train classifier $\boldsymbol{C}$, as shown in Eq. (3)

$$\mathcal{L}_{pred}^{train}(p, y) = -\sum_c y_c \log(p_c) \tag{3}$$

where $p = \boldsymbol{C}(F_x)$ is the categorical probabilities of features, p_c is the probability of class c, and y is the ground truth label. In Training Step (SIM), the classifier $\boldsymbol{C}$ is trained only on real features (*i.e.* $c \in C^S$), and the total optimization target is the weighted sum of prediction loss and KL loss for z, where α is a hyperparameter to balance losses, as shown in Eq. (4).

$$\boldsymbol{E}^*,\ \boldsymbol{SIM}^*,\ \boldsymbol{C}^* = \min_{\boldsymbol{E}, \boldsymbol{SIM}, \boldsymbol{C}} \mathcal{L}_{pred}^{train} + \alpha \mathcal{L}_{KLD} \tag{4}$$

Training Step (G) attempts to train a generator $\boldsymbol{G}$, with a fixed encoder $\boldsymbol{E}$ and $\boldsymbol{SIM}$. The generator is needed to synthesize image features of unseen categories so that the classifier $\boldsymbol{C}$ can recognize unseen categories after being trained on synthetic features. $\boldsymbol{G}$ generates synthetic features from a latent code. The latent code consists of two parts: (1) semantic word embedding e and (2) normally distributed prior z. The stochasticity of z prevents the generative model from collapsing, as discussed in [59, 17]. The mapping network $\boldsymbol{M}$ maps ground truth annotations to semantic word embeddings, $e = \boldsymbol{M}(y)$. Its weights are initialized with word2vec [35] and fasttext [24]. The generator $\boldsymbol{G}$ produces a synthetic feature $\hat{F}_x = \boldsymbol{G}(e \oplus z)$.

The synthesized features $\hat{F}_x$ have to be close to real features of seen categories. We follow previous work [5, 18] and use Maximum Mean Discrepancy (MMD) loss [31] to reduce the distribution distance between real and synthetic features. Total loss $\mathcal{L}_{MMD}$ is the summation of MMD loss on seen classes $\mathcal{L}_{MMD}(c)$, as shown in Eq. (5).

$$\mathcal{L}_{MMD} = \sum_c \mathcal{L}_{MMD}(c) \quad ; \quad c \in C^S \tag{5}$$

where,

$$\begin{aligned}\mathcal{L}_{MMD}(c) = &\sum_{f, f' \in F_{x,c}} k(f, f') + \sum_{\hat{f}, \hat{f}' \in \hat{F}_{x,c}} k(\hat{f}, \hat{f}') \\ &- 2 \sum_{f \in F_{x,c}} \sum_{\hat{f} \in \hat{F}_{x,c}} k(f, \hat{f})\end{aligned} \tag{6}$$

where $F_{x,c}$ and $\hat{F}_{x,c}$ are real and synthetic features for class c in sample x's feature, respectively. We choose Gaussian kernel function $k(f, f') = \exp(-\frac{1}{2}\|f - f'\|^2)$ as suggested in [5]. In order to make the synthesized image features realistic, we add a discriminator $\boldsymbol{D}$ and training $\boldsymbol{G}$ to deceive $\boldsymbol{D}$ by optimizing an adversarial loss, as shown in Eq. (7) [17].

$$\mathcal{L}_{adv} = \mathbb{E}_{f \in F_x}[\log(\boldsymbol{D}(f))] + \mathbb{E}_{\hat{f} \in \hat{F}_x}[\log(1 - \boldsymbol{D}(\hat{f}))] \tag{7}$$

The total loss for Training Step (G) is composed of MMD loss and adversarial loss, and hyperparameter β controls the trade-off between two losses, as shown in Eq. (8).

$$\boldsymbol{G}^*,\ \boldsymbol{D}^* = \min_{\boldsymbol{G}} \max_{\boldsymbol{D}} \mathcal{L}_{adv} + \beta \mathcal{L}_{MMD} \tag{8}$$

Since the trainable networks and the losses do not overlap in Eqs. (4) and (8), we jointly optimize them for efficiency. Finally, to synthesize features for unseen categories, during the Transfer Learning Step, a pseudo-ground-truth $\hat{y}$ (*i.e.*, a pseudo label including unseen categories) is fed into $\boldsymbol{M}$ and $\boldsymbol{G}$. The synthetic features are used to train the classifier $\boldsymbol{C}$ so that $\boldsymbol{C}$ can recognize the unseen categories. The prediction loss of the Transfer Learning Step is shown in Eq. (9).

$$\mathcal{L}_{pred}^{trans}(p, \hat{y}) = -\sum_c \widehat{y_c} \log(p_c) \tag{9}$$

The classifier $\boldsymbol{C}$ is optimized on synthetic features as well as real features to avoid performance drop on seen categories, as shown in Eq. (10).

$$\boldsymbol{C}^* = \min_{\boldsymbol{C}} \mathcal{L}_{pred}^{train} + \mathcal{L}_{pred}^{trans} \tag{10}$$

3.3. Relative Positional Encoding

As illustrated in Fig. 2(d), the Spatial Information Module consists of the encoding module and the reparameterization module. The encoding module uses a residual structure [21] and incorporates spatial information into image features using *positional encoding* [50]. The reparameterization module [28] takes the output of the encoding module and generates a stochastic latent code. In Section 4.4, we discuss and compare different architectures for SIM.

RPE uses sine and cosine functions to incorporate pixel positions into a feature map [50]. To handle 2D positional encoding, we use a 600-dimensional vector, in which the first 300 dimensions are used for horizontal location encoding and the last 300 dimensions are used for vertical location encoding. Eqs. (11) and (12) show horizontal positional encoding and Eqs. (13) and (14) show vertical positional encoding. ${}^{i}pos^*_u$ and ${}^{i}pos^*_v$ represent the relative horizontal and vertical positions of pixel i, respectively, and d denotes the dimension. The overall dimensionality of positional encoding in each direction is $d_{model} = 300$.

$$PE({}^{i}pos^*_u, 2d) = \sin({}^{i}pos^*_u / 10000^{2d/d_{model}}) \quad (11)$$

$$PE({}^{i}pos^*_u, 2d+1) = \cos({}^{i}pos^*_u / 10000^{2d/d_{model}}) \quad (12)$$

$$PE({}^{i}pos^*_v, 2d) = \sin({}^{i}pos^*_v / 10000^{2d/d_{model}}) \quad (13)$$

$$PE({}^{i}pos^*_v, 2d+1) = \cos({}^{i}pos^*_v / 10000^{2d/d_{model}}) \quad (14)$$

In order to handle arbitrary image sizes, we do not use the absolute position pos of a given pixel in the feature map. Rather, we use the relative position pos^*, as shown in Eq. (15) and Eq. (16).

$${}^{i}pos^*_u = c \cdot {}^{i}pos_u / W \quad (15)$$

$${}^{i}pos^*_v = c \cdot {}^{i}pos_v / H \quad (16)$$

where c is a constant which we set to 512, and H and W are the height and width of the image feature, respectively. Please note that despite using the same name, our RPE is different from the ones [10, 23] used in natural language problems, which use pair-wise token relation for positional encoding and is fundamentally different from positional encoding for image features.

3.4. Annealed Self-Training

In prior literature [60], self-training was used to leverage the model's prediction on unlabeled samples to obtain additional *pseudo-annotations* for fine-tuning the model. In zero-shot segmentation, the model produces class labels and confidence values (*i.e.*, output of softmax layer) upon encountering pixels of unseen classes. The produced class labels are used as pseudo labels for self-training to learn unseen classes, based on the output confidence values. We anticipate that the generated pseudo-labeled pixels may be incorrect, and therefore, noisy labels may degrade the performance of the segmentation model. Previous methods [5, 18] threshold the prediction confidence and use only high confidence pseudo labels (*e.g.*, highest 75% in [5]) during training to reduce the influence of incorrect pseudo-annotations.

However, finding a suitable threshold is not trivial since the model's confidence in each sample is different. Inspired by knowledge distillation in transfer learning [22], we propose Annealed Self-Training (AST), which uses all pseudo-annotations but assigns different loss weights according to the confidence score, as shown in Eq. (17)

$$w_i = \frac{1}{Z} \frac{\exp(p_i/T)}{\sum_i \exp(p_i/T)} \quad (17)$$

where Z is a normalization term so that the maximum value of loss weights ($\max\{w_i\}$) is 1. The loss re-weighing is achieved by applying *annealing softmax* function on confidence score p, and the annealed temperature T is used to adjust re-weighing intensity. Note that we only do loss re-weighing on pseudo-annotations and the loss weights of seen classes are always 1.

4. Experimental Evaluation

4.1. Benchmark Datasets

Following [53, 18], we used (1) Pascal Visual Object Classes (VOC) [13], (2) Pascal Context [40] and (3) COCO Stuff [6] for evaluation. Pascal VOC contains 20 categories with 1,464 and 1,449 images, for training and testing, respectively. Since Pascal VOC is relatively small, external Semantic Boundaries Dataset (SBD) dataset [20] is also used during training as suggested in previous works [53, 5, 18]. After introducing SBD and excluding duplicate images in Pascal VOC test set, there are 8,284 and 2,299 images for training and validation, respectively. Pascal Context contains 33 categories, including 4998, 500 and 5105 images, for training, validation and testing. COCO Stuff is a large semantic segmentation dataset with 171 categories. There are 118,287 images for training and 5,000 for testing. We split the last 10,000 images in the training set for validation.

We follow the evaluation protocol from [53, 18] for splitting seen and unseen categories. For Pascal VOC, the last five classes (potted plant, sheep, sofa, train, tv-monitor) are used as unseen categories [53, 18]. Class "background" is ignored in Pascal VOC during both training and testing, as suggested by [18], since it is unreasonable to use single semantic word representation for all kinds of background objects (*e.g.*, sky, road). Four categories (cow, motorbike, sofa, cat) are classified as unseen classes in Pascal Context [18]. For COCO Stuff, 15 classes (frisbee, skateboard, cardboard, carrot, scissors, suitcase, giraffe, cow, road, wallconcrete, tree, grass, river, clouds, playingfield) are treated as unseen [53, 18].

4.2. Experiment Setup And Evaluation Metrics

Implementation Details: We use word embeddings from both word2vec [35] and fasttext [24], and concatenate them to represent words (a total of 600 dimensional vector; 300 for each). We follow [18] to use average word embeddings when a category has multiple words. We use Deeplab-v2 [8] built upon ResNet-101 [21] as the semantic segmentation backbone. We apply SGD [45] optimizer for Deeplab-v2 backbone, SIM and classifier with initial learning rate 2.5×10^{-4}, and Adam [27] optimizer for the generator with initial learning rate 2×10^{-4}. A poly learning rate scheduler was applied for backbone as suggested by [53]. We empirically set loss weights to $\alpha = 100$ and $\beta = 50$.
Evaluation Metrics: We report performance based on mean intersection-over-union (mIoU) and conduct evalu-

Table 1. Zero-shot semantic segmentation mIoU performance on Pascal VOC, Pascal Context and COCO Stuff. "ST" and "AST" stand for self-training and annealed self-training, respectively. Evaluation metric is mean intersection over union (mIoU)

	Pascal VOC			Pascal Context			COCO Stuff		
Methods	**Seen(%)**	**Unseen(%)**	**Harmonic(%)**	**Seen(%)**	**Unseen(%)**	**Harmonic(%)**	**Seen(%)**	**Unseen(%)**	**Harmonic(%)**
SPNet [53]	78.00	15.63	26.10	35.14	4.00	7.18	**35.18**	8.73	13.98
ZS3 [5]	77.30	17.65	28.74	33.04	7.68	12.46	34.66	9.53	14.95
CaGNet [18]	**78.40**	26.59	39.72	**36.10**	14.42	20.61	33.49	12.23	18.19
SIGN (Ours)	75.40	**28.86**	**41.74**	33.67	**14.93**	**20.67**	32.31	**15.47**	**20.93**
ZS3 + ST	78.02	21.15	33.28	33.98	9.53	14.88	34.89	10.55	16.20
CaGNet + ST	78.59	30.31	43.66	**36.44**	16.30	22.52	35.55	13.40	19.46
SIGN + ST	78.62	33.12	46.61	34.91	16.71	22.60	**36.39**	15.15	21.39
SIGN + AST	**83.49**	**41.29**	**55.26**	34.90	**17.86**	**23.62**	31.94	**17.53**	**22.64**

ation under generalized zero-shot learning (GZSL) metric. Generalized zero-shot learning assesses performance on seen and unseen classes at the same time, rather than evaluating only on the unseen classes under zero-shot learning (ZSL) metric. Similar to previous work on zero-shot classification [55] and segmentation [53, 5], we report mIoU of seen and unseen categories and harmonic mIoU of seen and unseen categories.

4.3. Comparison With Baselines:

We compare our method against (1) SPNet [53], (2) ZS3 [5] and (3) CaGNet [18]. We do not compare with CSRL [30] since they use a different protocol and their code is not available. For fair comparison, we use the same word2vec [35] and fasttext [24] embeddings, and use the same Deeplab-v2 [8] as the segmentation backbone for all the methods.

We report the performance with and without self-training. Table 1 summarizes the results of different methods. We can see that our SIGN model achieves the best performance on unseen categories and harmonic mIoU on all the three benchmark datasets, which indicates the effectiveness of our method on recognizing unseen categories. In addition, our Annealed Self-Training further improves performance over conventional self-training. Compared to seen categories, AST works better for unseen ones due to higher utilization of pseudo-annotations. We notice that the performance impact of AST on Pascal VOC is higher than Context and COCO Stuff. Performance difference can be attributed to the smaller number of categories in Pascal VOC, which leads to a higher chance of correct pseudo labels.

Fig. 3 shows a qualitative comparison between SIGN and baselines. We can see that SIGN achieves high accuracy, even when there are multiple unseen categories (please see last row).

4.4. Ablation Studies

We conduct ablation studies on Pascal VOC dataset to show the effectiveness of Relative Positional Encoding and Annealed Self-Training.

SIM Architecture: We designed SIM as a residual module [21] and experimented with four architectures. (A) Convolution-based SIM uses simple residual blocks [21] with consecutive convolution and activation layers. (B) Attention-based [18] SIM learns three attention maps of different scales from deep features. Deep features are then element-wise multiplied with attention maps, and concatenated together. (C) Self-attention-based SIM uses the structure of Transformer Encoder [50]. The difference between self-attention- and attention-based SIM is: (1) attention map is computed according to the correlation of pixels on deep features instead of pixel-wise attention, and (2) feature aggregation computes the weighted sum of previous deep features, rather than concatenating features of different scales. (D) Multihead self-attention-based [50] runs several self-attention in parallel. The input feature is first linearly transformed into smaller dimension in each head, and self-attention is applied separately. Then, the attention results are concatenated together and linearly transformed back to the original dimension.

Table 2 summarizes the number of parameters of the four SIM architectures and their performance on Pascal VOC. The best performance is achieved by multihead self-attention-based SIM, followed by attention-based SIM. According to the performance in Table 2, performance numbers in all following experiments are reported based on multihead self-attention-based SIM. Please refer to Appendix A for the detailed model structure.

Relative versus Absolute Positional Encoding: To evaluate the effectiveness of the proposed Relative Positional Encoding, we compare it with two other positional encoding strategies: (1) Absolute Positional Encoding (APE), which use the absolute index of pixel to compute positional vec-

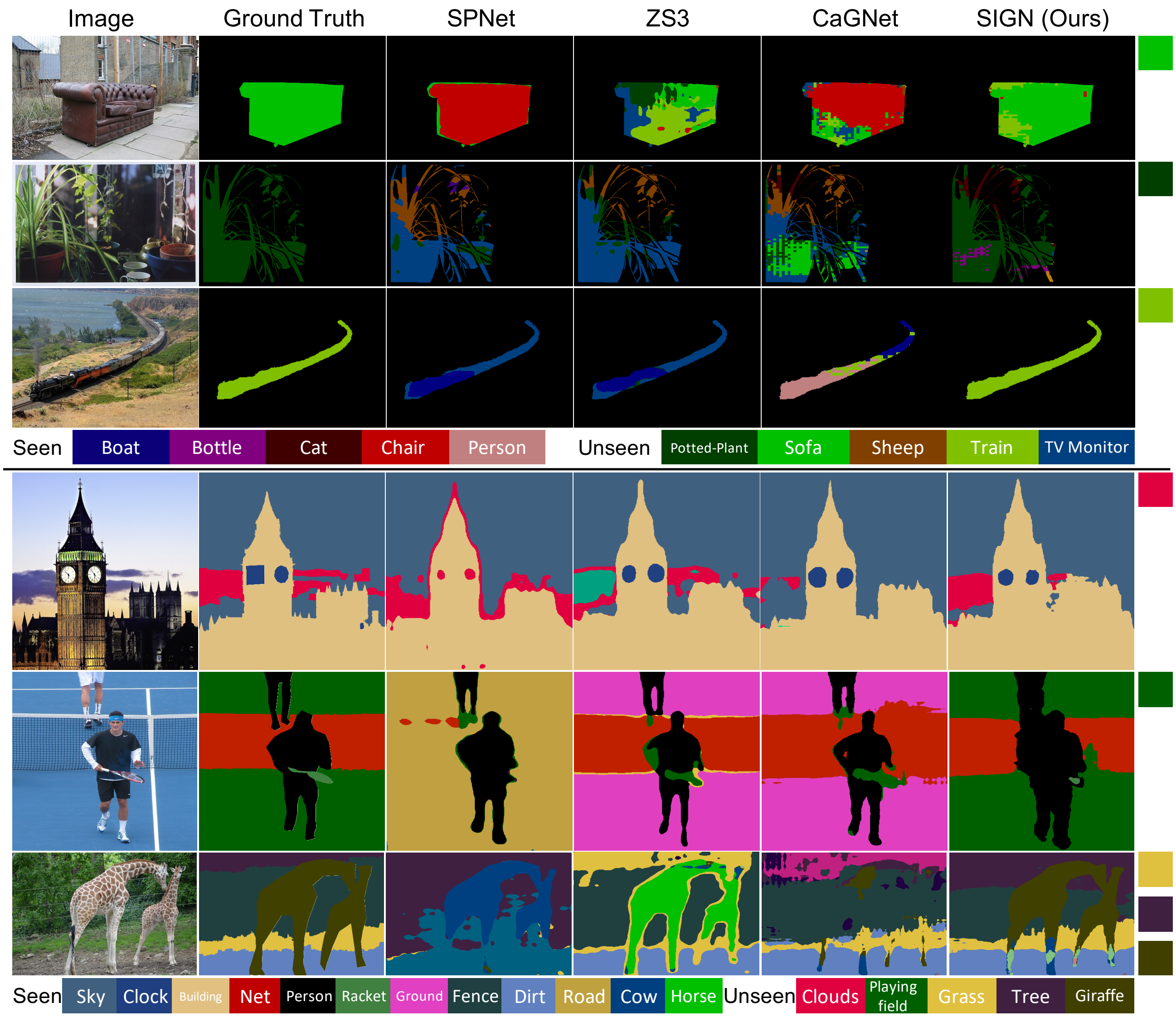

Figure 3. Qualitative comparison with SPNet [53], ZS3 [5] and CaGNet [18]. The top three samples are from Pascal VOC and the bottom three samples are from COCO Stuff. Color bar below the samples indicate the correspondence between colors and categories (including false positive categories). The square(s) on the right indicates the unseen class(es) in the sample on the left.

Table 2. Number of parameters in four different SIM architecture and the corresponding performance on Pascal VOC.

Archi.	Conv	Attention	Self-Attn.	Multi SA
# Params	5.25M	5.24M	4.60M	4.86M
H. mIoU(%)	39.13	40.86	40.51	**41.74**

tor, and (2) Absolute Positional Encoding with Interpolation during testing [12], which computes positional vector based on the training image size and does bilinear interpolation on positional vector during testing.

The results are presented in Table 3. We see that on unseen categories mIoU and harmonic mIoU, RPE improves performance by 3% compared to APE. Adding bilinear interpolation to APE improves performance by roughly 2% but still cannot match the performance of RPE. Interestingly, we notice a performance degradation of APE compared to the model without PE. We speculate that this is due to the mismatch between training and test image size, and due to the larger test image size, APE fails to encode all spatial information. Please note that larger image sizes or even multi-scale input sizes are commonly used during testing, because it can provide better prediction performance [33, 8].

Effect Of Annealed Temperature: In annealed self-training, pseudo-annotations with higher confidence are

Table 3. Mean IoU of model without PE, Absolute PE, Absolute PE with interpolation and Relative PE on Pascal VOC. Numbers in parentheses show the improvement over model without PE.

Methods	Seen(%)	Unseen(%)	Harmonic(%)
w/o PE	71.86	26.07	38.26
APE	70.44	25.68	37.64
APE w/ Inter.	71.17	27.36	39.53
RPE	**75.40** (+3.54)	**28.86** (+2.79)	**41.74** (+3.21)

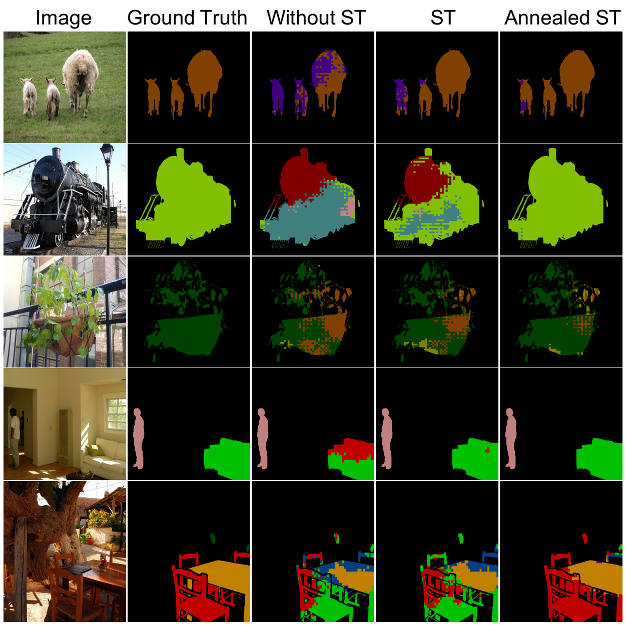

Figure 4. Qualitative comparison of predictions without self-training, with self-training in [5] and our Annealed Self-Training

assigned higher weights in the loss calculation. In the *softmax* function, the annealed temperature controls the smoothness of the output. The higher the temperature, the smoother the output. We adjust the loss weights assigned to high-confidence and low-confidence pseudo-annotations by changing the annealed temperature. Fig. 5 shows the performance curve w.r.t. to annealed temperature on Pascal VOC. We noticed that a temperature of 2 is empirically optimal, and different temperature values leads to lower performance. The reasons for this observation could include (1) for low temperatures, *softmax* function produces sharp output which completely ignores the pseudo-annotations with low confidence, and (2) for high temperatures, due to the smoothness of the output of the *softmax* function, the loss weights assigned to high-confidence and low-confidence pseudo-annotations are too close.

Impact Of Spatial Information On Semantic Segmentation: We tested all four structures of SIM on seen categories before the transfer learning step. In the Deeplab-v2 model without SIM, we slightly increased the number of trainable parameters to ensure that the total parameters are roughly the same as models with SIM. Table 4 shows that even with the simplest convolutional SIM (second column),

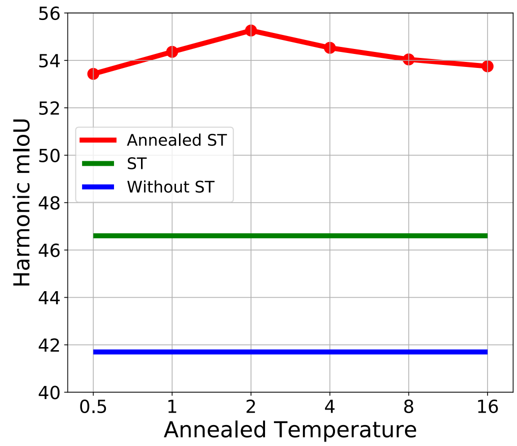

Figure 5. Harmonic mIoU on Pascal VOC under different annealed temperature in Annealed Self-Training (red line).

the performance is improved after adding spatial information. For the SIM based on self-attention and multi-head self-attention, greater improvements can be seen (last two columns). Our hypothesis for this observation is that the self-attention mechanism relaxes the restrictions on the receiving field, so it can make more effective use of spatial information.

Table 4. mIoU(%) on seen categories before transfer learning on Pascal VOC.

w/o SIM	Conv	Attention	Self-att.	Multi SA
76.59	76.83	76.97	77.75	**77.87**

5. Conclusion

We proposed a new zero-shot semantic segmentation framework that incorporates spatial information into prediction. Our method is flexible to handle varying image size using a novel Relative Positional Encoding scheme. We introduced a new self training strategy - Annealed Self Training, which automatically adjusts the importance of pseudo-annotations from prediction confidence. We conducted an extensive experimental study and validated the effectiveness of the proposed RPE and AST, and also investigated network architectures for encoding spatial information. Finally, our SIGN model showed state-of-the-art performance for zero-shot semantic segmentation on benchmark datasets and has the potential to improve performance on conventional semantic segmentation problems.

Acknowledgement This material is based on research sponsored by Air Force Research Laboratory (AFRL) under agreement number FA8750-19-1-1000. The U.S.Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation therein. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, eigher expressed or implied, of Air Force Laboratory, DARPA or the U.S. Government.

References

[1] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1425–1438, 2015. 1, 2

[2] 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. 1, 2

[3] Ziad Al-Halah and Rainer Stiefelhagen. How to transfer? zero-shot object recognition via hierarchical transfer of semantic attributes. In *Proceedings of IEEE Winter Conference on Applications of Computer Vision*, pages 837–843. IEEE, 2015. 1

[4] Ankan Bansal, Karan Sikka, Gaurav Sharma, Rama Chellappa, and Ajay Divakaran. Zero-shot object detection. In *Proceedings of the European Conference on Computer Vision*, pages 384–400, 2018. 1

[5] Maxime Bucher, VU Tuan-Hung, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. In *Advances in Neural Information Processing Systems*, pages 466–477, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[6] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. 5

[7] Long Chen, Hanwang Zhang, Jun Xiao, Wei Liu, and Shih-Fu Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1043–1052, 2018. 1, 2

[8] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017. 4, 5, 6, 7

[9] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *CoRR, abs/1706.05587*, 2017. 1

[10] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Association for Computational Linguistics*, 2019. 5

[11] Berkan Demirel, Ramazan Gokberk Cinbis, and Nazli Ikizler-Cinbis. Zero-shot object detection by hybrid region embedding. 2018. 1

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations9*, 2021. 1, 2, 7

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 5

[14] Rafael Felix, Vijay BG Kumar, Ian Reid, and Gustavo Carneiro. Multi-modal cycle-consistent generalized zero-shot learning. In *Proceedings of the European Conference on Computer Vision*, pages 21–37, 2018. 1, 2

[15] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*, pages 2121–2129, 2013. 1, 2

[16] Zhenyong Fu, Tao Xiang, Elyor Kodirov, and Shaogang Gong. Zero-shot object recognition by semantic manifold distance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2635–2644, 2015. 1

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 1, 2, 4

[18] 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, 3, 4, 5, 6, 7, 13

[19] Yuchen Guo, Guiguang Ding, Jungong Han, and Yue Gao. Zero-shot learning with transferred samples. *Proceedings of IEEE Transactions on Image Processing*, 26(7):3277–3290, 2017. 1, 2

[20] Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *Proceedings of the IEEE International Conference on Computer Vision*, 2011. 5

[21] 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. 4, 5, 6, 13

[22] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *Advances in Neural Information Processing Systems*, 2014. 2, 5

[23] Zhiheng Huang, Davis Liang, Peng Xu, and Bing Xiang. Improve transformer models with better relative position embeddings. *arXiv preprint arXiv:2009.13658*, 2020. 5

[24] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016. 4, 5, 6

[25] Michael Kampffmeyer, Yinbo Chen, Xiaodan Liang, Hao Wang, Yujia Zhang, and Eric P Xing. Rethinking knowledge graph propagation for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11487–11496, 2019. 1, 2

[26] 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*, 2019. 1, 3

[27] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. 5

[28] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. 2014. 4

[29] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. 2

[30] Peike Li, Yunchao Wei, and Yi Yang. Consistent structural relation learning for zero-shot segmentation. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 3, 6

[31] Yujia Li, Kevin Swersky, and Rich Zemel. Generative moment matching networks. In *International Conference on Machine Learning*, pages 1718–1727, 2015. 1, 3, 4

[32] Hugo Liu and Push Singh. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226, 2004. 1

[33] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 4, 7

[34] Yang Long, Li Liu, Fumin Shen, Ling Shao, and Xuelong Li. Zero-shot learning using synthesised unseen visual data with diffusion regularisation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2498–2512, 2017. 1, 2

[35] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013. 1, 3, 4, 5, 6

[36] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 1

[37] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5191–5198, 2020. 2

[38] Ashish Mishra, Shiva Krishna Reddy, Anurag Mittal, and Hema A Murthy. A generative model for zero shot learning using conditional variational autoencoders. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2188–2196, 2018. 1, 2

[39] Pedro Morgado and Nuno Vasconcelos. Semantically consistent regularization for zero-shot recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6060–6069, 2017. 1, 2

[40] 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. 5

[41] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *International Conference on Learning Representations*, 2014. 1, 2

[42] Shafin Rahman, Salman Khan, and Nick Barnes. Polarity loss for zero-shot object detection. *arXiv preprint arXiv:1811.08982*, 2018. 1

[43] Shafin Rahman, Salman Khan, and Fatih Porikli. Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. In *Asian Conference on Computer Vision*, pages 547–563. Springer, 2018. 1

[44] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*, 2021. 1

[45] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. 2014. 5

[46] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning*, pages 2152–2161, 2015. 1, 2

[47] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*, pages 234–241. Springer, 2015. 4

[48] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems*, pages 935–943, 2013. 1, 2

[49] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, pages 3483–3491, 2015. 2

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 1, 4, 6, 13

[51] Xiaolong Wang, Yufei Ye, and Abhinav Gupta. Zero-shot recognition via semantic embeddings and knowledge graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6857–6866, 2018. 1, 2

[52] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 69–77, 2016. 1, 2

[53] 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, 3, 5, 6, 7

[54] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5542–5551, 2018. 1, 2

[55] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4582–4591, 2017. 6

[56] Hang Xu, Chenhan Jiang, Xiaodan Liang, and Zhenguo Li. Spatial-aware graph relation network for large-scale object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9298–9307, 2019. 1

[57] Hang Zhang, Han Zhang, Chenguang Wang, and Junyuan Xie. Co-occurrent features in semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 548–557, 2019. 1

[58] 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. 1, 2

[59] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017. 1, 4

[60] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005. 2, 5