

# Synthesizing the Unseen for Zero-shot Object Detection

Nasir Hayat<sup>1</sup>, Munawar Hayat<sup>1,2</sup>, Shafin Rahman<sup>3</sup>, Salman Khan<sup>1,2</sup>, Syed Waqas Zamir<sup>1</sup>, and Fahad Shahbaz Khan<sup>1,2</sup>

<sup>1</sup>Inception Institute of AI, Abu Dhabi, <sup>2</sup>MBZ University of AI, Abu Dhabi,

<sup>3</sup>North South University, Dhaka.

`nh2218@nyu.edu, {munawar.hayat, salman.khan, fahad.khan}@mbzuai.ac.ae`

**Abstract.** The existing zero-shot detection approaches project visual features to the semantic domain for seen objects, hoping to map unseen objects to their corresponding semantics during inference. However, since the unseen objects are never visualized during training, the detection model is skewed towards seen content, thereby labeling unseen as background or a seen class. In this work, we propose to *synthesize* visual features for unseen classes, so that the model learns both seen and unseen objects in the visual domain. Consequently, the major challenge becomes, *how to accurately synthesize unseen objects merely using their class semantics?* Towards this ambitious goal, we propose a novel generative model that uses class-semantics to not only generate the features but also to discriminatively separate them. Further, using a unified model, we ensure the synthesized features have high diversity that represents the intra-class differences and variable localization precision in the detected bounding boxes. We test our approach on three object detection benchmarks, PASCAL VOC, MSCOCO, and ILSVRC detection, under both conventional and generalized settings, showing impressive gains over the state-of-the-art methods. Our codes are available at [https://github.com/nasir6/zero\\_shot\\_detection](https://github.com/nasir6/zero_shot_detection)

**Keywords:** Zero-shot object detection, generative adversarial learning, visual-semantic relationships.

## 1 Introduction

Object detection is a challenging problem that seeks to simultaneously localize and classify object instances in an image [1]. Traditional object detection methods work in a supervised setting where a large amount of annotated data is used to train models. Annotating object bounding boxes for training such models is a labor-intensive and expensive process. Further, for many rare occurring objects, we might not have any training examples. Humans, on the other hand, can easily identify unseen objects solely based upon the objects' attributes or their natural language description. Zero Shot Detection (ZSD) is a recently introduced paradigm which enables simultaneous localization and classification of

previously *unseen* objects. It is arguably the most extreme case of learning with minimal supervision [2, 3].

ZSD is commonly accomplished by learning to project visual representations of different objects to a pre-defined semantic embedding space, and then performing nearest neighbor search in the semantic space at inference [2–5]. Since the unseen examples are never visualized during training, the model gets significantly biased towards the seen objects [6, 7], leading to problems such as confusion with background and mode collapse resulting in high scores for only some unseen classes. In this work, we are motivated by the idea that if an object detector can visualize the unseen data distribution, the above-mentioned problems can be alleviated. To this end, we propose a conditional feature generation module to synthesize visual features for unseen objects, that are in turn used to directly adapt the classifier head of Faster-RCNN [1]. While such feature synthesis approaches have been previously explored in the context of zero-shot classification, they cannot be directly applied to ZSD due to the unique challenges in detection setting such as localizing multiple objects per image and modeling diverse backgrounds.

The core of our approach is a novel feature synthesis module, guided by semantic space representations, which is capable of generating diverse and discriminative visual features for unseen classes. We generate exemplars in the feature space and use them to modify the projection vectors corresponding to unseen classes in the Faster-RCNN classification head. The major contributions of the paper are: **(i)** it proposes a novel approach to visual feature synthesis conditioned upon class-semantics and regularized to enhance feature diversity, **(ii)** feature generation process is jointly driven by classification loss in the semantic space for both seen and unseen classes, to ensure that generated features are discriminant and compatible with the object-classifier, **(iii)** extensive experiments on Pascal VOC, MSCOCO and ILSVRC detection datasets to demonstrate the effectiveness of the proposed method. For instance, we achieve a relative mAP gain of 53% on MS-COCO dataset over existing state-of-the-art on ZSD task. Our approach is also demonstrated to work favorably well for Generalized ZSD (GZSD) task that aims to detect both *seen* and *unseen* objects.

## 2 Related Work

**Zero-shot Recognition:** The goal of Zero shot learning (ZSL) is to classify images of unseen classes given their textual semantics in the form of wordvecs [8], text-descriptions [5, 9] or human annotated attributes [10]. This is commonly done by learning a joint embedding space where semantics and visual features can interact. The embeddings can be learnt to project from visual-to-semantic [11], or semantic-to-visual space [8]. Some methods also project both visual and semantic features into a common space [12]. The existing methods which learn a projection or embedding space have multiple inherent limitations such as the hubness problem [13] caused by shrunken low dimensional semantic space with limited or no diversity to encompass variations in the visual image space. These

methods are therefore prone to mis-classify unseen samples into seen due to non-existence of training samples for the unseen. Recently, generative approaches deploying variational auto-encoders (VAEs) or generative adversarial networks (GANs) have shown promises for ZSL [14–17]. These approaches model the underlying data distribution of visual feature space by training a generator and a discriminator network that compete in a minimax game, thereby synthesizing features for unseen classes conditioned on their semantic representations.

**Zero-shot Detection:** The existing literature on zero shot learning is dominated by zero shot classification (ZSC). Zero Shot Detection (ZSD), first introduced in [2,3], is significantly more challenging compared with ZSC, since it aims to simultaneously localize and classify an unseen object. [2] maps visual features to a semantic space and enforces max-margin constraints along-with meta-class clustering to enhance inter-class discrimination. The authors in [3] incorporate an improved semantic mapping for the background in an iterative manner by first projecting the seen class visual features to their corresponding semantics and then the background bounding boxes to a set of diverse unseen semantic vectors. [4] learns an embedding space as a convex combination of training class wordvecs. [5] uses a Recurrent Neural Network to model natural language description of objects in the image.

Unlike ZSC, synthetic feature generation for unseen classes is less investigated for ZSD and only [18] augments features. Ours is a novel feature synthesis approach that has the following major differences from [18] **(i)** For feature generation, we only train a single GAN model, in comparison to [18] which trains three isolated models. Our unified GAN model is capable of generating diverse and distinct features for unseen classes. **(ii)** We propose to incorporate a semantics guided loss function, which improves feature generation capability of the generator module for unseen categories. **(iii)** To enhance diversification amongst the generated features, we incorporate a mode seeking regularization term. We further compare our method directly with [18] and show that it outperforms [18] by a significant margin, while using a single unified generation module.

### 3 Method

**Motivation:** Most of the existing approaches for ZSD address this problem in the semantic embedding space. This means that the visual features are mapped to semantic domain where unseen semantics are related with potential unseen object features to predict decision scores. We identify three problems with this line of investigation. **(i) Unseen background confusion:** Due to the low objectness scores for unseen objects, they frequently get confused as background during inference. To counter this, [3,19] use external data in the form of object annotations or vocabulary that are neither seen nor unseen. **(ii) Biasness problem:** Since, unseen objects are never experienced during training, the model becomes heavily biased towards seen classes. For this, approaches usually design specialized loss functions to regularize learning [2,19]. **(iii) Hubness problem:** Only a few unseen classes get the highest scores in most cases. Addressing the problem in semantic

space intensifies the hubness issue [20]. Very recently, GTNet [18] attempted to address these issues in the visual domain instead of the semantic space. Similar to [15], they generate synthesized features to train unseen classes in a supervised manner. We identify two important drawbacks in this approach. **(i)** They train multiple GAN models to incorporate variance due to intra-class differences and varying overlaps with ground-truth (IoU). These generative models are trained in a sequential manner, without an end-to-end learning mechanism, making it difficult to fix errors in early stages. **(ii)** In addition to synthesized unseen object features, they need to generate synthesized background features. As the background semantic is not easy to define, synthesized background features become too noisy than that of object features, thereby significantly hindering the learning process. In this paper, we attempt to solve this problem by training one unified GAN model to generate synthesized unseen object features that can be used to train with real background features without the help of synthesized background features. Further, without requiring multiple sequential generative models to inject feature diversity [18], we propose a simple regularization term to promote diversity in the synthesized features.

### 3.1 Overview

**Problem Formulation:** Consider the train set  $\mathcal{X}^s$  contains image of seen objects and the test set  $\mathcal{X}^u$  contains images of seen+unseen objects. Each image can have multiple objects. Let's denote  $\mathcal{Y}_s = \{1, \dots, S\}$  and  $\mathcal{Y}_u = \{S+1, \dots, S+U\}$  respectively as the label sets for seen and unseen classes. Note that  $S$  and  $U$  denote total number of seen and unseen classes respectively, and  $\mathcal{Y}_s \cap \mathcal{Y}_u = \emptyset$ . At training, we are given annotations in terms of class labels  $y \in \mathcal{Y}_s$  and bounding-box coordinates  $b \in \mathbb{R}^4$  for all seen objects in  $\mathcal{X}^s$ . We are also given semantic embeddings  $\mathbf{W}_s \in \mathbb{R}^{d \times S}$  and  $\mathbf{W}_u \in \mathbb{R}^{d \times U}$  for seen and unseen classes respectively (e.g., Glove [21] and fastText [22]). At inference, we are required to correctly predict the class-labels and bounding-box coordinates for the objects in images of  $\mathcal{X}^u$ . For ZSD settings, only unseen predictions are required, while for generalized ZSD, both seen and unseen predictions must be made.

We outline different steps used for our generative ZSD pipeline in Alg. 1 and Fig. 1 illustrates our method. The proposed ZSD framework is designed to work with any two-stage object detector. For this paper, we implement Faster-RCNN model with ResNet-101 backbone. We first train the Faster-RCNN model  $\phi_{\text{faster-rcnn}}$  on the training images  $\mathcal{X}^s$  comprising of only seen objects and their corresponding ground-truth annotations. Given an input image  $\mathbf{x} \in \mathcal{X}^s$ , it is first represented in terms of activations of a pre-trained ResNet-101. Note that the backbone ResNet-101 was trained on ImageNet data by *excluding* images belonging to the overlapping unseen classes of the evaluated ZSD datasets. The extracted features are feed-forwarded to the region proposal network (RPN) of Faster-RCNN, which generates a set of candidate object bounding box proposals at different sizes and aspect ratios. These feature maps and the proposals are then mapped through an RoI pooling layer, to achieve a fixed-size representation for each proposal. Let's denote the feature maps corresponding to  $K$  bounding

**Algorithm 1** The proposed feature synthesis base ZSD method**Input:**  $\mathcal{X}^s, \mathcal{X}^u, y \in \mathcal{Y}_s, b, \mathbf{W}_s, \mathbf{W}_u$ 

- 1:  $\phi_{\text{faster-rcnn}} \leftarrow$  Train Faster-RCNN using seen data  $\mathcal{X}^s$  and annotations
- 2:  $\mathbf{F}_s, \mathbf{Y}_s \leftarrow$  Extract features for b-boxes of  $\mathcal{X}^s$  using RPN of  $\phi_{\text{faster-rcnn}}$
- 3:  $\phi_{\text{ws-cls}} \leftarrow$  Train  $\phi_{\text{ws-cls}}$  using  $\mathbf{F}_s, \mathbf{Y}_s$
- 4:  $\phi_{\text{wu-cls}} \leftarrow$  Define  $\phi_{\text{wu-cls}}$  using  $\phi_{\text{ws-cls}}$  by replacing  $\mathbf{W}_s$  with  $\mathbf{W}_u$
- 5:  $\mathbf{G} \leftarrow$  Train GAN by optimizing loss in Eq. 1
- 6:  $\tilde{\mathbf{F}}_u, \mathbf{Y}_u \leftarrow$  Synthesize features for unseen classes using  $\mathbf{G}$  and  $\mathbf{W}_u$
- 7:  $\phi'_{\text{cls}} \leftarrow$  Train  $\phi_{\text{cls}}$  using  $\tilde{\mathbf{F}}_u, \mathbf{Y}_u$
- 8:  $\phi_{\text{faster-rcnn}} \leftarrow$  Update  $\phi_{\text{faster-rcnn}}$  with  $\phi'_{\text{cls}}$
- 9: Evaluate  $\phi_{\text{faster-rcnn}}$  on  $\mathcal{X}^u$

**Output:** Class labels and bbox-coordinates for  $\mathcal{X}^u$ 

box proposals of an image with  $\mathbf{f}_i \in \mathbb{R}^{1024}, i = 1, \dots, K$ . The features  $\mathbf{f}_i$  are then passed through two modules: bounding-box-regressor, and object-classifier. Once  $\phi_{\text{faster-rcnn}}$  is trained on the seen data  $\mathcal{X}^s$ , we use it to extract features for seen object anchor boxes. All candidate proposals with an intersection-over-union (IoU)  $\geq 0.7$  are considered as foreground, whereas the ones with IoU  $\leq 0.3$  are considered backgrounds. For  $N_{tr}$  training images in  $\mathcal{X}^s$ , we therefore get bounding-box features  $\mathbf{F}_s \in \mathbb{R}^{1024 \times K \cdot N_{tr}}$  and their class-labels  $\mathbf{Y}_s \in \mathbb{R}^{K \cdot N_{tr}}$ . Next, we learn a unified generative model to learn the relationship between visual and semantic domains.

### 3.2 Unified Generative Model

Given object features  $\mathbf{F}_s$ , their class-labels  $\mathbf{Y}_s$ , and semantic vectors  $\mathbf{W}_s$  for seen training data  $\mathcal{X}^s$ , our goal is to learn a conditional generator  $\mathbf{G} : \mathcal{W} \times \mathcal{Z} \mapsto \mathcal{F}$ , which takes a class embedding  $\mathbf{w} \in \mathcal{W}$  and a random noise vector  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{1}) \in \mathbb{R}^d$  sampled from a Gaussian distribution and outputs the features  $\tilde{\mathbf{f}} \in \mathcal{F}$ . The generator  $\mathbf{G}$  learns the underlying distribution of the visual features  $\mathbf{F}_s$  and their relationship with the semantics  $\mathbf{W}_s$ . Once trained, the generator  $\mathbf{G}$  is used to generate unseen class visual features. Specifically, our feature generation module optimizes the the following objective function,

$$\min_{\mathbf{G}} \max_{\mathbf{D}} \alpha_1 \mathcal{L}_{\text{WGAN}} + \alpha_2 \mathcal{L}_{C_s} + \alpha_3 \mathcal{L}_{C_u} + \alpha_4 \mathcal{L}_{\text{div}}, \quad (1)$$

where  $\mathcal{L}_{\text{WGAN}}$  minimizes the Wasserstein distance, conditioned upon class semantics,  $\mathcal{L}_{C_s}$  ensures the seen class features generated by  $\mathbf{G}$  are suitable and aligned with a pre-trained classifier  $\phi_{\text{cls}}$ , and  $\mathcal{L}_{C_u}$  ensures the synthesized features for unseen classes are aligned with their semantic representations  $\mathbf{W}_u$ .  $\alpha_1, \alpha_2, \alpha_3, \alpha_4$  are the weighting hyper-parameters optimized on a held-out validation set. The proposed approach is able to generate sufficiently discriminative visual features to train the softmax classifier. Each term in Eq. 1 is discussed next.

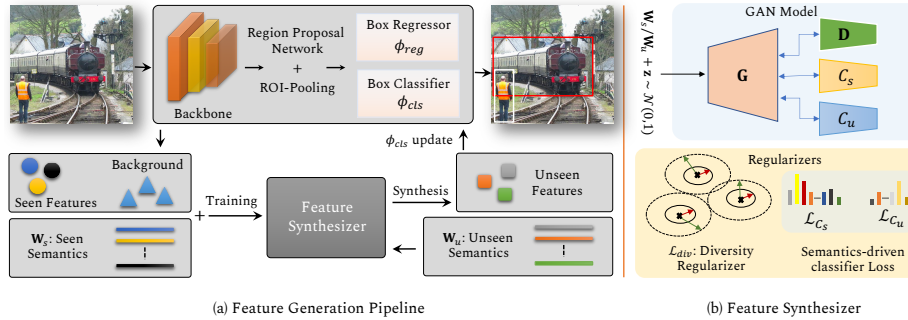


Fig. 1. Overview of proposed generative ZSD approach.

### 3.3 Conditional Wasserstein GAN

We build upon improved WGAN [23] and extend it to conditional WGAN (cWGAN), by integrating the class embedding vectors. The loss  $\mathcal{L}_{\text{WGAN}}$  is given by,

$$\mathcal{L}_{\text{WGAN}} = \mathbb{E}[\mathbf{D}(\mathbf{f}, y)] - \mathbb{E}[\mathbf{D}(\tilde{\mathbf{f}}, y)] + \lambda \mathbb{E}[(\|\nabla_{\tilde{\mathbf{f}}} \mathbf{D}(\tilde{\mathbf{f}}, y)\|_2 - 1)^2], \quad (2)$$

where  $\mathbf{f}$  are the real visual features,  $\tilde{\mathbf{f}} = \mathbf{G}(\mathbf{w}, \mathbf{z})$  denotes the synthesized visual features conditioned upon class semantic vector  $\mathbf{w} \in \mathbf{W}_s$ ,  $\hat{\mathbf{f}} = \alpha \mathbf{f} + (1 - \alpha) \tilde{\mathbf{f}}$ ,  $\alpha \sim \mathcal{N}(0, 1)$  and  $\lambda$  is the penalty coefficient. The first two terms provide an approximation of the Wasserstein distance, while the third term enforces gradients to a unit norm along the line connecting pairs of real and generated features.

### 3.4 Semantically Guided Feature Generation

Our end goal is to augment visual features using the proposed generative module such that they enhance discrimination capabilities of the classifier  $\phi_{\text{cls}}$ . In order to encourage the synthesized features  $\tilde{\mathbf{f}} = \mathbf{G}(\mathbf{w}, \mathbf{z})$  to be meaningful and discriminative, we optimize the loglikelihood of predictions for synthesized seen-class features,

$$\mathcal{L}_{C_s} = -\mathbb{E}[\log p(y|\mathbf{G}(\mathbf{w}, \mathbf{z}); \phi_{\text{cls}})], \quad s.t., \mathbf{w} \in \mathbf{W}_s, \quad (3)$$

where,  $y \in \mathcal{Y}_s$  denotes the ground-truth seen class labels, and  $p(y|\mathbf{G})$  is the class prediction probability computed by the linear softmax classifier  $\phi_{\text{cls}}$ . Note that  $\phi_{\text{cls}}$  was originally trained on the seen data  $\mathcal{X}^s$  and is kept frozen for the purpose of computing  $\mathcal{L}_{C_s}$ . While the conditional Wasserstein GAN captures underlying data distribution of visual features, the  $\mathcal{L}_{C_s}$  term enforces additional constraint and acts as a regularizer to enforce the generated features to be discriminative.

The  $\mathcal{L}_{C_s}$  term in Eq. 3 can act as a regularizer for seen classes only. This is because  $\mathcal{L}_{C_s}$  employs pre-trained  $\phi_{\text{cls}}$  which was learnt for seen data. In order to enhance the generalization capability of our generator  $\mathbf{G}$  towards unseen classes, we propose to incorporate another loss term  $\mathcal{L}_{C_u}$ . For this purpose, we redefine

the classifier head in terms of class semantics, as  $\phi_{\text{ws-cl}s} : \mathbf{f} \rightarrow \mathbf{fc} \rightarrow \mathbf{W}_s \rightarrow \text{softmax} \rightarrow y_{pr}$ , where  $\mathbf{f} \in \mathbb{R}^{1024}$  are the input features,  $\mathbf{fc}$  is the learnable fully-connected layer with weight matrix  $\mathbf{W}_{\text{fc}} \in \mathbb{R}^{1024 \times d}$  and bias  $\mathbf{b}_{\text{fc}} \in \mathbb{R}^d$ ,  $\mathbf{W}_s \in \mathbb{R}^{d \times S}$  are the fixed non-trainable seen class semantics. The outputs of  $\mathbf{fc}$  layer are matrix multiplied with  $\mathbf{W}_s$  followed by softmax operation to compute class predictions  $y_{pr}$ . The classifier  $\phi_{\text{ws-cl}s}$  is trained on the features  $\mathbf{F}_s$  and ground-truth labels  $\mathbf{Y}_s$  of seen class bounding boxes. We can then easily define an unseen classifier  $\phi_{\text{wu-cl}s}$  by replacing the semantics matrix  $\mathbf{W}_s$  in  $\phi_{\text{ws-cl}s}$  with  $\mathbf{W}_u$ . The semantics guided regularizer loss term  $\mathcal{L}_{C_u}$  for synthesized unseen samples is then given by,

$$\mathcal{L}_{C_u} = -\mathbb{E}[\log p(y|\mathbf{G}(\mathbf{w}, \mathbf{z}); \phi_{\text{wu-cl}s})], \quad s.t., \mathbf{w} \in \mathbf{W}_u. \quad (4)$$

The  $\mathcal{L}_{C_u}$  term therefore incorporates the unseen class-semantics information into feature synthesis, by ensuring that unseen features, after being projected onto  $\mathbf{fc}$  layer are aligned with their respective semantics vectors.

### 3.5 Enhancing Synthesis Diversity

Variations in synthesized features are important for learning a robust classifier. Our cWGAN based approach maps a single class semantic vector to multiple visual features. We observed that the conditional generation approach can suffer from mode collapse [24] and generate similar output features conditioned upon prior semantics only, where the noise vectors (responsible for variations in the generated features) get ignored. In order to enhance the diversity of synthesized features, we adapt the mode seeking regularization which maximizes the distance between generations with respect to their corresponding input noise vectors [25]. For this purpose, we define the diversity regularization loss  $\mathcal{L}_{\text{div}}$  as,

$$\mathcal{L}_{\text{div}} = \mathbb{E}[\|\mathbf{G}(\mathbf{w}, \mathbf{z}_1) - \mathbf{G}(\mathbf{w}, \mathbf{z}_2)\|_1 / \|\mathbf{z}_1 - \mathbf{z}_2\|_1]. \quad (5)$$

$\mathcal{L}_{\text{div}}$  encourages the  $\mathbf{G}$  to diversify the synthesized feature space and enhance chances of generating features from minor modes.

### 3.6 Unseen Synthesis and Detection

Optimizing the loss defined in Eq. 1 results in conditional visual feature generator  $\mathbf{G}$ . We can synthesize an arbitrarily large number of features  $\tilde{\mathbf{f}}_u = \mathbf{G}(\mathbf{z}, \mathbf{w})$  for each unseen class by using its corresponding class semantics vector  $\mathbf{w} \in \mathbf{W}_u$  and a random noise vector  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ . Repeating the process for all unseen classes, we get synthesized features  $\tilde{\mathbf{F}}_u$  and their corresponding class-labels  $\mathbf{Y}_u$ , which can then be used to update softmax classifier  $\phi_{\text{cls}}$  of  $\phi_{\text{faster-rcnn}}$  for unseen classes. At inference, a simple forward pass through  $\phi_{\text{faster-rcnn}}$  predicts both class-wise confidence scores and offsets for the bounding-box coordinates. We consider a fixed number of proposals from the RPN (100 in our case) and apply non-maximal suppression (NMS) with a threshold of 0.5 to obtain final

detections. The classification confidence for the proposals are directly given by  $\phi_{c1s}$ , whereas the bounding-box offset coordinates of an unseen class are estimated by the predictions for the seen class with maximum classification response. We observe that this is a reasonable assumption since visual features for the unseen class and its associated confusing seen class are similar. For the case of Generalized zero-shot-detection (GZSD), we simply consider all detections from seen and unseen objects together, whereas for ZSD, detections corresponding to seen objects are only considered.

## 4 Results

**Datasets:** We extensively evaluate our proposed ZSD method on three popular object detection datasets: MSCOCO 2014 [26], ILSVRC Detection 2017 [27] and PASCAL VOC 2007/2012 [28]. For MSCOCO, we use 65/15 seen/unseen split proposed in [19]. As argued in [19], this split exhibits rarity and diverseness of the unseen classes in comparison to another 48/17 split proposed in [3]. We use 62,300 images for training set and 10,098 images from the validation set for testing ZSD and GZSD. For ILSVRC Detection 2017, we follow the 177/23 seen/unseen split proposed in [2] that provides 315,731 training images and 19,008 images for testing. For PASCAL VOC 2007/2012, we follow the 16/4 seen/unseen split proposed in [4] that uses a total of 5,981 images from the train set of 2007 and 2012 and 1,402 images for testing from val+test set of PASCAL VOC 2007. To test the seen detection results, it uses 4,836 images from the test+val set of 2007. For all these datasets, the testing set for ZSD contains at least one unseen object per image.

**Implementation details:** We rescale each image to have the smaller side of 600, 800 and 600 pixels respectively for PASCAL VOC, MSCOCO and ILSVRC Detection datasets. For training our generative module, we consider different anchor bounding boxes with an  $\text{IoU} \geq 0.7$  as foregrounds, whereas  $\text{IoU} \leq 0.3$  boxes are considered as background. We ignore other bounding-boxes with an  $\text{IoU}$  between 0.3 and 0.7, since a more accurate bounding box helps GAN in learning discriminative features. We first train our Faster-RCNN model on seen data for 12 epochs using standard procedure as in [29]. Our category classifier  $\phi_{c1s}$ , and bounding-box regressor  $\phi_{reg}$  both have a single fully-connected layer. The trained model is then used to extract visual features corresponding to bounding-boxes of ground-truth seen objects. We then train our generative model to learn the underlying data distribution of the extracted seen visual features.

The generator  $\mathbf{G}$  and discriminator  $\mathbf{D}$  of our GAN model are simple single-layered neural networks with 4096 hidden units. Through out our experiments, the loss re-weighting hyper-parameters in Eq. 1 are set as,  $\alpha_1 = 1.0, \alpha_2 = 0.1, \alpha_3 = 0.1, \alpha_4 = 1.0$ , using a small held-out validation set. The noise vector  $\mathbf{z}$  has the same dimensions as the class-semantics vector  $\mathbf{w} \in \mathbb{R}^d$  and is drawn from a unit Gaussian distribution with zero mean. We use  $\lambda = 10$  as in [23]. For training of our cWGAN model, we use Adam optimizer with learning rate  $10^{-4}$ ,  $\beta_1 = 0.5, \beta_2 = 0.999$ . The loss term  $\mathcal{L}_{C_u}$  is included after first 5 epochs,



when the generator  $\mathbf{G}$  has started to synthesize meaningful features. Once the generative module is trained, we synthesize 300 features for each unseen class, conditioned upon their class-semantics, and use them to train  $\phi_{\text{cls}}$  for 30 epochs using Adam optimizer. To encode class-labels, unless mentioned otherwise, we use the FastText [30] embedding vectors learnt on large corpus of non-annotated text. The implementation of the proposed method in Pytorch is available at [https://github.com/nasir6/zero\\_shot\\_detection](https://github.com/nasir6/zero_shot_detection)

**Evaluation metrics:** Following previous works [3, 19], we report recall@100 (RE) and mean average precision (mAP) with IoU=0.5. We also report per-class average precision (AP) to study category-wise performance. For GZSD, we report Harmonic Mean (HM) of performances for seen and unseen classes.

#### 4.1 Comparisons with the State-of-the-Art

**Comparison methods:** We compare our method against a number of recently proposed state-of-the-art ZSD and GZSD methods. These include: **(a) SB, LAB** [3], which is a background-aware approach that considers external annotations from object instances belonging to neither seen or unseen. This extra information helps SB, LAB [3] to address the confusion between unseen and background. **(b) DSES** [3] is a version of above approach that does not use background-aware representations but employs external data sources for background. **(c) HRE** [4]: A YOLO based end-to-end ZSD approach based on the convex combination of region embeddings. **(d) SAN** [2]: A Faster-RCNN based ZSD approach that takes advantage of super-class information and a max-margin loss to understand unseen objects better. **(e) PL-48, PL-65** [19]: A RetinaNet based ZSD approach that uses polarity loss for better alignment of visual features and semantics. **(f) ZSDTD** [5]: This approach uses textual description instead of a single-word class-label to define semantic representation. The additional textual description enriches the semantic space and helps to better relate semantics with the visual features. **(g) GTNet** [18]: uses multiple GAN models alongwith textual descriptions similar to [5], to generate unseen features to train a Faster-RCNN based ZSD model in a supervised manner. **(h) Baseline:** The baseline method trains a standard Faster-RCNN model for seen data  $\mathcal{X}^s$ . To extend it to unseen classes for ZSD, it first gets seen predictions  $\mathbf{p}_s$ , and then project them onto class semantics to get unseen predictions  $\mathbf{p}_u = \mathbf{W}_u \mathbf{W}_s^T \mathbf{p}_s$  as in [19]. **(i) Ours:** This is our proposed ZSD approach.

**MSCOCO results:** Our results and comparisons with different state-of-the-art methods for ZSD and GZSD on MSCOCO dataset are presented in Table 1.

*(a) ZSD results:* The results demonstrate that our proposed method achieves a significant gain on both metrics (mAP and RE) over the existing methods on ZSD setting. The gain is specifically pronounced for the mAP metric, which is more challenging and meaningful to evaluate object detection algorithms. This is because mAP penalizes false positives while the RE measure does not impose any penalty on such errors. Despite the challenging nature of mAP metric,

**Table 1.** ZSD and GZSD performance of different methods on MSCOCO in terms of mAP and recall (RE). Note that our proposed feature synthesis based approach achieves a significant gain over the existing state-of-the-art. For the mAP metric, compared with the second best method PL-65 [19], our method shows a relative gain of 53% on ZSD and 38% on harmonic mean of seen and unseen for GZSD.

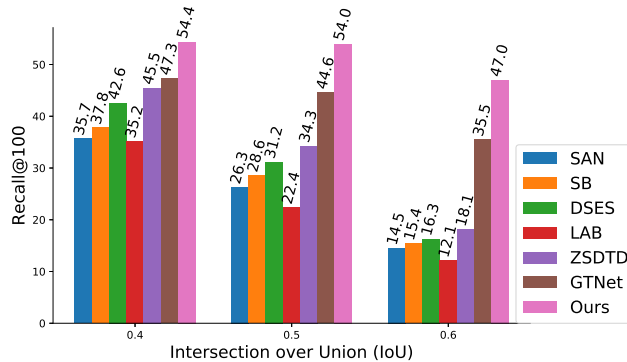
Metric	Method	Seen/Unseen split	ZSD	GZSD		
				seen	unseen	HM
mAP	SB [3]	48/17	0.70	-	-	-
	DSES [3]	48/17	0.54	-	-	-
	PL-48 [19]	48/17	10.01	35.92	4.12	7.39
	PL-65 [19]	65/15	12.40	34.07	12.40	18.18
	Baseline	65/15	8.80	36.60	8.80	14.19
	Ours	65/15	<b>19.0</b>	<b>36.90</b>	<b>19.0</b>	<b>25.08</b>
RE	SB [3]	48/17	24.39	-	-	-
	DSES [3]	48/17	27.19	15.02	15.32	15.17
	PL-48 [19]	48/17	43.56	38.24	26.32	31.18
	PL-65 [19]	65/15	37.72	36.38	37.16	36.76
	Baseline	65/15	44.40	56.40	44.40	49.69
	Ours	65/15	<b>54.0</b>	<b>57.70</b>	<b>53.90</b>	<b>55.74</b>

**Table 2.** Class-wise AP comparison of different methods on unseen classes of MSCOCO for ZSD. The proposed method shows significant gains for a number of individual classes. Compared with the second best method PL [19], our method shows an absolute mAP gain of 6.6%.

Method	Overall	aeroplane	train	parking meter	cat	bear	suitcase	frisbee	snow-board	fork	sand-wich	hot dog	toilet	mouse	toaster	hair drier
PL-Base [19]	8.48	4.0	28.7	.29	18.0	0.0	13.1	11.3	24.3	13.8	9.6	2.0	1.1	.24	.73	0.0
PL [19]	12.40	20.0	48.2	.63	28.3	13.8	12.4	21.8	15.1	8.9	8.5	.87	5.7	.04	1.7	.03
Ours-Baseline	8.80	1.9	31.8	0.0	59.3	3.8	0.6	0.1	19.6	10.7	2.8	0.0	0.8	0.0	0.0	0.0
Ours	<b>19.0</b>	10.1	<b>48.7</b>	<b>1.2</b>	<b>64.0</b>	<b>64.1</b>	12.2	0.7	<b>28.0</b>	<b>16.4</b>	<b>19.4</b>	0.1	<b>18.7</b>	<b>1.2</b>	0.5	<b>0.2</b>

our method achieves a relative mAP gain of 53% over the second-best method (PL [19]). We attribute such remarkable improvement to the fact that our approach addresses the zero shot learning problem by augmenting the visual features. In contrast, previous approaches such as SB [3], DSES [3], PL [19] map visual features to the semantic space that limits their flexibility to learn strong representations mainly due to the noise in semantic domain. In comparison, our approach helps in reducing the biases towards the seen classes during training, avoids unseen-background confusion, and minimizes the hubness problem.

In Fig. 2, we further show comparisons for ZSD recall@100 rates by varying the IoU. Note that the compared methods in Fig. 2 use additional information in the form of textual description of concepts instead of a single-word class name. Even though, our proposed method uses much simpler semantic information (only semantic vectors for class labels), the results in Fig. 2 indicate that our method consistently outperforms several established methods by a large margin



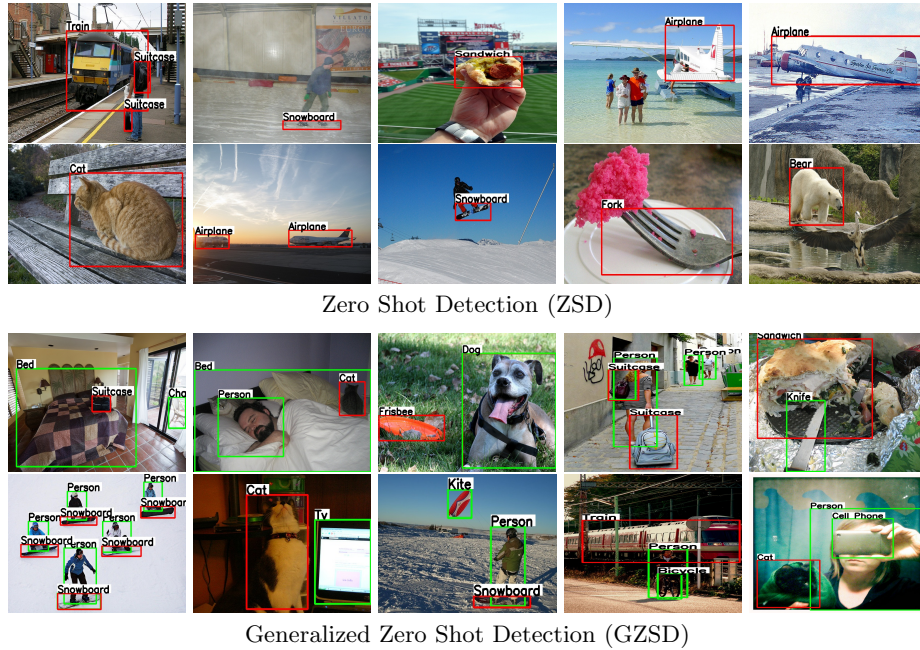
**Fig. 2.** Comparison of SAN [2], SB/DES/LAB [3], ZSDTD [5], GTNet [18] in terms of Recall@100 rates for different IoU settings on MSCOCO dataset. The proposed method consistently shows improved performance over existing state-of-the-art methods.

for a variety of IoU settings. This comparison includes a recent generative ZSD approach, GTNet [18], that employs an ensemble of GANs to synthesize features.

(b) *GZSD results:* Our GZSD results in Table 1 also achieve a significant boost in performance. The generated synthesized features allow training of the detection model in a supervised manner. In this way, unseen instances get equal emphases as seen class objects during training. We note that the GZSD setting is more challenging and realistic since both seen and unseen classes are present at inference. An absolute HM mAP gain of 6.9% for GZSD is therefore quite significant for our proposed method.

Compared with the baseline, which projects visual features to semantic space, our results demonstrate the effectiveness of augmenting the visual space, and learning a discriminative classifier for more accurate classification. These baseline results further indicate the limitations of mapping multiple visual features to a single class-semantic vector. One interesting trend is that the baseline still performs reasonably well according to the RE measure (in some cases even above the previous best methods), however the considerably low mAP scores tell us that the inflated performance from the baseline is prone to many false positives, that are not counted in the RE measure. For this reason, we believe the mAP scores are a more faithful depiction of ZSD methods.

(c) *Class-wise performances:* Our class-wise AP results on MSCOCO in Table 2 show that the performance gain for the proposed method is more pronounced for ‘train’, ‘bear’ and ‘toilet’ classes. Since our feature generation is conditioned upon class-semantics, we observe that the feature synthesis module generates more meaningful features for unseen classes which have similar semantics in the seen data. The method shows worst performance for classes ‘parking-meter’, ‘frisbee’, ‘hot dog’ and ‘toaster’. These classes do not have close counterparts



**Fig. 3.** Qualitative results on MSCOCO for ZSD (top 2 rows) and GZSD (bottom 2 rows). Seen classes are shown with green and unseen with red. (*best seen when zoomed*)

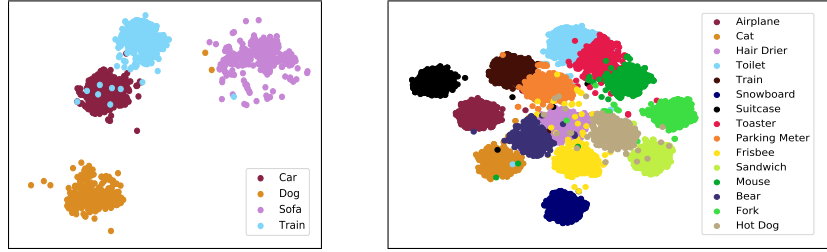
among the seen classes, which makes their detection harder.

(d) *Qualitative results:* Fig. 3 shows some examples of detections from our method both for ZSD (top 2 rows) and GZSD (bottom 2 rows) settings. The visual results demonstrate the effectiveness of the proposed method in localizing unseen objects, and its capability to detect multiple seen+unseen objects with challenging occlusions and background clutter in real-life images.

**PASCAL VOC results:** In Table 3, we compare different methods on PASCAL VOC dataset based on the setting mentioned in [4]. The results suggest that the proposed method achieves state-of-the-art ZSD performance. A few examples images on PASCAL VOC shown in Fig. 5 demonstrate the capability of our method to detect multiple unseen objects in real-life scenarios. The results in Table 3 further indicate that in addition to the unseen detection case, our method performs very well in the traditional seen detection task. We outperform the current best model PL [19] by a significant margin, i.e., 73.6% vs. 63.5% for seen detection and 64.9% vs. 62.1% for unseen detection. A t-SNE visualization of our synthesized features for unseen classes is shown in Fig. 4. We observe that our generator can effectively capture the underlying data distribution of visual features. The similar classes occur in close proximity of each other. We further

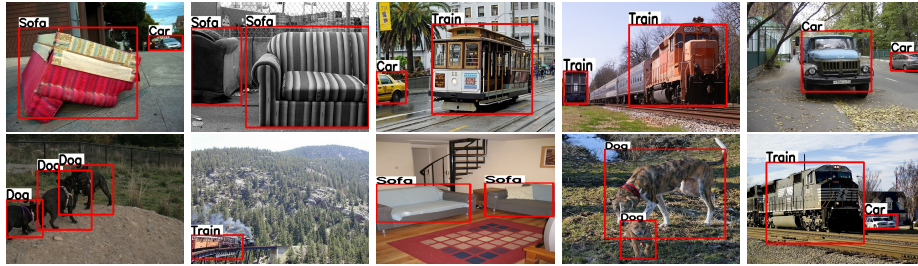
**Table 3.** mAP scores on PASCAL VOC’07. *Italic* classes are unseen.

Method	Seen	Unseen	aeroplane	bicycle	bird	boat	bottle	bus	cat	chair	cow	d.table	horse	motorbike	person	p.plant	sheep	tvmonitor	<i>car</i>	<i>dog</i>	<i>sofa</i>	<i>train</i>
HRE [4]	57.9	54.5	68.0	72.0	74.0	48.0	41.0	61.0	48.0	25.0	48.0	73.0	75.0	71.0	73.0	33.0	59.0	57.0	55.0	82.0	55.0	26.0
PL [19]	63.5	62.1	74.4	71.2	67.0	50.1	50.8	67.6	84.7	44.8	68.6	39.6	74.9	76.0	79.5	39.6	61.6	66.1	63.7	87.2	53.2	44.1
Ours	<b>73.6</b>	<b>64.9</b>	<b>83.0</b>	<b>82.8</b>	<b>75.1</b>	<b>68.9</b>	<b>63.8</b>	<b>69.5</b>	<b>88.7</b>	<b>65.1</b>	<b>71.9</b>	<b>56.0</b>	<b>82.6</b>	<b>84.5</b>	<b>82.9</b>	<b>53.3</b>	<b>74.2</b>	<b>75.1</b>	59.6	<b>92.7</b>	<b>62.3</b>	<b>45.2</b>



**Fig. 4.** A t-SNE visualization of synthesized features by our approach for unseen classes on PASCAL VOC dataset (left) and MSCOCO dataset (right). The generated features form well-separated and distinctive clusters for different classes.

observe that the synthesized features form class-wise clusters that are distinctive, thus aiding in learning a discriminative classifier on unseen classes. Synthesized features for similar classes (*bus* and *train*) are however sometimes confused with each other due to high similarity in their semantic space representation.



**Fig. 5.** Example unseen detections on PASCAL VOC. (*best seen when zoomed*).

**ILSVRC DET 2017 results:** In Table 4, we report ZSD results on ILSVRC Detection dataset based on the settings mentioned in [2]. We can notice from the results that, in most of the object categories, we outperform our closed competitor SAN [2] by a large margin. Note that for a fair comparison, we do not compare our method with reported results in [5, 18], since both these methods use additional information in the form of textual description of class-

**Table 4.** ZSD class-wise AP for unseen classes of ILSVRC DET 2017 dataset.

	mean	p.box	syringe	harmonica	maraca	burrito	pineapple	electric-fan	iPod	dishwasher	canopener	plate-rack	bench	bowtie	s.trunk	scorpion	snail	hamster	tiger	ray	train	unicycle	golfball	h.bar
Baseline	12.7	0.0	3.9	0.5	0.0	36.3	2.7	1.8	1.7	12.2	2.7	7.0	1.0	0.6	22.0	19.0	1.9	40.9	75.3	0.3	28.4	17.9	12.0	4.0
SAN ( $L_{mm}$ )	15.0	0.0	8.0	0.2	0.2	39.2	2.3	1.9	3.2	11.7	4.8	0.0	0.0	7.1	23.3	25.7	5.0	50.5	75.3	0.0	44.8	7.8	28.9	4.5
SAN [2]	16.4	5.6	1.0	0.1	0.0	27.8	1.7	1.5	1.6	7.2	2.2	0.0	4.1	5.3	26.7	65.6	4.0	47.3	71.5	21.5	51.1	3.7	26.2	1.2
Ours	<b>24.3</b>	<b>6.2</b>	<b>18.6</b>	<b>0.7</b>	<b>5.9</b>	<b>50.9</b>	<b>8.2</b>	<b>2.1</b>	<b>55.3</b>	11.5	<b>14.3</b>	3.0	<b>15.4</b>	2.7	11.4	41.9	<b>16.4</b>	<b>79.6</b>	67.6	14.5	<b>69.5</b>	<b>31.8</b>	<b>30.7</b>	0.1

labels. It has been previously shown in [5] that the additional textual description information boosts performance across the board. For example, in their paper, SAN [2] reports an mAP of 16.4 using single-word description for class-labels, whereas, [5] reports an mAP of 20.3 for SAN using multi-word textual description of class-labels. Our improvements over SAN again demonstrates the significance of the proposed generative approach for synthesizing unseen features.

**Distinct Foreground Bounding Boxes:** The seen visual features are extracted based upon the anchor bounding boxes generated by using the ground-truth bounding boxes for seen classes in  $\mathcal{X}^s$ . We perform experiments by changing the definition of background and foreground bounding-boxes. Specifically, we consider two settings: **(a)** Distinct bounding-boxes: foreground object has a high overlap ( $\text{IoU} \geq 0.7$ ), and the background has minimal overlap with the object ( $\text{IoU} \leq 0.3$ ), and **(b)** Overlapping bounding-boxes: foreground has a medium overlap with the object of interest ( $\text{IoU} > 0.5$ ), and some background boxes have medium overlap with the object ( $\text{IoU} < 0.5$ ). We achieve an mAP of 19.0 vs 11.7 for distinct and overlapping bounding boxes respectively on MSCOCO 65/15 split. This suggests that the generative module synthesizes the most discriminant features when the bounding-boxes corresponding to the real visual features have a high overlap with the respective object and minimal background.

## 5 Conclusion

The paper proposed a feature synthesis approach for simultaneous localization and categorization of objects in the framework of ZSD and GZSD. The proposed method can effectively learn the underlying visual-feature data distribution, by training a generative adversarial network model conditioned upon class-semantic. The GAN training is driven by a semantic-space unseen classifier, a seen classifier and a diversity enhancing regularizer. The method can therefore synthesize high quality unseen features which are distinct and discriminant for the subsequent classification stage. The proposed framework generalizes well to both seen and unseen objects and achieves impressive performance gains on a number of evaluated benchmarks including MSCOCO, PASCAL VOC and ILSVRC detection datasets.

## References

1. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*. (2015) 91–99
2. Rahman, S., Khan, S., Porikli, F.: Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. In: *Asian Conference on Computer Vision*, Springer (2018) 547–563
3. Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: Zero-shot object detection. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. (2018) 384–400
4. Demirel, B., Cinbis, R.G., Ikizler-Cinbis, N.: Zero-shot object detection by hybrid region embedding. *arXiv preprint arXiv:1805.06157* (2018)
5. Li, Z., Yao, L., Zhang, X., Wang, X., Kanhere, S., Zhang, H.: Zero-shot object detection with textual descriptions. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Volume 33. (2019) 8690–8697
6. Hayat, M., Khan, S., Zamir, S.W., Shen, J., Shao, L.: Gaussian affinity for max-margin class imbalanced learning. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. (2019)
7. Khan, S., Hayat, M., Zamir, S.W., Shen, J., Shao, L.: Striking the right balance with uncertainty. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (2019)
8. Zhang, L., Xiang, T., Gong, S.: Learning a deep embedding model for zero-shot learning. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. (2017) 2021–2030
9. Lei Ba, J., Swersky, K., Fidler, S., et al.: Predicting deep zero-shot convolutional neural networks using textual descriptions. In: *Proceedings of the IEEE International Conference on Computer Vision*. (2015) 4247–4255
10. Annadani, Y., Biswas, S.: Preserving semantic relations for zero-shot learning. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. (2018) 7603–7612
11. Lampert, C.H., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. *IEEE transactions on pattern analysis and machine intelligence* **36** (2013) 453–465
12. Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B.: Evaluation of output embeddings for fine-grained image classification. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. (2015) 2927–2936
13. Dinu, G., Lazaridou, A., Baroni, M.: Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568* (2014)
14. Chen, L., Zhang, H., Xiao, J., Liu, W., Chang, S.F.: Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. (2018) 1043–1052
15. Xian, Y., Lorenz, T., Schiele, B., Akata, Z.: Feature generating networks for zero-shot learning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2018) 5542–5551
16. Zhu, Y., Elhoseiny, M., Liu, B., Peng, X., Elgammal, A.: A generative adversarial approach for zero-shot learning from noisy texts. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2018) 1004–1013
17. Khan, S.H., Hayat, M., Barnes, N.: Adversarial training of variational auto-encoders for high fidelity image generation. *Proceedings of the IEEE Winter Conference on Applications of Computer Vision* (2018)

18. Zhao, S., Gao, C., Shao, Y., Li, L., Yu, C., Ji, Z., Sang, N.: Gtnet: Generative transfer network for zero-shot object detection. arXiv (2020) arXiv-2001
19. Rahman, S., Khan, S., Barnes, N.: Polarity loss for zero-shot object detection. arXiv preprint arXiv:1811.08982 (2018)
20. Zhang, L., Xiang, T., Gong, S.: Learning a deep embedding model for zero-shot learning. In: CVPR. (2017)
21. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). (2014) 1532–1543
22. Joulin, A., Grave, É., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. (2017) 427–431
23. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Advances in neural information processing systems. (2017) 5767–5777
24. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in neural information processing systems. (2016) 2234–2242
25. Mao, Q., Lee, H.Y., Tseng, H.Y., Ma, S., Yang, M.H.: Mode seeking generative adversarial networks for diverse image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 1429–1437
26. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV, Springer (2014) 740–755
27. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* **115** (2015) 211–252
28. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. *International journal of computer vision* **88** (2010) 303–338
29. Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C.C., Lin, D.: MMDetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155 (2019)
30. Mikolov, T., Grave, E., Bojanowski, P., Puhersch, C., Joulin, A.: Advances in pre-training distributed word representations. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018). (2018)