Semantics-Guided Contrastive Network for Zero-Shot Object detection

Caixia Yan, Xiaojun Chang, Minnan Luo, Huan Liu, Xiaoqin Zhang, and Qinghua Zheng

Abstract—Zero-shot object detection (ZSD), the task that extends conventional detection models to detecting objects from unseen categories, has emerged as a new challenge in computer vision. Most existing approaches on ZSD are based on a strict mapping-transfer strategy that learns a mapping function from visual to semantic space over seen categories, then directly generalizes the learned mapping function to unseen object detection. However, the ZSD task still remains challenging, since those works fail to consider the two key factors that hamper the ZSD performance: (a) the domain shift problem between seen and unseen classes leads to poor transferable ability of the model; (b) the original visual feature space is suboptimal for ZSD since it lacks discriminative information. To alleviate these issues, we develop a novel Semantics-Guided Contrastive Network for ZSD (ContrastZSD), a detection framework that first brings the contrastive learning paradigm into the realm of ZSD. Particularly, ContrastZSD incorporates two semantics-guided contrastive learning tasks that contrast between region-category and region-region pairs respectively. The pairwise contrastive tasks take advantage of class label and semantic relation as additional supervision signals. Under the guidance of those explicit semantic supervision, the model can learn more knowledge about unseen categories to avoid over-fitting to the seen concepts, while optimizing the data structure of both visual features and semantic embeddings in the joint embedding space for better visual-semantic alignment. Extensive experiments are conducted on two popular benchmarks for ZSD, *i.e.*, PASCAL VOC and MS COCO. Results show that our method outperforms the previous state-of-the-art on both ZSD and generalized ZSD tasks.

Index Terms—Object detection, zero-shot learning, zero-shot object detection, supervised contrastive learning.

1 Introduction

Because of its importance to image understanding and analysis, object detection has received increasing attention in recent years [1], [2], [3]. With the impressive development of deep learning, a surge of novel detection models built upon deep Convolutional Neural Networks (CNNs) have been developed in recent years, pushing the detection performance forward remarkably [4], [5], [6], [7]. The most state-of-the-art object detection models follow a region proposal based paradigm [1], [8], [9], which detect objects by (1) first generating region proposals as candidates that might have objects within them, and (2) then performing bounding box regression and classification simultaneously on each proposal. Despite their efficacy, the detection performance of these methods purely relies on the discriminative capabilities of region features, which often depends on sufficient training data with complete annotations for each category. However, labeling for object detection, which requires a pair of a class label and a bounding box location for each object within each image, is both prohibitively costly and labor-intensive. Furthermore, even if all the data samples can be well annotated, we still face the problem of data scarcity, due to the fact that novel categories (*e.g.*, rare animals) are constantly emerging in practical scenarios [10]. In such a scenario, the traditional object detection models often become infeasible because scarce or even no visual data from those novel categories is available for model training. The above mentioned issues, namely the burden of manual labelling and the problem of data scarcity, lead us to investigate the detection task with additional source of complexity, *i.e.*, zero-shot object detection (ZSD). The goal of ZSD is to concurrently recognize and localize objects from previously unseen categories, thereby scaling the traditional detection problem to a far more challenging zero-shot setting. Compared with conventional zero-shot recognition (ZSL) task [11], [12], [13], the problem setting of ZSD gives rise to its unique challenges: (1) Conventional ZSL only needs to recognize one dominant object in each image, while ZSD aims to detect candidate boxes from multiple categories. (2) In addition to class label prediction in ZSL, part of ZSD is predicting the bounding box location of each object. On this proviso, it is still far from optimal to apply existing ZSL methods directly to the ZSD task.

Recently, preliminary efforts have been put into the study of zero-shot object detection (ZSD) [14], [15], [16], [17], [18]. Most of these methods follow a strict mapping-transfer strategy that tackle the ZSD task with a two-stage pipeline. During the training stage, a mapping function is learned to project the seen visual features and the corresponding semantic vectors to a joint embedding space. In previous literature, there are three types of joint embedding space for ZSD models: learning a common intermediate embedding space between the visual space and semantic space [17], learning an embedding from visual space to semantic space [14], [15], [16], or learning an embedding from semantic space to visual space [19], [20]. Subsequently, both visual

- *Caixia Yan, Minnan Luo, Huan Liu and Qinghua Zheng are with the Ministry of Education Key Lab of Intelligent Networks and Network Security, National Engineering Lab for Big Data Analytics, School of Electronic and Information Engineering, Xi'an Jiaotong University, Shaanxi 710049, China, e-mail: yancaixia@stu.xjtu.edu.cn, {minnluo,huanliu,qhzheng}@mail.xjtu.edu.cn*
- *Xiaojun Chang is with the Faculty of Information Technology, Monash University, Australia, e-mail: cxj273@gmail.com.*
- *Xiaoqin Zhang is with the College of Computer Science and Artificial Intelligence, Wenzhou University, China.*

Manuscript received April 19, 2005; revised August 26, 2015.

features and semantic representations are compared directly in the joint embedding space using a compatibility function, such that its score for the correct class is higher than that for an incorrect class by a fixed margin. At the testing stage, the mapping function learned on labeled visual data from seen classes is directly applied to project the visual features and semantic representations of the unseen test classes into the joint embedding space, followed by a nearest neighbor search for unseen class label prediction. Despite their efficacy, we have noted that all these efforts fail to consider the domain shift problem between seen and unseen categories, leading to limited transferable ability of the model [21], [22]. Especially in the more challenging generalized zero-shot object detection (GZSD) setting, where the test samples may come from either seen or unseen classes, the domain shift problem would degrade the performance significantly. This is because the learned models merely rely on the visual data and class embeddings from seen categories, thus the objects from unseen categories tend to be recognized as seen class objects at the testing stage.

To alleviate the domain shift problem, an alternative paradigm has been developed to tackle the ZSD task with Generative Adversarial Network (GAN) [23], [24], [25]. This paradigm can generate synthetic samples for unseen classes conditioned on their attribute information to compensate for the lack of training samples of unseen classes. Both the real seen training features and the synthetic unseen features are used for model training, yielding a fully-observed training set for both seen and unseen classes, thus converts ZSD to the conventional detection task. Those feature generating based ZSD methods synthesize visual features in the original visual space. However, issues with visual-semantic gap still exist [26], [27], meaning that the visual features synthesized by those feature generating methods are located in different structural spaces with the class semantic embeddings and thus are lack of discriminative ability. Taking both the domain shift problem and visual-semantic gap into consideration, a simple node-to-node projection across different spaces, as shown in Fig. 1(a), may not align the visual features and semantic embeddings well.

To tackle the above challenges, in this paper, we develop a semantics-guided contrastive network, namely ContrastZSD, that seeks to bridge the visual-semantic gap and simultaneously alleviate the domain shift problem for improved zero-shot detection. Particularly, we build our ContrastZSD framework on top of the popular Faster R-CNN architecture due to its simplicity yet effectiveness for object detection. Equipped with the similar region feature encoding network as Faster R-CNN, ContrastZSD first extracts the global feature maps from the input images with CNN backbone, then produces region proposals in an objectiveness manner using the region proposal network (RPN). Subsequently, both the region features and semantic embeddings are mapped to a joint embedding space for visual-semantic alignment. Unlike most existing works on ZSD that learn projection function from visual to semantic space, a common intermediate embedding space is learned in ContrastZSD, making it possible to adjust the data structures of both semantic vectors and visual features. As illustrated in Fig. 1(b), when mapping the seen region proposals and semantic embeddings to the common space, ContrastZSD incorporates two semantics-guided contrastive learning subnets for better visual-semantic matching: (1) a region-category contrastive learning (RCCL) subnet, which is the key component that endows our model with the ability of detecting unseen objects. It contrasts seen region proposals with both seen and unseen class embeddings to prevent the network from over-fitting the seen classes, thereby alleviating the domain shift problem; (2) a region-region contrastive learning (RRCL) subnet, which regulates the region feature distribution by resorting to the class label information, thereby inducing semanticity to the embedding space. To optimize the deep network defined above, we further design a novel multi-task loss that includes both the classification, bounding box regression and contrastive loss. Our main contributions are summarized below:

- We apply contrastive learning mechanism for the ZSD task, and develop a novel semantics-guided contrastive network to address the issue of domain shift and visual-semantic gap in ZSD. To the best of our knowledge, this is the first work to apply contrastive learning mechanism for ZSD.
- The proposed deep model incorporates both region-category and region-region contrastive learning to optimize the visual and semantic data structure in the joint embedding space, with a boost to visual-semantic mapping as a byproduct.
- We extend the conventional self-supervised contrastive learning to a supervised paradigm, and design novel contrastive learning losses supervised by explicit semantics to guarantee both the discriminative and transferable property of the proposed ZSD model.
- We conduct extensive experiments on popular object detection datasets, *i.e.*, PASCAL VOC and MS COCO, to demonstrate the effectiveness and superiority of the proposed model over both the ZSD and GZSD task.

2 Related Works

In this section, we briefly review the related works on the fields relevant to our study: object detection, zero-shot learning, zero-shot object detection and supervised contrastive learning.

Object detection. As one of the most important tasks in computer vision, object detection has received considerable attention and experienced significant development in the past decade. Modern object detection models can be roughly categorized into two groups with different pipelines. One follows the conventional two-stage detection pipeline, namely region proposal based methods. They first generate all possible regions of interest, then pass the region proposals to the down-stream task-specific layers for classification and bounding box regression. The region proposal based methods mainly include Faster R-CNN [28], R-FCN [3], FPN [29], SPP-net [30] and Mask R-CNN [5]. The other popular group, referred as one-stage detection models, adopts a single-step regression pipeline to map straightly from image pixels to bounding box coordinates and class probabilities. The most representative methods in this group include SSD [31], YOLO [32], FCOS [6] and RetinaNet [33]. Although these approaches perform well on pre-defined categories with sufficient training data, they are unable to

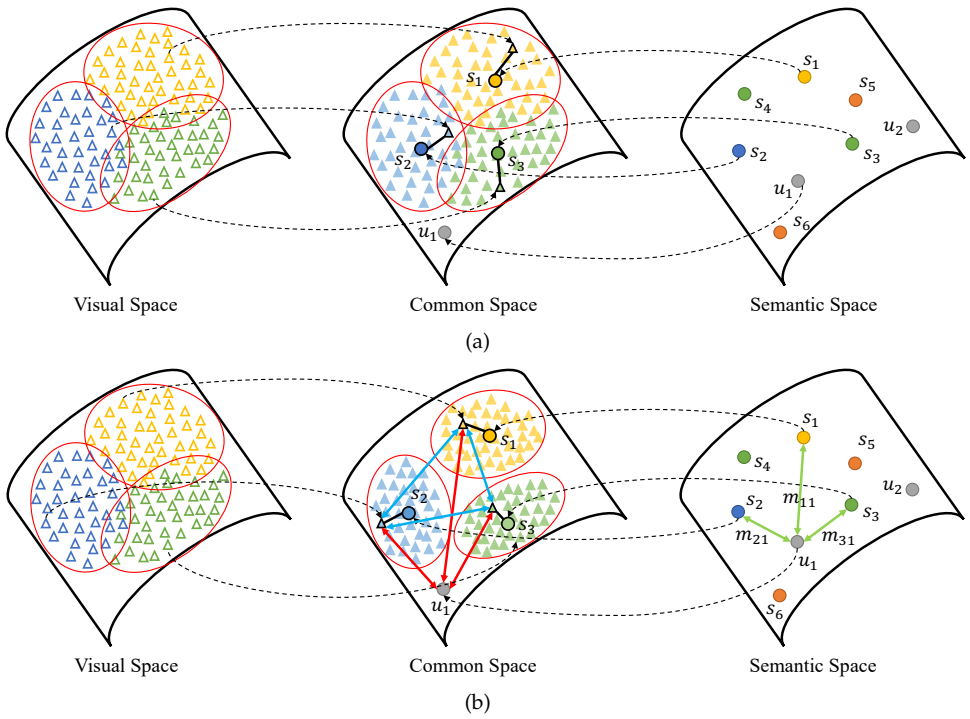


Fig. 1. Illustration of (a) the conventional embedding function based ZSL or ZSD methods that rely on node-to-node projection (black dotted arrows), where s_i and u_j refer to seen and unseen classes respectively, and (b) the proposed ContrastZSD improves the projection with different contrastive learning strategies (red and blue arrows) under the guidance of class labels (*e.g.*, s_1, s_2, s_3) and semantic relations (*e.g.*, m_{11}, m_{21}, m_{31}) for better visual-semantic alignment.

deal with the detection problem on novel concepts without training samples. In general, one stage methods with a simple single-step pipeline enjoy reduced time expense, but typically achieve lower accuracy rates than region proposal based methods. Thus, we here focus on tackling the ZSD problem with region proposal based detection models due to their high performance.

Zero-Shot Learning. The previous research literature on zero-shot learning exhibits great diversity, such as learning independent attribute classifiers [34], [35], learning embedding functions [36], [37], [38], [39] and generative adversarial networks based methods [40], [41]. In this section, we focus on the embedding based methods that are the most relevant to ours. The key idea of those methods is to learn an embedding function that maps the semantic vectors and visual features into an embedding space, where the visual features and semantic vectors can be compared directly. Compared with earlier ZSL works that learn independent attribute classifiers, the embedding function based methods show superior performance and have dominated the zero-shot learning literature. Embedding based ZSL methods differ in what embedding space is employed, which can be broadly divided into three types: learning a common embedding space for visual space and semantic space [36], [37], learning an embedding from visual space to semantic space [38], [39], [42], [43], and learning an embedding from semantic space to visual space [44]. Among those embedding strategies, the common intermediate embedding space makes it possible to adjust data structures both of semantic vectors and visual features [45]. Thus, the common intermediate space mapping strategy is adopted in our work to allow for better visual-semantic alignment.

Zero-Shot Object detection. ZSD is a recently introduced task in [15] and still remains under-explored in the computer vision literature. Only a few recent works have made some attempts on this challenging task [14], [15], [16], [17], [18]. Most of them focus on learning an embedding function from visual to semantic space. For example, Rahman *et al.* [15] developed a Faster R-CNN based semantic alignment network for ZSD underpinned by a novel semantic clustering loss to take advantage of super-class information. Considering the ambiguous nature of background class in ZSD, Bansal *et al.* [14] designed several background-aware detectors to address the confusion between unseen and background objects using external annotations. Demirel *et al.* [18] developed a hybrid region embedding model that joins a convex combination of semantic embeddings with an object detection framework. Apart from those visual-to-semantic mapping methods, there also exists some methods that learn a common intermediate space between visual space and semantic space [17] or learned an embedding from semantic to visual space [19], [20]. In addition, an alternative direction for dealing with ZSD is based on generative adversarial networks, which can convert ZSD to conventional detection problem by synthesizing features for unseen classes [23], [24], [25]. Despite their efficacy, all of these methods fail to consider the two key factors that hamper the ZSD performance, *i.e.*, the domain shift problem and visual-semantic gap. To alleviate these issues, the proposed model goes further to bring contrative learning mechanism into the realm of ZSD, allowing for further improvement of ZSD performance.

Supervised Contrastive Learning. Contrastive learning,

which can be considered as learning by comparing, has achieved significant advancement in self-supervised representation learning [46], [47], [48], [49]. Recently, a trend has emerged of leveraging contrastive learning to facilitate self-supervised computer vision tasks [50], [51], [52]. First, a number of positive/negative samples is usually created for each anchor image through data augmentation. Then, contrastive learning is performed between positive and negative pairs of images against each other, with the objective of pulling the representation of "similar" samples together and pushing that of "dissimilar" samples further away in the embedding space. However, contrastive learning used in those self-supervised algorithms fails to consider the high-level class semantics since they assigns only the augmented view for each image. For this issue, a few approaches have been proposed to leverage human-annoted labels, which has been shown to be more robust to corruption. For example, Khosla *et al.* [53] directly used class labels to define similarity, where samples from the same class are positive and samples from different classes are negative samples. Majumder *et al.* [54] devised few-shot learning with Instance discrimination based contrastive learning in a supervised setup. Inspired by the success of these methods, we first introduce contrastive learning mechanism to ZSD, and develop two contrastive learning subnets using high-level semantic information as additional supervision signals.

3 The Proposed Method

This section begins with the problem setting of the proposed model for Zero-Shot Object Detection (ZSD) (see Section 3.1). We then describe our overall framework of the proposed model (see Section 3.2), and introduce our semantics-guided contrastive learning subnets (see Section 3.3), and finally describe the training and inference details of the proposed model (see Section 3.4).

3.1 Problem Formulation

In the framework of ContrastZSD, we denote the set of all classes as $\mathcal{Y} = \mathcal{Y}^f \cup \{y_0\}$, where $\mathcal{Y}^f$ denotes the set of all foreground classes and y_0 refers to the background class. More specifically, $\mathcal{Y}^f$ can be decomposed into two disjoint subsets, *i.e.*, $\mathcal{Y}^f = \mathcal{Y}^s \cup \mathcal{Y}^u$ ($\mathcal{Y}^s \cap \mathcal{Y}^u = \varnothing$), where $\mathcal{Y}^s = \{y_1, y_2, \cdots, y_{n_s}\}$ and $\mathcal{Y}^u = \{y_{n_s+1}, y_{n_s+2}, \cdots, y_{n_s+n_u}\}$ denote the set of seen and unseen classes respectively. Given all the classes defined above, the whole label space turns to be $\mathcal{Y} = \{y_0, y_1, y_2, \cdots, y_{n_s+n_u}\}$ with the cardinality being $n_c = n_s + n_u + 1$. Inspired by previous works on ZSL, each foreground class in $\mathcal{Y}$ can be represented by a d_c-dimensional semantic embedding generated in an unsupervised manner from external linguistic sources, such as Word2Vec [55] or Glove [56]. Considering the ambiguous nature of the background class, it's unfeasible to learn a fixed class embedding from off-the-shelf linguistic sources for it. In order to reduce the confusion between the background and unseen objects, we adapt the Background Learnable RPN [57] into our ContrastZSD framework to learn a discriminative semantic vector a_0 for the background class y_0. We denote $A = [a_0, a_1, a_2, \ldots, a_{n_s+n_u}] \in \mathbb{R}^{(n_s+n_u+1)\times d_c}$ as the matrix that collects the semantic embeddings of all the categories; here, a_i refers to the label embedding of class y_i in $\mathcal{Y}$. To enable semantic relation guided contrastive learning, we further introduce a matrix $S = \{s_{ij}\}_{i,j=1}^{n_c}$ to characterize the semantic relation between different classes. The semantic relation s_{ij} between class y_i and y_j is obtained by computing the cosine similarity of their corresponding semantic word embeddings a_i and a_j, which can be formulated as,

$$s_{ij} = \text{cosine}(a_i, a_j) = \frac{a_i \cdot a_j}{\|a_i\|_2 \|a_j\|_2}, \tag{1}$$

where $\|\cdot\|_2$ stands for the ℓ_2-norm and $\cdot$ refers to the dot product operation.

In ZSD and GZSD setting, we are given an image set $\mathcal{X}$ that includes n images about $n_s + n_u$ object categories. Each image in $\mathcal{X}$ consists of several objects with boxes $\mathcal{B} = \{b_i\}_{i=1}^{r}$ and ground truth labels $\{c_i\}_{i=1}^{r}$, where r is the number of boxes and b_i is the i-th object box with c_i being the ground truth label. More specifically, $\mathcal{X}$ is composed of two subsets, *i.e.*, $\{\mathcal{X}^{tr}, \mathcal{X}^{te}\}$, where $\mathcal{X}^{tr}$ and $\mathcal{X}^{te}$ correspond to the training and testing image set respectively. The training image set $\mathcal{X}^{tr} = \{x_1, x_2, \cdots, x_{n_{tr}}\}$ collects n_{tr} labeled visual data that contain only objects from seen categories $\mathcal{Y}^s$, while the images in testing set $\mathcal{X}^{te} = \{x_{n_{tr}+1}, x_{n_{tr}+2}, \cdots, x_{n_{tr}+n_{te}}\}$ contain objects belonging to testing categories $\mathcal{Y}^{te}$. Notably, the definition of testing category set $\mathcal{Y}^{te}$ depends on the task settings, where $\mathcal{Y}^{te} = \mathcal{Y}^u$ for ZSD and $\mathcal{Y}^{te} = \mathcal{Y}^s \cup \mathcal{Y}^u$ for GZSD respectively. Conditioned on the common semantics between seen and unseen classes, our ContrastZSD model is trained on the seen object annotations of the training set $\mathcal{X}^{tr}$, with the objective of generalizing to the detection of unseen objects in $\mathcal{X}^{te}$. For each image in $\mathcal{X}^{te}$, our goal of ZSD or GZSD is to recognize all the foreground objects that belong to testing categories $\mathcal{Y}^{te}$ and simultaneously localize their bounding box coordinates in the image.

3.2 Model Architecture

The overall framework of our approach is shown in Fig. 2. It consists of two major parts marked in color. The blue part delineates the region feature encoding network, which takes raw images as input to produce region proposals and object-level visual features for each image. The visual-semantic alignment network, *i.e.*, the green part, is the key component to endowing our model with the ability of zero-shot object detection. It incorporates two different contrastive learning subnets to better align visual features and semantic descriptions. Here, we elaborate each part in detail.

3.2.1 Region Feature Encoding Network

CNN backbone. Given an arbitrary image, the CNN backbone network produces intermediate convolutional activations as image-level feature map. In our work, the basic architecture of the CNN backbone is ResNet composed of five convolutional blocks (conv1 to conv5) [58]. The output of each convolution module is fed into the top-down pathway of feature pyramid networks (FPN) [29], [59] to generate multi-scale feature maps. Taking a RGB image with dimension $\mathbb{R}^{3\times H\times W}$ as input, the output of the CNN backbone network is a tuple of feature maps with dimension $\mathbb{R}^{C\times \frac{H}{2^{i+1}}\times \frac{W}{2^{i+1}}}$ (for $i \in \{1, 2, 3, 4, 5\}$), where H

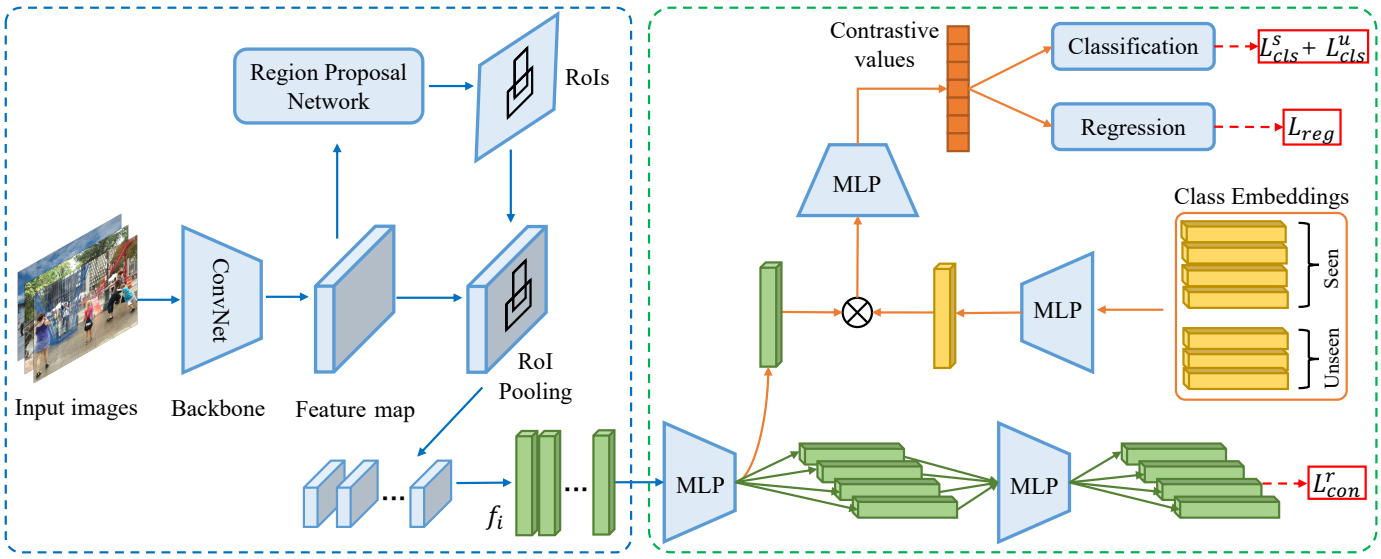


Fig. 2. The framework of the proposed ContrastZSD model. It consists of two major parts marked with two different colored boxes. Blue box: the region feature encoding network mainly composed of a CNN backbone network and a region proposal network, which takes raw images as input to produce region proposals and object-level visual features. Green box: the visual-semantic alignment network that learns the embedding function with both the region-region and region-category contrastive learning.

and W denote the height and width of the input image respectively with C being the output channel. Subsequently, the image-level feature maps will be fed into the region proposal network to generate regions of interest and object-level features.

Region Proposal Network (RPN). Taking the image's multi-scale feature maps as input, the RPN first generates k anchor boxes at each sliding window location of each feature map, where the total number of anchor boxes is $k \times \sum_{i=1}^{5} \frac{H}{2^{i+1}} \times \frac{W}{2^{i+1}}$. Then, all the anchor boxes are fed into two modules: (1) the classification module scores each proposal as either an object (positive anchor) or background (negative anchor); (2) the box regression module predicts the coordinate offsets for each region proposal. Next, it ranks the positive anchors at each sliding window location, then generates a set of candidate object proposals (RoIs) after applying the predicted offsets, denoted as $\mathcal{R} = \{r_i\}_{i=1}^{n_r}$. Since the top ranking candidate proposals can be of variable sizes, a RoI-pooling layer is further applied to project the visual features of varying-sized proposals to fixed-dimensional representations. We denote $F = [f_1, f_2, \ldots, f_{n_r}] \in \mathbb{R}^{n_r \times d_r}$ as the feature representation matrix of $\mathcal{R}$, where f_i refers to the visual feature of proposal r_i with d_r being the feature dimension. In the second part of our framework, we align these region features together with the semantic embeddings to establish a better synergy between visual and semantic domains.

3.2.2 Visual-Semantic Alignment Network

Mapping-Transfer Framework. Most existing methods on ZSD are based on a strict mapping-transfer strategy, where the mapping function is learned on seen classes then transferred directly to unseen classes. The mapping function connects the visual features and auxiliary semantic descriptions by projecting them into a joint embedding space, such that they can be compared directly. The space spanned by semantic embeddings is often chosen as the embedding space in previous works [14], [15], [17]. After projecting the extracted visual region features to semantic space, a compatibility function $\mathcal{S}(W_p^\top f_i, a_j)$ is employed to measure the matching degree between the projected proposal r_i and class y_j, where $W_p \in \mathbb{R}^{d_r \times d_c}$ is the trainable weight matrix of the projection layer. The mapping function is usually trained by standard cross entropy loss or max-margin loss to facilitate the separation between ground truth class and the rest classes [18], [26].

At the testing stage, the model trained on seen classes is directly generalized to the detection of unseen objects. Given a test sample r, the label prediction is performed by simply selecting the most matching unseen category in the embedding space, *i.e.*,

$$y^* = \arg\max_{y_k \in \mathcal{Y}^u} \mathcal{S}(W_p^\top f, a_k), \tag{2}$$

where $f \in \mathbb{R}^{d_r}$ denotes the visual feature of test sample r. The key to these methods is to learn an exact projection by tightly mapping labeled visual data to their corresponding seen classes. Considering the domain shift problem between seen and unseen categories, we argue that such a strict projection constraint has sacrificed the model's generalization ability to unseen classes. For this issue, we develop a novel mapping-contrastive framework to improve this over-simplified mapping-transfer strategy through semantics-guided contrastive learning.

Mapping-Contrastive Framework. Considering the huge gap between visual and semantic spaces, we choose the common intermediate space as the embedding space to optimize the data structures of both visual features and semantic descriptions. First, we employ two mapping functions, *i.e.*, $p_v(\cdot)$ and $p_s(\cdot)$, to embed the visual features and semantic descriptions into the joint embedding space respectively:

$$p_v(f_i) = \delta(W_v f_i + b_v), \tag{3}$$
$$p_s(a_j) = \delta(W_s a_j + b_s), \tag{4}$$

where $p_v(\cdot)$ and $p_s(\cdot)$ are implemented as Multi-Layer Perceptron (MLP) with W_v, W_s being the trainable weight matrix and b_v, b_s denoting the bias; δ refers to the nonlinear activation. Subsequently, the mapped visual features $p_v(f_i)$ and semantic descriptions $p_s(a_j)$ in the joint embedding space are fed into two semantics-guided contrastive learning subnets, *i.e.*, RCCL and RRCL, for visual-semantic alignment.

Different from the mapping-transfer framework that uses a fixed compatibility function $\mathcal{S}(\cdot)$, the RCCL subnet in our model automatically judges how well the object is consistent with a specific class through contrastive learning. To alleviate the domain shift problem, the RCCL subnet contrasts seen visual features with unseen class embeddings based on the semantic relation between seen and unseen classes. In this way, the proposed model can explicitly transfer knowledge from seen classes to unseen categories during the training phase, leading to improved generalization ability.

In addition, the RRCL subnet introduces region-region contrastive learning to regulate the visual data distribution in the joint embedding space. Under the guidance of class label information, samples belonging to the same class are pulled together in embedding space, while simultaneously pushing apart samples from different classes. As a result, our model can produce more discriminative region features with high intra-class compactness and large inter-class margin, thus reduces the visual-semantic gap. We will introduce the specific details of each contrastive learning subnet in Section 3.3.

3.3 Semantics-Guided Contrastive Learning

The key to zero-shot object detection lies in training an effective detector that is both "discriminative" enough to discriminate between seen classes and "transferable" well to unseen classes. Accordingly, in this section, we introduce two semantics-guided contrastive learning tasks to guarantee both the discriminative and transferable ability of the proposed ContrastZSD model.

3.3.1 Region-Category Contrastive Learning

Visual-Semantic Consistency. At training time, only the visual features from seen categories are provided, while the semantic embeddings corresponding to both seen and unseen classes are available to access. In order to enable an explicit knowledge transfer, we propose to contrast the visual features of seen objects with both seen and unseen classes to distinguish whether they are consistent or not. Recall that we have encoded the visual features of seen region proposals and the class embeddings into the common intermediate space in Section 3.2. For each region-category pair (f_i, a_j) encoded as $\langle p_v(f_i), p_s(a_j)\rangle$, we first fuse their information and then distinguish how consistent the fusion is, which can be formulated as

$$o(f_i, a_j) = \sigma(g(p_v(f_i) \otimes p_s(a_j))), \tag{5}$$

where $\otimes$ refers to the element-wise product operation for visual-semantic information fusion; $o(f_i, a_j)$ denotes the consistency value between $p_v(f_i)$ and $p_s(a_i)$; $g(\cdot)$ is a projection head implemented as MLP network with σ being the Sigmoid thresholding.

Consistency Based Classification Branch. Given n_r region proposals, we first operate Eq. (5) over all the classes to predict the consistency scores in a matrix form, denoted as $O \in \mathbb{R}^{n_r \times (n_s+n_u+1)}$. The relation matrix between seen and unseen categories, *i.e.*, $\hat{S} = \{s_{ij} | i \in [0, n_s], j \in [n_s + 1, n_s + n_u]\} \in \mathbb{R}^{(n_s+1)\times n_u}$, is employed to characterize the unseen similarity distribution, where each row of $\hat{S}$ is then normalized by applying softmax function. Subsequently, we utilize not only the ground truth label but also the unseen similarity distribution $\hat{S}$ as supervision signals, and derive the full classification loss as

$$L_{cls} = L^s_{cls} + \lambda L^u_{cls}, \tag{6}$$

where L^s_{cls} and L^u_{cls} refer to the classification loss with respect to seen and unseen classes respectively; λ is a trade-off parameter. More specifically, L^s_{cls} is designed to endow the model with the discriminative ability to distinguish different seen classes, which can be formulated by the binary cross-entropy loss,

$$L^s_{cls} = -\sum_{i=1}^{n_r}\sum_{j=0}^{n_s} v_{ij}\log(o_{ij}) + (1 - v_{ij})\log(1 - o_{ij}), \tag{7}$$

where v_{ij} is a binary class indicator, $v_{ij} = 1$ if j equals to the ground truth label index c_i and $v_{ij} = 0$ otherwise; o_{ij} refers to the element at the i-th row and j-th column of O. To enable explicit knowledge transfer from seen to unseen classes, we take advantage of the unseen similarity distribution in $\hat{S}$ as additional supervision signals. Thus, L^u_{cls}, the second item in Eq. (6), turns to be

$$L^u_{cls} = -\sum_{i=1}^{n_r}\sum_{j=n_s+1}^{n_s+n_u} s_{c_ij}\log(o_{ij}) + (1 - s_{c_ij})\log(1 - o_{ij}).$$

By minimizing L^u_{cls}, the predicted unseen class probability is enforced to be consistent with the true unseen similarity distribution, thus improves the model's generalization capability to the unseen domain without disturbing the seen object detection optimized by L^s_{cls}.

Consistency Based Regression Branch. Unlike the image classification task containing only classification results, object detection also predicts object locations, which is performed by adding suitable offsets to the generated regions in order to align them with the ground truth coordinates. Given a predicted bounding box coordinate $(x^o_i, y^o_i, w^o_i, h^o_i)$ (center coordinate, width, height) and its corresponding ground truth box coordinates $(x^t_i, y^t_i, w^t_i, h^t_i)$, the regressor is configured to learn scale-invariant transformation between two centers and log-scale transformation between widths and heights. Thus, the ground truth offsets $b^*_i = (t^*_{ix}, t^*_{iy}, t^*_{iw}, t^*_{ih})$ can be derived as follows:

$$t^*_{ix} = \frac{|x^t_i - x^o_i|}{x^o_i},\ t^*_{iy} = \frac{|y^t_i - y^o_i|}{y^o_i},\ t^*_{iw} = \log\frac{w^t_i}{w^o_i},\ t^*_{ih} = \log\frac{h^t_i}{h^o_i}.$$

The standard regression branch in Faster R-CNN predicts the offsets of each region proposal based solely on its visual characteristics. In order to better adapt this branch to the ZSD task, we further take advantage of the semantic information in the form of consistency values. For each region proposal, we concatenate the visual feature $f_i \in \mathbb{R}^{d_r}$ with the consistency score vector $o_i \in \mathbb{R}^{n_c}$ as the input of

the box regression layer to predict the coordinate offsets as $b_i = (t_{ix}, t_{iy}, t_{iw}, t_{ih})$. Subsequently, we minimize the regression loss for all the n_r region proposals, *i.e.*,

$$L_{reg} = \sum_{i=1}^{n_r} \sum_{j \in \{x,y,w,h\}} \text{smooth}_{\ell_1}(t_{ij} - t_{ij}^*), \tag{8}$$

where $\text{smooth}_{\ell_1}(\cdot)$ denotes the same smooth ℓ_1 loss used in Faster R-CNN that tweaks the predicted region coordinates to the corresponding target bounding box.

3.3.2 Region-Region Contrastive Learning

The key to region-category contrastive learning lies in that the embedded semantic vector of one class should try to be consistent with every visual instance features of the same class. However, the distribution of instances in the visual space tends to be indistinctive, which can inevitably decrease the model's discriminative ability. The case may be even worse for the object detection task since the top ranking proposals may only cover parts of objects instead of whole objects.

In order to optimize the visual data structure in the common space, we propose to contrast between different region proposals based on the semantics information. Given n_r region proposals generated from the same batch of images, we first map their features $\{p_v(f_i)\}_{i=1}^{n_r}$ to new representations $\{z_i\}_{i=1}^{n_r}$ with a projection network $h_v(\cdot)$:

$$z_i = h_v(p_v(f_i)) = \theta(W_{h_v} p_v(f_i) + b_{h_v}), \tag{9}$$

where $h_v(\cdot)$ is a MLP with weight matrix W_{h_v} and bias b_{h_v}; θ is nonlinear activation. Unlike the conventional self-supervised contrastive learning that focuses only on instance discrimination, we aim to achieve class discrimination by effectively leveraging the label information.

For each region proposal r, we treat the proposals from the same class with r as positive samples, and all the other proposals generated from the same batch of images as negative samples. Taking the i-th region proposal encoded as z_i as an example, we assume that there are p_i positive proposals $\{z_1^+, z_2^+, \cdots, z_{p_i}^+\}$ and n_i negative samples $\{z_1^-, z_2^-, \cdots, z_{n_i}^-\}$. Each positive sample z^+ shares the same label with z_i, while the class label of z^- is different from z_i. The region-region contrastive loss used for a pair of bounding boxes takes the following form,

$$\ell_{con}^r(z_i, z_j^+) = -\log \frac{\exp(\frac{z_i \cdot z_j^+}{\tau})}{\sum_{k=1}^{p_i} \exp(\frac{z_i \cdot z_k^+}{\tau}) + \sum_{k=1}^{n_i} \exp(\frac{z_i \cdot z_k^-}{\tau})}, \tag{10}$$

where τ is the temperature parameter set as 0.1 by default as in [53]. Thus, the total contrastive loss L_{con}^r for n_r region proposals can be formulated as

$$L_{con}^r = \frac{1}{n_r \times n_p} \sum_{i=1}^{n_r} \sum_{j=1}^{p_i} \ell_{con}^r(z_i, z_j^+), \tag{11}$$

Benefiting from this constraint, the region features in the same class is pulled closer, while the instances from different classes are pushed farther apart, resulting in a more distinguishable visual data structure.

3.4 Training and Inference Details

Training. Unlike previous works on ZSD that usually rely on multi-step training, we adopt an end-to-end training mechanism to jointly optimize the network parameters. We keep the bottom layers fixed to the weights pre-trained on ImageNet [60], then train the region proposal network (RPN) and semantics-guided contrastive learning network. More specifically, the RPN is trained with the same classification and regression loss as in Faster R-CNN. Notably, the RPN, which is trained on seen visual data without the exploitation of any semantic information, can generate proposals for unseen objects also, since it is designed to generate object proposal based on the objectness measure. To optimize the semantics-guided contrstive network, we minimize a multi-task loss designed specifically for ZSD, including both the classification, bounding box regression and contrastive losses. The overall loss for all the region proposals takes the following form:

$$L = L_{cls}^s + \lambda L_{cls}^u + L_{reg} + \beta L_{con}^r, \tag{12}$$

where λ and β are two hyper-parameters that control the trade-off between the loss terms in Eq. (12).

Inference. Given a test image I_{te}, we first forward I_{te} through the trained ContrastZSD network to get all the candidate proposals. Each proposal is associated with not only its bounding box coordinates but also the contrastive values with respect to all the testing classes. For each testing class y, we compare the contrastive values of each region proposal on this class with a pre-defined threshold, then collect all the region proposals with a contrastive value above the threshold as the preliminary detection results. Next, the Non-Maximum Suppression (NMS) is applied to remove the proposals with small IoU values and get the final detection results. Notably, if there are more than N_m bboxes after NMS, we will rank the detected results based on their confidence scores, and then only keep the top N_m ones, where N_m specifies the maximum number of objects detected in a single image.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate the proposed ContrastZSD model on two widely-used datasets for object detection, *i.e.*, PASCAL VOC 2007+2012 [61] and MS COCO 2014 [62]. PASCAL VOC consists of 20 common object categories for object class recognition. More specifically, PASCAL VOC 2007 contains 2501 training images, 2510 validation images and 5011 test images. PASCAL VOC 2012 was released without test images provided, and includes 5717 training images and 5823 validation images. MS COCO was designed for object detection and semantic segmentation tasks. It contains 82783 training and 40504 validation images from 80 categories. Being zero-shot, each dataset should be split into the combination of seen/unseen subsets. For the purpose of fair comparison, we follow previous works that also target on the ZSD task to split the datasets. For the PASCAL VOC dataset, we adopt the same setting in [18] to split the 20 categories, where 4 classes are selected as unseen and the remaining 16 are seen classes. In terms of the MS

COCO dataset, we follow the same procedures described in [14] to divide the dataset into two different splits: (1) 48 seen and 17 unseen classes; (2) 65 seen and 15 unseen classes. Conditioned on the above seen/unseen class splits, we follow the steps in [63] to create the train and test set for each dataset.

Implementation Details. As for the class semantic embeddings, we use the ℓ_2 normalized 300-dim Word2Vec for MS COCO classes, which is produced by a model trained on a Wikipedia corpus in an unsupervised manner. For PASCAL VOC classes, we use the average of 64-dim binary per-instance attribute annotation of all training images from aPY dataset [35]. The image scale in MS COCO and PASCAL VOC is resized to (1333, 800) and (1000, 600) for the longer and shorter edge, while keeping the original image aspect ratio. We perform horizontal flip for augmenting the training data. The number of region proposals generated for each image is 128 and 300 during training and testing respectively. Non-Maximum Suppression (NMS) with an IoU threshold of 0.7 is employed to remove redundant bounding boxes. The maximum number of objects detected in a single image N_m is 100. We adopt ResNet-50 [58] pretrained on ImageNet [60] as the CNN backbone with feature pyramid network (FPN) [29]. The mapping functions p_v and p_s are implemented as two fully-connected layers, taking 1024-dim region features and d_c-dim semantic embeddings as input respectively, then transform them to the same dimension as the common space (1024-dim in our case). In terms of the semantics-guided contrastive learning network, we implement the MLP networks in RCCL and RRCL as stacked linear layers with output size of [1024, 512, 256, 1] and [1024, 512, 128] respectively. Except for the last layer of the MLP network in RCCL that uses a Sigmoid activation, all the other linear layers are implemented with ReLU activation. We employ SGD with momentum of 0.9 and learning rate of 10^{-5} to optimize the proposed model.

Comparison Methods. To demonstrate the effectiveness of the proposed method, we compare it with both baseline method and state-of-the-art approaches developed for the ZSD task. We provide a brief description of the comparison methods as follows. **ConSE** is the baseline method that adapts the standard Faster R-CNN model trained without any semantic information to the ZSD task by employing ConSE [64] at the testing stage. **SAN** [15] is the first deep network developed for the ZSD task that jointly models the interplay between visual and semantic domain information. **HRE** [18] is a YOLO [32] based end-to-end zero-shot detector that learns a direct mapping from region pixels to the space of class embeddings. **SB and DSES** are background-aware zero-shot detectors proposed in [14] that differentiate background regions based on a large open vocabulary. **TD** [17] learns both visual-unit-level and word-level attention to tackle the ZSD task with textual descriptions instead of a single word. **PL** [63] designs a novel polarity loss for RetinaNet based ZSD framework to better align visual and semantic concepts. **BLC** [57] combines Cascade Semantic R-CNN, semantic information flow and background learnable RPN into a unified framework for the ZSD task.

Evaluation Metrics. We adopt the evaluation protocols used in [14], [15], including Recall@100 and mAP, to evaluate the performance of our model, where a larger recall or mAP value indicates better performance. More specifically, Recall@100 is defined as the recall with only the top 100 detections selected from an image, while mAP indicates the mean average precision of the detection results for all the categories. For mAP, we first calculate the per-class average precision (AP) for each individual class to study category-wise performance, then take the mean (mAP) as a measure of overall performance. More specifically, the widely adopted 11-point interpolation approach [61] is used to compute AP, which is defined as the average precision of eleven equally spaced recall levels [0, 0.1, 0.2, ..., 1]. For ZSD, the testing phase only involves samples from unseen categories, and thus the performance is measured over the set of unseen classes $\mathcal{Y}^u$. While for GZSD, we take advantage of samples from both seen categories $\mathcal{Y}^s$ and unseen categories $\mathcal{Y}^u$ to test the model performance. The harmonic mean performance on seen and unseen classes is computed to reflect the overall performance for GZSD.

TABLE 1
ZSD and GZSD mAP(%) at IoU threshold 0.5 on PASCAL VOC dataset, where "S" and "U" refer to the average performance on seen and unseen classes with "HM" denoting their harmonic mean.

Model	Seen	ZSD	GZSD		
			S	U	HM
ConSE	**77.0**	52.1	59.3	22.3	32.4
SAN	69.6	59.1	48.0	37.0	41.8
HRE	65.6	54.2	62.4	25.5	36.2
PL	63.5	62.1	-	-	-
BLC	75.1	55.2	58.2	22.9	32.9
ContrastZSD	76.7	**65.7**	**64.1**	**48.3**	**55.1**

4.2 Quantitative Results

4.2.1 PASCAL VOC

ZSD and GZSD Performance. We present the mAP performance in Table 1 to compare different methods over the PASCAL VOC dataset. Based on the settings in [18], the performance of each method is reported in three different testing configurations, *i.e.*, "Seen", "ZSD" and "GZSD", where "Seen" refers to the conventional object detection task used to detect objects from $\mathcal{Y}^s$. We can observe from Table 1 that our method outperforms all the comparison methods under the "ZSD" setting, increasing the mAP from 62.1% achieved by the second-best method PL to 65.7%, which indicates the good transferable ability of the proposed ContrastZSD model to unseen classes. In addition to the state-of-the-art ZSD performance, it's interesting to see that our model also performs very well on the common "Seen" object detection task, even comparable to the ConSE baseline that only focuses on training an excellent Faster R-CNN model over seen classes. We can attribute this to the region-region contrastive learning subnet in our model that optimizes the visual structure for better discriminating different seen classes. Despite the effectiveness of ConSE on seen object detection, it achieves the worst performance on ZSD task due to the lack of semantic information in the training phase. In contrast to "Seen" and "ZSD", "GZSD" is a more challenging and realistic task where both seen and unseen classes are present at inference. As depicted in Table

TABLE 2
Class-wise AP and mAP (%) on the PASCAL VOC dataset at IoU threshold 0.5, where mAP_s and mAP_u refer to the mAP values with respect to seen and unseen classes respectively.

Methods	*aeroplane*	*bicycle*	*bird*	*boat*	*bottle*	*bus*	*cat*	*chair*	*cow*	*d. table*	*horse*	*motorbike*	*person*	*p. plant*	*sheep*	*tvmonitor*	mAP_s	*car*	*dog*	*sofa*	*train*	mAP_u
	Seen Classes																	Unseen Classes				
ConSE	**82.2**	**85.8**	83.2	66.7	70.0	77.5	87.4	**60.1**	**80.0**	69.4	84.5	**85.0**	84.6	**56.6**	**81.4**	78.1	**77.0**	49.0	75.0	53.0	31.3	52.1
SAN	71.4	78.5	74.9	61.4	48.2	76.0	**89.1**	51.1	78.4	61.6	84.2	76.8	76.9	42.5	71.0	71.7	69.6	56.2	85.3	62.6	26.4	57.6
HRE	70.0	73.0	76.0	54.0	42.0	**86.0**	64.0	40.0	54.0	**75.0**	80.0	80.0	75.0	34.0	69.0	**79.0**	65.6	55.0	82.0	55.0	26.0	54.2
PL	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	63.5	63.7	**87.2**	53.2	44.1	62.1
BLC	78.5	83.2	77.6	**67.7**	70.1	75.6	87.4	55.9	77.5	71.2	**85.2**	82.8	77.6	56.1	77.1	78.5	75.1	43.7	86.0	60.8	30.1	55.2
ContrastZSD	81.9	85.6	**85.0**	66.6	**70.8**	77.0	88.9	58.4	79.5	66.8	84.7	82.2	**84.9**	55.4	81.1	78.4	76.7	**65.5**	86.4	**63.1**	**47.9**	**65.7**

1, for each comparison method, the unseen object detection performance of GZSD drops significantly compared with the corresponding ZSD results. One possible reason for this performance degradation is that those methods can easily overfit the seen classes, such that most unseen objects are recognized as seen classes. Compared with those methods, our model shows more promising results on unseen object detection for GZSD, *i.e.*, 48.3% *vs* 37.0%, while not disturbing the seen object detection performance, *i.e.*, 64.1% *vs* 62.4%. As a result, our method enjoys a more balanced performance on seen and unseen classes for GZSD.

Class-wise Performance. To study the per-category results, we present the class-wise mAP performance on PASCAL VOC in Table 2. The results on seen and unseen classes are evaluated independently in "Seen" and "ZSD" setting for fair comparison with other methods. Not surprisingly, ConSE shows more promising results on seen classes than other methods. As shown in Table 2, ConSE achieves the best performance on 7 out of 16 seen categories, *e.g.*, "aeroplane", "chair" and "cow". As for the class-wise ZSD results, our method outperforms the competitors on three of the four unseen classes by a large margin, which further verifies the superiority of our model for the ZSD task. Compared with other methods, the performance gain is more pronounced for "car" and "train" classes. We think this is because the car and train objects are visually similar, which makes the system hard to distinguish. Benefiting from the region-region contrastive learning strategy, our model can learn more discriminative visual features for better distinguishing objects belonging to the two categories.

TABLE 3
ZSD performance in terms of Recall@100(%) and mAP(%) with different IoU thresholds on MS COCO dataset.

Model	Split	Recall@100			mAP
		IoU=0.4	IoU=0.5	IoU=0.6	IoU=0.5
ConSE	48/17	28.0	19.6	8.7	3.2
SB	48/17	34.5	22.1	11.3	0.3
DSES	48/17	40.2	27.2	13.6	0.5
TD	48/17	45.5	34.3	18.1	-
PL	48/17	-	43.5	-	10.1
BLC	48/17	51.3	48.8	45.0	10.6
ContrastZSD	48/17	**56.1**	**52.4**	**47.2**	**12.5**
ConSE	65/15	30.4	23.5	10.1	3.9
PL	65/15	-	37.7	-	12.4
BLC	65/15	57.2	54.7	51.2	14.7
ContrastZSD	65/15	**62.3**	**59.5**	**55.1**	**18.6**

TABLE 4
GZSD performance in terms of Recall@100 (%) and mAP (%) achieved with IoU=0.5 over each seen/unseen split of MS COCO.

Method	Split	Recall@100			mAP		
		S	U	HM	S	U	HM
ConSE	48/17	43.8	12.3	19.2	37.2	1.2	2.3
	65/15	41.0	15.6	22.6	35.8	3.5	6.4
PL	48/17	38.2	26.3	31.2	35.9	4.1	7.4
	65/15	36.4	37.2	36.8	34.1	12.4	18.2
BLC	48/17	57.6	46.4	51.4	42.1	4.5	8.2
	65/15	56.4	51.7	53.9	36.0	13.1	19.2
ContrastZSD	48/17	**65.7**	**52.4**	**58.3**	**45.1**	**6.3**	**11.1**
	65/15	**62.9**	**58.6**	**60.7**	**40.2**	**16.5**	**23.4**

4.2.2 MS COCO

ZSD Performance. For the MS COCO dataset, we follow the experimental settings in [14] and [17] to evaluate the ZSD performance with different Intersection over Union (IoU) thresholds, *i.e.*, 0.4, 0.5 and 0.6, where IoU is used to measure the overlap between the predicted and ground truth bounding boxes. The experimental results in terms of both Recall@100 and mAP are presented in Table 3. For the 48/17 split, we compare our model with ConSE, SB, DSES, TD, PL and BLC. From the ZSD results in Table 3, we can observe that our proposed method achieves a significant gain on both metrics (mAP and Recall@100). Compared with the second-best method BLC, the proposed model gains an absolute improvement of 1.9% in mAP and 3.6% in Recall@100 at IoU threshold 0.5. On the 65/15 split, we compare our model only with ConSE, PL and BLC, since other methods didn't report their results on this split. As shown in Table 3, the proposed model outperforms all the comparison methods by a large margin, which improves the mAP and Recall@100 achieved by the second-best method BLC from 14.7% and 54.7% to 18.6% and 59.5% at IoU threshold 0.5. Moreover, compared with the 48/17 split, the performance gain is more pronounced on the 65/15 split. We think this is because the 65/15 split has a larger proportion of seen categories than the 48/17 split, which enables our model to learn more knowledge about the unseen classes from similar seen classes.

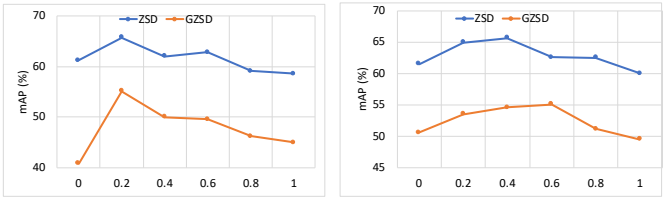
TABLE 5
Class-wise Recall@100 for the 48/17 and 65/15 split of MS-COCO with the IoU threshold being 0.5.

48/17 split	*bus*	*dog*	*cow*	*elephant*	*umbrella*	*tie*	*skateboard*	*cup*	*knife*	*cake*	*couch*	*keyboard*	*sink*	*scissors*	*airplane*	*cat*	*snowboard*	mean(%)
BLC	77.4	88.4	71.9	77.2	0.0	0.0	41.7	38.0	**45.6**	34.3	65.2	23.8	14.1	20.8	48.3	79.9	61.8	46.4
ContrastZSD	**82.8**	**92.1**	**76.9**	**82.0**	**2.3**	**1.1**	**45.0**	**51.7**	41.7	**44.2**	**74.2**	**33.7**	**21.0**	**32.3**	**55.6**	**83.8**	**69.5**	**52.4**

65/15 split	*airplane*	*train*	*parking meter*	*cat*	*bear*	*suitcase*	*frisbee*	*snowboard*	*fork*	*sandwich*	*hot dog*	*toilet*	*mouse*	*toaster*	*hair drier*	mean(%)
BLC	58.7	72.0	10.2	96.1	91.6	46.9	44.1	65.4	37.9	82.5	**73.6**	43.8	7.9	35.9	2.7	51.3
ContrastZSD	**67.7**	**77.5**	**17.3**	**97.4**	**94.6**	**56.6**	**57.2**	**72.0**	**43.7**	**85.0**	**73.6**	**67.7**	**17.6**	**47.4**	**4.1**	**58.6**

TABLE 6
The effect of each key component for ZSD and GZSD performance in terms of mAP at IoU threshold 0.5 over PASCAL VOC dataset.

Variants	RCCL_u	RRCL	ZSD	GZSD		
				S	U	HM
w/o. RRCL	√		61.5	59.3	44.2	50.6
w/o. RCCL_u		√	61.2	61.0	30.6	40.8
ContrastZSD	√	√	**65.7**	**64.1**	**48.3**	**55.1**



(a) Varying λ (b) Varying β

Fig. 3. Hyper-parameter sensitivity of the proposed ContrastZSD model on the PASCAL VOC dataset.

GZSD Performance. In Table 4, we further present the GZSD results achieved by ConSE, PL, BLC and the proposed ContrastZSD. The results demonstrate that our model exceeds the three comparison methods in terms of both mAP and Recall@100. As shown in Table 4, the proposed ContrastZSD outperforms the second-best method BLC by a large margin, where the absolute HM performance gain is 6.9% Recall@100 and 2.9% mAP for the 48/17 split and 6.8% Recall@100 and 4.2% mAP for the 65/15 split. Due to the lack of semantic information during model training, the performance of ConSE is far worse than the other methods on both of mAP and Recall@100 metrics. Based on the direct mapping-transfer strategy, PL achieves much higher recall and mAP on seen classes than unseen classes, leading to a low harmonic mean (HM) performance. This is because the mapping-transfer strategy is prone to over-fitting the seen classes, such that very little knowledge is learned for unseen classes. Furthermore, the performance gain of our model is more remarkable on GZSD than ZSD, as shown in Table 3 and 4. We can attribute this improvement to the explicit knowledge transfer from seen objects to unseen classes, which prevents our model from over-fitting the seen categories on the GZSD task.

Class-wise Performance. The class-wise performance on unseen classes of the two splits is reported in Table 5 under the GZSD setting. Compared with the state-of-the-art BLC method, our model achieves higher Recall@100 on 16 out of 17 unseen classes on 48/17 split and all the 15 unseen classes on 65/15 split. This phenomenon suggests that our model can improve the GZSD performance evenly, instead of only focusing on certain categories. We have also noted that the BLC method fails to detect any objects for the "umbrella" and "tie" class, resulting in a recall rate of 0. One possible reason is that those classes have fewer semantically similar concepts in the seen category set, which greatly increases the difficulty of implicit knowledge transfer in BLC. Benefiting from the region-category contrastive learning mechanism, explicit knowledge transfer is performed from seen classes to unseen classes in our model. As a result, our model successfully surpasses BLC over those unseen classes without close counterparts among the seen classes, *e.g.*, "umbrella", "tie" and "hair drier" class.

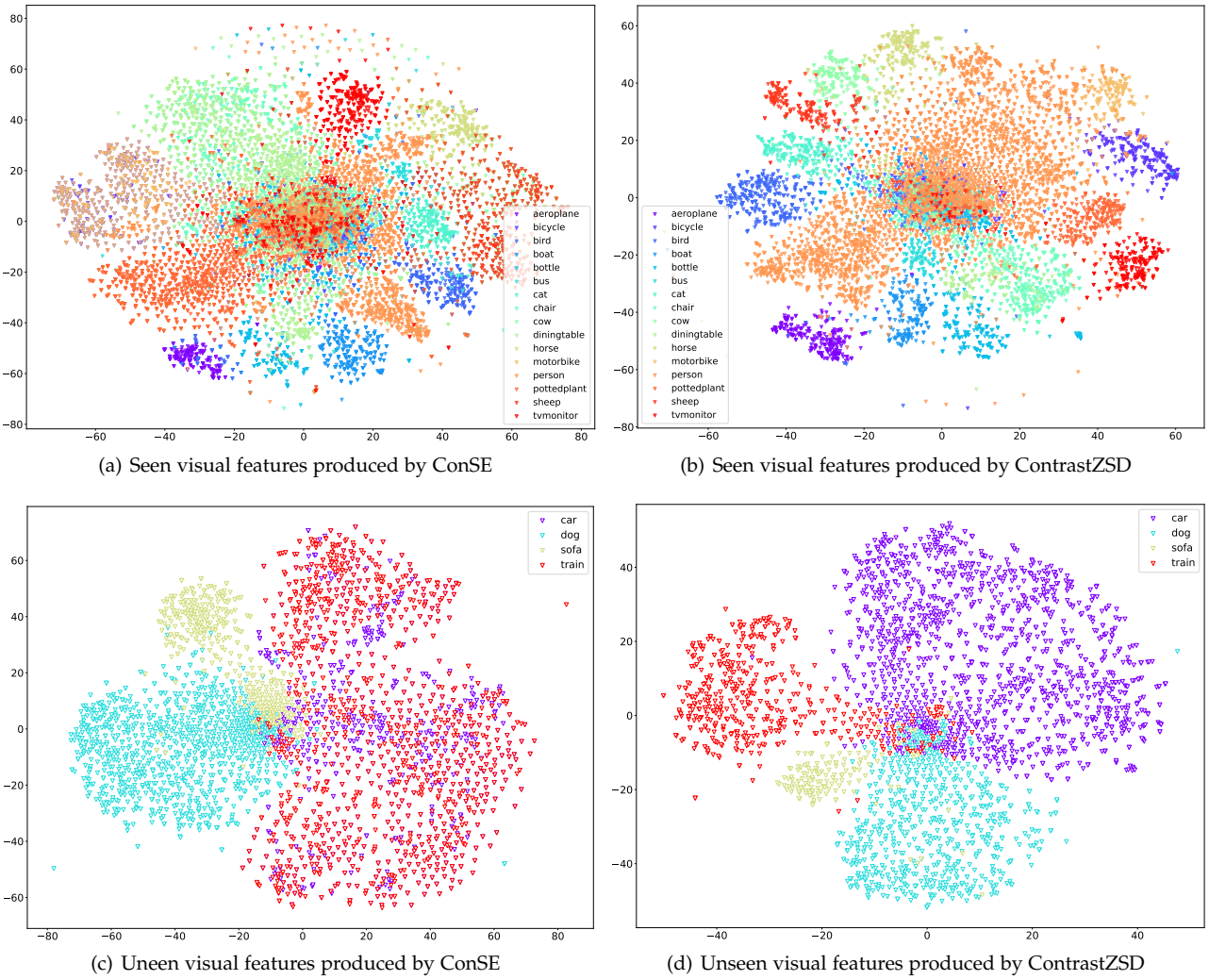
4.3 Ablation Studies

In this section, we present further analysis of the proposed method, including the ablation study for each contrastive learning subnet and the sensitivity of our model to the hyper-parameters.

4.3.1 Ablation for Contrastive Learning Subnets

We conduct extensive quantitative analysis for the key components, *i.e.*, RRCL and RCCL, in the proposed model by leaving one component out of our framework at a time. In table 6, we present the ZSD and GZSD performance in terms of mAP on the PASCAL VOC dataset to compare the effects of different contrastive learning subnets. The results of "ContrastZSD" are obtained by simultaneously considering all the components, leading to the best ZSD and GZSD performance.

Effectiveness of RCCL_u. The method "w/o. RCCL_u" removes the unseen class contrastive learning process in the RCCL subnet to contrast seen objects with only seen categories, thus the explicit knowledge transfer from seen to unseen classes cannot be conducted. As a result, both the ZSD and GZSD mAP suffer from a degradation compared with ContrastZSD. While there is only a small decrease on



(a) Seen visual features produced by ConSE (b) Seen visual features produced by ContrastZSD

(c) Uneen visual features produced by ConSE (d) Unseen visual features produced by ContrastZSD

Fig. 4. T-SNE plot of the visual feature distribution on PASCAL VOC, where the points from different categories are marked in different colors. For better illustration, we show the visual features of seen and unseen classes in (a) (b) and (c) (d) respectively.

the ZSD performance, the mAP on unseen classes of GZSD drops significantly from 48.3% to 30.6%, leading to a low harmonic mean performance. This phenomenon indicates the explicit knowledge transfer plays a more important role in GZSD than ZSD, since it can prevent the model from over-fitting the seen classes.

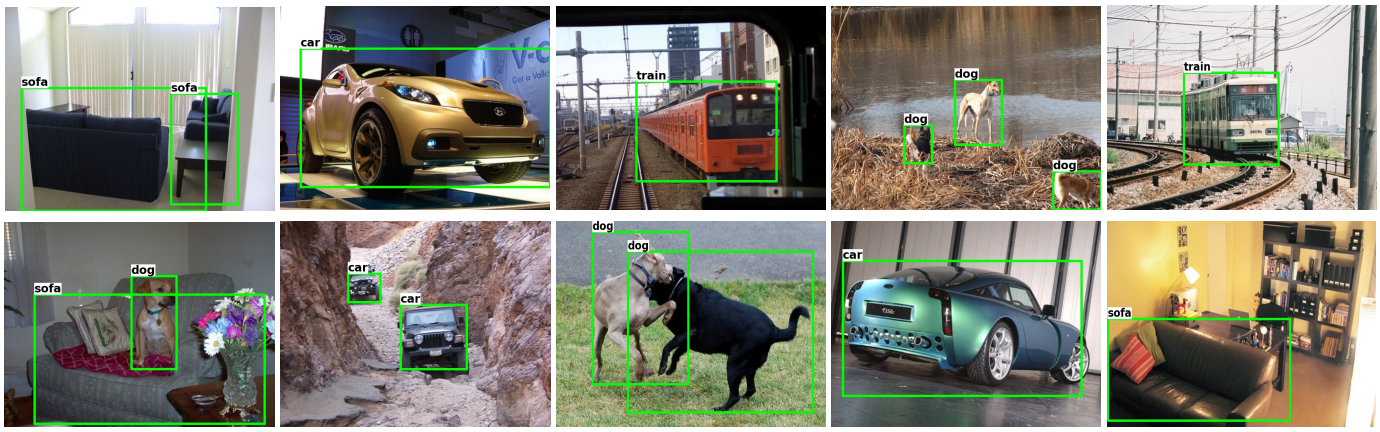
Effectiveness of RRCL. "w/o. RRCL" denotes the variant method that removes the RRCL subnet, such that the indistinctive visual data distribution cannot be optimized based on class label information. Compare with ContrastZSD, the mAP performance of both ZSD and GZSD experiences a decline, *i.e.*, 61.5% *vs* 65.7% on ZSD and 50.6% *vs* 55.1% on GZSD. This is because the original visual space is lack of discriminative ability and thus is suboptimal for ZSD and GZSD. Benefiting the RRCL subnet, our model can optimize the visual data structure, including both the seen and unseen distribution, to be more distinguishable.

4.3.2 Sensitivity Analysis

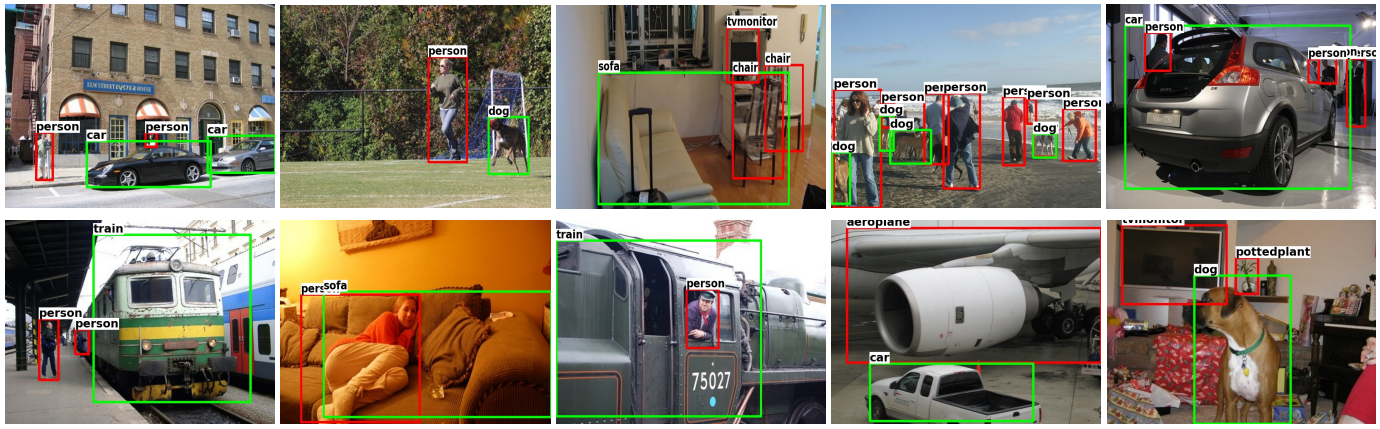
In order to investigate the importance of each key component, we further analyze the effect of hyper-parameters to our model by varying λ and β in the range of $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$. The ZSD and GZSD performance in terms of mAP achieved with varying parameters on PASCAL VOC are demonstrated in Fig. 3.

Sensitivity Analysis for λ. We first discuss the impact of parameter λ on the performance of the proposed ContrastZSD. As shown in Fig. 3(a), when the value of λ increases from 0, the performance of our model gains a notable improvement. This indicates that the explicit knowledge transfer in RCCL can indeed enable the model to learn more knowledge about the unseen domain. Notably, choosing λ around 0.2 tends to yield the best ZSD and GZSD performance. If we keep increases the value of λ, both of the ZSD and GZSD performance begin to decrease. Thus, we set λ to 0.2 in the other experiments.

Sensitivity Analysis for β. Then we discuss the impact of the parameter β on our model that controls the contribution of the RRCL subnet. As shown in Fig. 3(b), the best choice of β is 0.4 for ZSD and 0.6 for GZSD respectively over the PASCAL VOC dataset. Larger or smaller values of parameter β tend to degrade the detection performance. It proves that the visual structure constraint in RRCL subnet can effectively optimize the visual data distribution to be more distinguishable with proper β, allowing for better visual-semantic alignment. Taking both ZSD and GZSD into



(a) ZSD results on PASCAL VOC



(b) GZSD results on PASCAL VOC

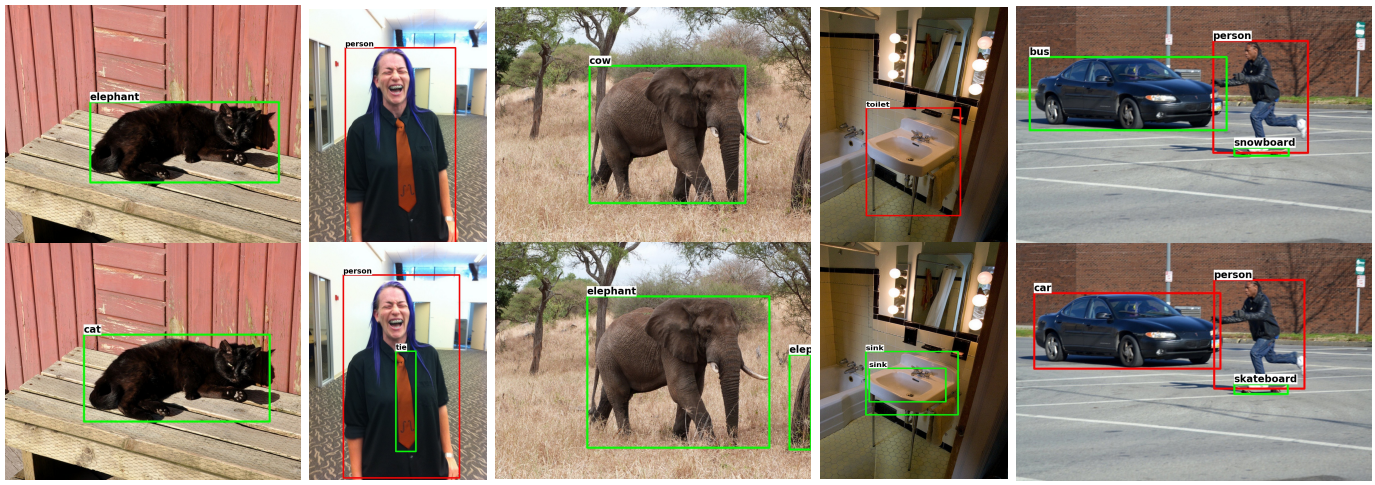
Fig. 5. Some ZSD and GZSD detection results on the PASCAL VOC dataset. The region proposals of seen and unseen categories are marked as red and green boxes respectively.

consideration, we set β to 0.5 in our experiments.

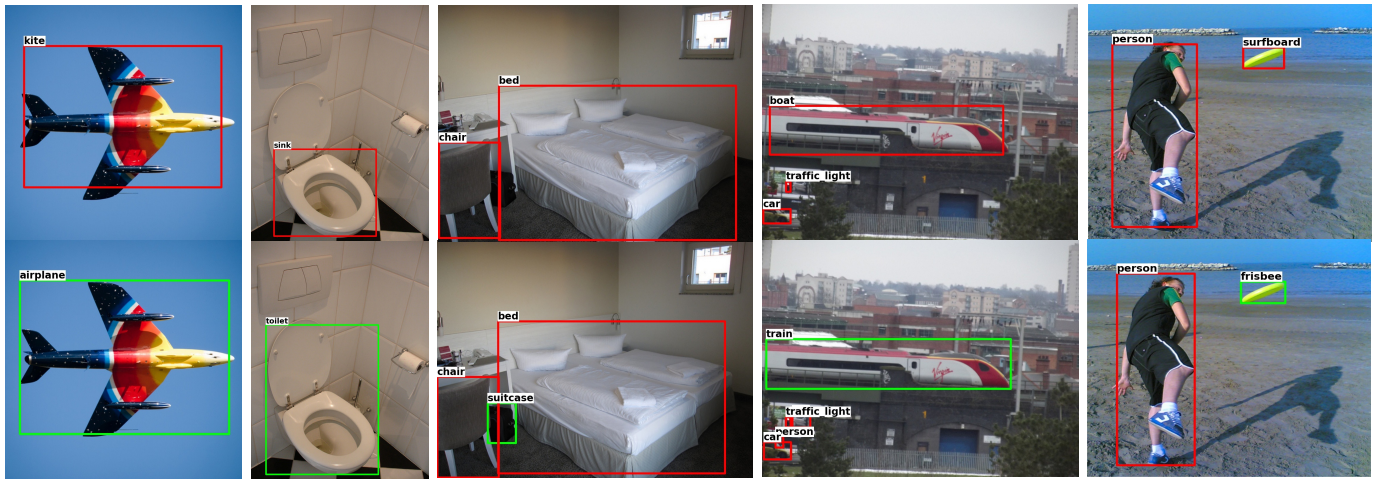
4.4 Qualitative Analysis

Visual Structure Optimization. To study the effectiveness of our model in visual structure optimizing, we utilize t-SNE [65] to visualize the visual features of detected region proposals on the PASCAL VOC dataset. The visual feature distribution corresponding to seen and unseen categories produced by the baseline method ConSE are illustrated in Fig. 4(a) and 4(c) respectively, where most clusters of the categories fail to have a clear frontier. For example, the intra-class distance of the "horse" and "sheep" objects in Fig. 4(a) is sometimes even larger than their inter-class distance, while the "car" and "train" class objects in Fig. 4(c) suffer from an extremely large overlap. In such scenarios, the objects from different classes are extremely hard to be distinguished, thereby significantly inhibiting the learning of embedding functions. By contrast, it can be clearly observed from Fig. 4(b) and 4(d) that the visual features learned by our model demonstrate higher intra-class compactness, as well as a much larger inter-class margin on both the seen and unseen categories of PASCAL VOC, exhibiting more obvious clustering patterns. This suggests that our model is able to produce more discriminative visual features to enable better visual-semantic alignment, which further substantiates the above-mentioned quantitative improvements on the selected datasets.

Detection Results. For qualitative analysis of the detection performance, we present some ZSD and GZSD results on PASCAL VOC and MS COCO dataset in Fig. 5 and Fig. 6 respectively. From the ZSD results on PASCAL VOC shown in Fig. 5(a), we can figure out that our model is capable of detecting unseen objects under different scenarios: (a) a single object in an image, *e.g.*, "car", "train" and "sofa"; (b) multiple objects from the same category, *e.g.*, "car" and "dog"; (c) multiple objects from different categories, *e.g.*, "sofa" and "dog'. Besides, we have also noted that our model is capable of detecting objects from both seen and unseen classes in the same image, as depicted in Fig. 5(b). For example, {"car", "person"}, {"sofa", "chair", "tvmonitor"} and {"dog", "pottedplant", "tvmonitor"} are detected on the same image respectively, where "car", "sofa" and "dog" are unseen objects. These examples confirm that the proposed model can be applied successfully to both the ZSD and GZSD tasks. For MS COCO, we show qualitative comparison between our model and the baseline method ConSE, both of which are based on the Faster R-CNN framework. From Fig. 6, it's interesting to see that ConSE can localize



(a) Detection results on the 48/17 split of MS COCO



(b) Detection results on the 65/15 split of MS COCO

Fig. 6. Some ZSD and GZSD detection results on two splits of the MS COCO dataset. For each split, the detection results in the first and second row are produced by ConSE and ContrastZSD respectively.

the bounding box for most of the objects from either seen or unseen classes, although it did not use any semantic information during training. We can attribute this to the good generalization ability of the region proposal network in Faster R-CNN that generates objects in an objectness manner. However, ConSE fails to predict the true class label for most of the unseen objects. For example, ConSE recognizes the "elephant" object as "cow" in Fig. 6(a), and "airplane" object as "kite" in Fig. 6(b), *etc*. By contrast, our method provides more accurate detection results for either seen or unseen objects in the selected images. Moreover, our model also successfully detects the objects that have been missed by ConSE, like the "tie" object in Fig. 6(a) and "suitcase" object in Fig. 6(b).

5 Conclusion

In this paper, we have made the first attempt to facilitate the zero-shot object detection task with contrastive learning, and developed a novel ContrastZSD framework for ZSD. To endow the model with the ability of detecting unseen objects, the proposed ContrastZSD incorporates two contrastive learning subnets guided by semantics information, both of which can boost the performance significantly. The RCCL subnet enables explicit knowledge transfer from seen classes to unseen classes, thereby alleviating the projection domain shift problem. To further bridge the visual-semantic gap, the RRCL subnet optimizes the visual data distribution in the joint embedding space to be more distinguishable based on class label information. The quantitative and qualitative experimental results confirm that the proposed framework improves the performance of both the ZSD and GZSD task.

Acknowledgment

This work is supported by the National Key Research and Development Program of China (2018YFB1004500), the National Nature Science Foundation of China (61872287, 61772411, 61532015 and 61672419), the Innovative Research Group of the National Natural Science Foundation of China (61721002), the Innovation Research Team of Ministry of

Education (IRT_17R86), and the Project of China Knowledge Center for Engineering Science and Technology.

REFERENCES

[1] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *CVPR*, 2018.
[2] Y. Zhu, C. Zhao, H. Guo, J. Wang, X. Zhao, and H. Lu, "Attention couplenet: Fully convolutional attention coupling network for object detection," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 113–126, 2018.
[3] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *NIPS*, 2016.
[4] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra r-cnn: Towards balanced learning for object detection," in *CVPR*, 2019.
[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.
[6] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *CVPR*, 2019.
[7] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *ECCV*, 2020.
[8] H. Zhang, H. Chang, B. Ma, N. Wang, and X. Chen, "Dynamic r-cnn: Towards high quality object detection via dynamic training," in *ECCV*, 2020.
[9] Y. Wu, Y. Chen, L. Yuan, Z. Liu, L. Wang, H. Li, and Y. Fu, "Rethinking classification and localization for object detection," in *CVPR*, 2020.
[10] Y. Annadani and S. Biswas, "Preserving semantic relations for zero-shot learning," in *CVPR*, 2018.
[11] L. Niu, J. Cai, A. Veeraraghavan, and L. Zhang, "Zero-shot learning via category-specific visual-semantic mapping and label refinement," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 965–979, 2018.
[12] C. Luo, Z. Li, K. Huang, J. Feng, and M. Wang, "Zero-shot learning via attribute regression and class prototype rectification," *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 637–648, 2017.
[13] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *ICML*, 2015.
[14] A. Bansal, K. Sikka, G. Sharma, R. Chellappa, and A. Divakaran, "Zero-shot object detection," in *ECCV*, 2018.
[15] S. Rahman, S. Khan, and F. Porikli, "Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts," in *ACCV*, 2018.
[16] C. Yan, Q. Zheng, X. Chang, M. Luo, C.-H. Yeh, and A. G. Hauptman, "Semantics-preserving graph propagation for zero-shot object detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 8163–8176, 2020.
[17] Z. Li, L. Yao, X. Zhang, X. Wang, S. Kanhere, and H. Zhang, "Zero-shot object detection with textual descriptions," in *AAAI*, 2019.
[18] B. Demirel, R. G. Cinbis, and N. Ikizler-Cinbis, "Zero-shot object detection by hybrid region embedding," in *BMVC*, 2018.
[19] L. Zhang, X. Wang, L. Yao, L. Wu, and F. Zheng, "Zero-shot object detection via learning an embedding from semantic space to visual space," in *IJCAI*, 2020.
[20] D. Gupta, A. Anantharaman, N. Mamgain, V. N. Balasubramanian, C. Jawahar *et al.*, "A multi-space approach to zero-shot object detection," in *WACV*, 2020.
[21] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Transductive multi-view zero-shot learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 11, pp. 2332–2345, 2015.
[22] Y. Ye, Y. He, T. Pan, J. Li, and H. T. Shen, "Alleviating domain shift via discriminative learning for generalized zero-shot learning," *IEEE Transactions on Multimedia*, 2021.
[23] S. Zhao, C. Gao, Y. Shao, L. Li, C. Yu, Z. Ji, and N. Sang, "Gtnet: Generative transfer network for zero-shot object detection," in *AAAI*, vol. 34, no. 07, 2020, pp. 12 967–12 974.
[24] N. Hayat, M. Hayat, S. Rahman, S. Khan, S. W. Zamir, and F. S. Khan, "Synthesizing the unseen for zero-shot object detection," in *ACCV*, 2020.
[25] P. Zhu, H. Wang, and V. Saligrama, "Don't even look once: Synthesizing features for zero-shot detection," in *CVPR*, 2020.
[26] Y. Li, Z. Jia, J. Zhang, K. Huang, and T. Tan, "Deep semantic structural constraints for zero-shot learning," in *AAAI*, 2018.
[27] Y. Li and D. Wang, "Joint learning of attended zero-shot features and visual-semantic mapping," in *BMVC*, 2019.
[28] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.
[29] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.
[31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *ECCV*, 2016.
[32] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *CVPR*, 2017.
[33] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017.
[34] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 3, pp. 453–465, 2013.
[35] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *CVPR*, 2009.
[36] J. Lei Ba, K. Swersky, S. Fidler *et al.*, "Predicting deep zero-shot convolutional neural networks using textual descriptions," in *ICCV*, 2015.
[37] Y. Yang and T. M. Hospedales, "A unified perspective on multi-domain and multi-task learning," *arXiv preprint arXiv:1412.7489*, 2014.
[38] A. Frome, G. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model," 2013.
[39] R. Socher, M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, and A. Y. Ng, "Zero-shot learning through cross-modal transfer," *arXiv preprint arXiv:1301.3666*, 2013.
[40] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *CVPR*, 2018.
[41] R. Felix, I. Reid, G. Carneiro *et al.*, "Multi-modal cycle-consistent generalized zero-shot learning," in *ECCV*, 2018.
[42] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele, "Latent embeddings for zero-shot classification," in *CVPR*, 2016.
[43] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, "Evaluation of output embeddings for fine-grained image classification," in *CVPR*, 2015.
[44] Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, and Y. Matsumoto, "Ridge regression, hubness, and zero-shot learning," in *Joint European conference on machine learning and knowledge discovery in databases*, 2015.
[45] X. Wang, S. Pang, J. Zhu, Z. Li, Z. Tian, and Y. Li, "Visual space optimization for zero-shot learning," *arXiv preprint arXiv:1907.00330*, 2019.
[46] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *ICML*, 2020.
[47] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.
[48] Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, and P. Isola, "What makes for good views for contrastive learning," *arXiv preprint arXiv:2005.10243*, 2020.
[49] X. Liu, F. Zhang, Z. Hou, Z. Wang, L. Mian, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *arXiv preprint arXiv:2006.08218*, vol. 1, no. 2, 2020.
[50] E. Xie, J. Ding, W. Wang, X. Zhan, H. Xu, Z. Li, and P. Luo, "Detco: Unsupervised contrastive learning for object detection," *arXiv preprint arXiv:2102.04803*, 2021.
[51] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for unpaired image-to-image translation," in *ECCV*, 2020.
[52] R. Qian, T. Meng, B. Gong, M.-H. Yang, H. Wang, S. Belongie, and Y. Cui, "Spatiotemporal contrastive video representation learning," *arXiv preprint arXiv:2008.03800*, 2020.
[53] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *arXiv preprint arXiv:2004.11362*, 2020.
[54] O. Majumder, A. Ravichandran, S. Maji, M. Polito, R. Bhotika, and S. Soatto, "Revisiting contrastive learning for few-shot classification," *arXiv preprint arXiv:2101.11058*, 2021.

[55] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013.
[56] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *EMNLP*, 2014.
[57] Y. Zheng, R. Huang, C. Han, X. Huang, and L. Cui, "Background learnable cascade for zero-shot object detection," in *ACCV*, 2020.
[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.
[59] T. Kong, F. Sun, C. Tan, H. Liu, and W. Huang, "Deep feature pyramid reconfiguration for object detection," in *ECCV*, 2018.
[60] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
[61] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.
[62] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.
[63] S. Rahman, S. Khan, and N. Barnes, "Polarity loss for zero-shot object detection," *arXiv preprint arXiv:1811.08982*, 2018.
[64] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean, "Zero-shot learning by convex combination of semantic embeddings," in *ICLR*, 2014.
[65] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

Caixia Yan received the BS degree in Computer Science and technology from Xi'an Jiaotong University in 2015. She is currently working toward the Ph.D. degree in computer science and technology at Xi'an Jiaotong University. She is also working as a visiting scholar in the School of Computer Science at Carnegie Mellon University. Her research interests include machine learning and optimization, multiple feature learning and image processing.

Xiaojun Chang is a faculty member at Faculty of Information Technology, Monash University Clayton Campus, Australia. Before joining Monash, he was a Postdoc Research Associate in School of Computer Science, Carnegie Mellon University. He received his Ph.D. degree in Centre for Artificial Intelligence & Faculty of Engineering and Information Technology, University of Technology Sydney. He has spent most of his time working on exploring multiple signals (visual, acoustic, textual) for automatic content analysis in unconstrained or surveillence videos.

Minnan Luo received the Ph. D. degree from the Department of Computer Science and Technology, Tsinghua University, China, in 2014. Currently, she is an Assistant Professor in the School of Electronic and Information Engineering at Xi'an Jiaotong University. She was a Post-Doctoral Research with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA. Her research interests include machine learning and optimization, cross-media retrieval and fuzzy system.

Huan Liu received the B.S. and Ph.D. degrees in computer science from Xi'an Jiaotong University, China, in 2013 and 2020, respectively. He is currently an assistant professor at Xi'an Jiaotong University. His research areas include deep learning, machine learning, and their application in computer vision and EEG-based affective computing.

Xiaoqin Zhang received the B.Sc. degree in electronic information science and technology from Central South University, China, in 2005, and the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2010. He is currently a professor with Wenzhou University, China. He has authored or co-authored over 100 papers in international and national journals and international conferences. His research interests are in pattern recognition, computer vision, and machine learning.

Qinghua Zheng received the B.S. degree in computer software in 1990, the M.S. degree in computer organization and architecture in 1993, and the Ph.D. degree in system engineering in 1997 from Xi'an Jiaotong University, China. He was a postdoctoral researcher at Harvard University in 2002. He is currently a professor in Xi'an Jiaotong University. His research areas include computer network security, intelligent E-learning theory and algorithm, multimedia e-learning, and trustworthy software.