Zero-Shot Object Detection

Ankan Bansal[1(✉)], Karan Sikka[2], Gaurav Sharma[3], Rama Chellappa[1], and Ajay Divakaran[2]

[1] University of Maryland, College Park, MD, USA
{ankan,rama}@umiacs.umd.edu
[2] SRI International, Princeton, NJ, USA
{karan.sikka,ajay.divakaran}@sri.com
[3] NEC Labs America, Cupertino, CA, USA
grv@nec-labs.com

Abstract. We introduce and tackle the problem of zero-shot object detection (ZSD), which aims to detect object classes which are not observed during training. We work with a challenging set of object classes, not restricting ourselves to similar and/or fine-grained categories as in prior works on zero-shot classification. We present a principled approach by first adapting visual-semantic embeddings for ZSD. We then discuss the problems associated with selecting a background class and motivate two background-aware approaches for learning robust detectors. One of these models uses a fixed background class and the other is based on iterative latent assignments. We also outline the challenge associated with using a limited number of training classes and propose a solution based on dense sampling of the semantic label space using auxiliary data with a large number of categories. We propose novel splits of two standard detection datasets – MSCOCO and VisualGenome, and present extensive empirical results in both the traditional and generalized zero-shot settings to highlight the benefits of the proposed methods. We provide useful insights into the algorithm and conclude by posing some open questions to encourage further research.

1 Introduction

Humans can effortlessly make a mental model of an object using only textual description, while machine recognition systems, until not very long ago, needed to be shown visual examples of every category of interest. Recently, some work has been done on *zero-shot* classification using textual descriptions [53], leveraging progress made on both visual representations [51] and semantic text embeddings [21,34,39]. In zero-shot classification, at training time visual examples are provided for some visual classes but during testing the model is expected to recognize instances of classes which were not seen, with the constraint that the new classes are semantically related to the training classes.

A. Bansal—Most of the work was done when AB was an intern at SRI International.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11205, pp. 397–414, 2018.
https://doi.org/10.1007/978-3-030-01246-5_24

This problem is solved within the framework of transfer learning [13,40], where visual models for seen classes are transferred to the unknown classes by exploiting semantic relationships between the two. For example, as shown in Fig. 1, the semantic similarities between classes "hand" and "arm" are used to detect an instance of a related (unseen) class "shoulder". While such a setting has been used for object classification, object detection has remained mostly in the fully supervised setting as it is much more challenging. In comparison to object classification, which aims to predict the class label of an object in an image, object detection aims at predicting bounding box locations for multiple objects in an image. While classification can rely heavily on contextual cues, e.g. airplane co-occurring with clouds, detection needs to exactly localize the object of interest and can potentially be degraded by contextual correlations [56]. Furthermore, object detection requires learning additional invariance to appearance, occlusion, viewpoint, aspect ratio etc. in order to precisely delineate a bounding box [19].

In the past few years, several CNN-based object detection methods have been proposed. Early methods [16,17] started with an object proposal generation step and classified each object proposal as belonging to a class from a fixed set of categories. More recent methods either generate proposals inside a CNN [46], or have implicit regions directly in the image or feature maps [32,44]. These methods achieved significant performance improvements on small datasets which contain tens to a few hundreds of object categories [8,30]. However, the problem of detecting a large number of classes of objects has not received sufficient attention. This is mainly due to the lack of available annotated data as getting bounding box annotations for thousands of categories of objects is an expensive process. Scaling supervised detection to the level of classification (tens to hundreds of thousands of classes) is infeasible due to prohibitively large annotations costs. Recent works have tried to avoid such annotations, e.g. [45] proposed an object detection method that can detect several thousand object classes by using available (image-level) class annotations as weak supervision for object detection. Zero-shot learning has been shown to be effective in situations where there is a lack of annotated data [12,14,31,38,53,54,59,60]. Most prior works on zero-shot learning have addressed the classification problem [5–7,11,20,23,26,27,37,41,43,50,52], using semantic word-embeddings [11,23] or attributes [12,27,28,59] as a bridge between seen and unseen classes.

In the present work, we introduce and study the challenging problem of *zero-shot detection* for diverse and general object categories. This problem is difficult owing to the multiple challenges involved with detection, as well as those with operating in a zero-shot setting. Compared to fully supervised object detection, zero-shot detection has many differences, notably the following. While in the fully supervised case a background class is added to better discriminate between objects (e.g. car, person) and background (e.g. sky, wall, road), the meaning of "background" is not clear for zero-shot detection, as it could involve both background"stuff" as well as objects from unannotated/unseen classes. This leads to non-trivial practical problems for zero-shot detection. We propose two ways to address this problem: one using a fixed background class and the other using a

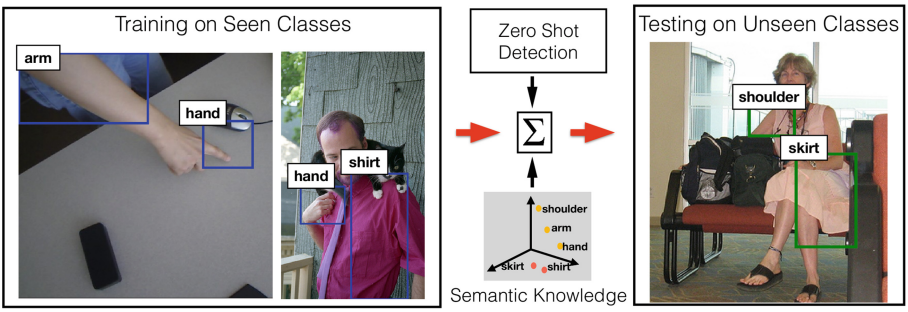

Fig. 1. We highlight the task of zero-shot object detection where objects "arm", "hand", and "shirt" are observed (seen) during training, but "skirt", and "shoulder" are not. These unseen classes are localized by our approach that leverages semantic relationships between seen and unseen classes along with the proposed zero-shot detection framework. The example has been generated by our model.

large open vocabulary for differentiating different background regions. We start with a standard zero-shot classification architecture [13] and adapt it for zero-shot object detection. This architecture is based on embedding both images and class labels into a common vector space. In order to include information from background regions, following supervised object detection, we first try to associate the background image regions into a single background class embedding. However, this method can be improved by using a latent assignment based alternating algorithm which associates the background boxes to potentially different classes belonging to a large open vocabulary. Since most object detection benchmark datasets usually have a few hundred classes, the label space can be sparsely populated. We show that dense sampling of the class label space by using additional data improves zero-shot detection. Along with these two enhancements, we provide qualitative and quantitative results to provide insights into the success as well as failure cases of the zero-shot detection algorithms, that point us to novel directions towards solving this challenging problem.

To summarize, the main contributions of this paper are: (i) we introduce the problem of zero-shot object detection (ZSD) in real world settings and present a baseline method for ZSD that follows existing work on zero-shot image classification using multimodal semantic embeddings and fully supervised object detection; (ii) we discuss some challenges associated with incorporating information from background regions and propose two methods for training background-aware detectors; (iii) we examine the problem with sparse sampling of classes during training and propose a solution which densely samples training classes using additional data; and (iv) we provide extensive experimental and ablation studies in traditional and generalized zero-shot settings to highlight the benefits and shortcomings of the proposed methods and provide useful insights which point to future research directions.

2 Related Work

Word Embeddings. Word embeddings map words to a continuous vector representation by encoding semantic similarity between words. Such representations are trained by exploiting co-occurrences in words in large text corpuses [21,34,35,39]. These word vectors perform well on tasks such as measuring semantic and syntactic similarities between words. In this work we use the word embeddings as the common vector space for both images and class labels and thus enable detection of objects from unseen categories.

Zero-Shot Image Classification. Previous methods for tackling zero-shot classification used attributes, like shape, color, pose or geographical information as additional sources of information [10,26,27]. More recent approaches have used multimodal embeddings to learn a compatibility function between an image vector and class label embeddings [1,2]. In [52], the authors augment the bilinear compatibility model by adding latent variables. The deep visual-semantic embedding model [11] used labeled image data and semantic information from unannotated text data to classify previously unseen image categories. We follow a similar methodology of using labeled object bounding boxes and semantic information in the form of unsupervised word embeddings to detect novel object categories. For a more compehensive overview of zero-shot classification, we refer the reader to the detailed survey by Fu et al. [13].

Object Detection. Early object detection approaches involved getting object proposals for each image and classifying those object proposals using an image classification CNN [16,17,46,55]. More recent approaches use a single pass through a deep convolution network without the need for object region proposals [32,44]. Recently, Redmon et al. [45] introduced an object detector which can scale upto 9000 object categories using both bounding box and image-level annotations. Unlike this setting, we work in a more challenging setting and do not observe any labels for the test object classes during training. We build our detection framework on an approach similar to the proposal-based approaches mentioned above.

Multi-modal Learning. Using multiple modalities as additional sources of information has been shown to improve performance on several computer vision and machine learning tasks. These methods can be used for cross-modal retrieval tasks [9], or for transferring classifiers between modalities. Recently, [4] used images, text, and sound for generating deep discriminative representations which are shared across the three modalities. Similarly, [58] used images and text descriptions for better natural language based visual entity localization. In [18], the authors used a shared vision and language representation space to obtain image-region and word descriptors that can be shared across multiple vision and language domains. Our work also uses multi-modal learning for building a robust object detector for unseen classes. Another related work is by Li et al. [28], which learns object-specific attributes to classify, segment, and predict novel objects. The problem proposed here differs considerably from this in detecting a large set of objects in unconstrained settings and does not rely on using attributes.

Comparison with Recent Works on ZSD. After completion of this work, we found two parallel works by Zhu et al. [61] and Rahman et al. [42] that target a similar problem. Zhu et al. focus on a different problem of generating object proposals for unseen objects. Rahman et al. [42] propose a loss formulation that combines max-margin learning and a semantic clustering loss. Their aim is to separate individual classes and reduce the noise in semantic vectors. A key difference between our work and Rahman et al. is the choice of evaluation datasets. Rahman et al. use the ILSVRC-2017 detection dataset [47] for training and evaluation. This dataset is more constrained in comparison to the ones used in our work (MSCOCO and VisualGenome) because it contains only about one object per image on an average. We would also like to note that due to a relatively simpler test setting, Rahman et al. does not consider the corrruption of the background class by unseen classes as done in this work and by Zhu et al.

3 Approach

We first outline our baseline zero-shot detection framework that adapts prior work on zero-shot learning for the current task. Since this approach does not consider the diversity of the background objects during training, we then present an approach for training a background-aware detector with a fixed background class. We highlight some possible limitations of this approach and propose a latent assignment based background-aware model. Finally, we describe our method for densely sampling labels using additional data, which improves generalization.

3.1 Baseline Zero-Shot Detection (ZSD)

We denote the set of all classes as $\mathcal{C} = \mathcal{S} \cup \mathcal{U} \cup \mathcal{O}$, where $\mathcal{S}$ denotes the set of seen (train) classes, $\mathcal{U}$ the set of unseen (test) classes, and $\mathcal{O}$ the set of classes that are neither part of seen or unseen classes. Note that our methods do not require a pre-defined test set. We fix the unseen classes here just for quantitative evaluation. We work in a zero-shot setting for object detection where, during training we are provided with labeled bounding boxes that belong to the seen classes only, while during testing we detect objects from unseen classes. We denote an image as $I \in \mathbb{R}^{M \times N \times 3}$, provided bounding boxes as $b_i \in \mathbb{N}^4$, and their associated labels as $y_i \in \mathcal{S}$. We extract deep features from a given bounding box obtained from an arbitrary region proposal method. We denote extracted deep features for each box b_i as $\phi(b_i) \in \mathbb{R}^{D_1}$. We use semantic embeddings to capture the relationships between seen and unseen classes and thus transfer a model trained on the seen classes to the unseen classes as described later. We denote the semantic embeddings for different class labels as $w_j \in \mathbb{R}^{D_2}$, which can be obtained from pre-trained word embedding models such as Glove [39] or fastText [21]. Our approach is based on visual-semantic embeddings where both image and text features are embedded in the same metric space [11,50]. We project features from the bounding box to the semantic embedding space itself via a linear projection,

$$\psi_i = W_p \phi(b_i) \tag{1}$$

where, $W_p \in \mathbb{R}^{D_2 \times D_1}$ is a projection matrix and ψ_i is the projected feature. We use the common embedding space to compute a similarity measure between a projected bounding box feature ψ_i and a class embedding w_j for class label y_j as the cosine similarity S_{ij} between the two vectors. We train the projection by using a max-margin loss which enforces the constraint that the matching score of a bounding box with its true class should be higher than that with other classes. We define loss for a training sample b_i with class label y_i as,

$$\mathcal{L}(b_i, y_i, \theta) = \sum_{j \in \mathcal{S}, j \neq i} \max(0, m - S_{ii} + S_{ij}) \tag{2}$$

where θ refers to the parameters of the deep CNN and the projection matrix, and m is the margin. We also add an additional reconstruction loss to $\mathcal{L}$, as suggested by Kodirov et al. [23], to regularize the semantic embeddings. In particular, we use the projected box features to reconstruct the original deep features and calculate the reconstruction loss as the squared $L2$-distance between the reconstructed feature and the original deep feature. During test we predict the label $(\hat{y}_i)$ for a bounding box (b_i) by finding its nearest class based on the similarity scores with different class embeddings, i.e.

$$\hat{y}_i = \arg\max_{j \in \mathcal{U}} S_{ij} \tag{3}$$

It is common for object detection approaches to include a background class to learn a robust detector that can effectively discriminate between foreground objects and background objects. This helps in eliminating bounding box proposals which clearly do not contain any object of interest. We refer to these models as background-aware detectors. However, selecting a background for zero-shot detection is a non-trivial problem as we do not know if a given background box includes background "stuff" in the classical sense e.g. sky, ground etc. or an instance of an unseen object class. We thus train our first (baseline) model only on bounding boxes that contain seen classes.

3.2 Background-Aware Zero-Shot Detection

While background boxes usually lead to improvements in detection performance for current object detection methods, for ZSD to decide which background bounding boxes to use is not straight-forward. We outline two approaches for extending the baseline ZSD model by incorporating information from background boxes during training.

Statically Assigned Background (SB) Based Zero-Shot Detection. Our first background-aware model follows as a natural extension of using a fixed background class in standard object detectors to our embedding framework. We accomplish this by adding a fixed vector for the background class in our embedding space. Such 'statically-assigned' background modeling in ZSD, while providing a way to incorporate background information, has some limitations. First, we

are working with the structure imposed by the semantic text embeddings that represent each class by a vector relative to other semantically related classes. In such a case it is difficult to learn a projection that can map all the diverse background appearances, which surely belong to semantically varied classes, to a single embedding vector representing one monolithic background class. Second, even if we are able to learn such a projection function, the model might not work well during testing. It can map any unseen class to the single vector corresponding to the background, as it has learned to map everything, which is not from seen classes, to the singleton background class.

Latent Assignment Based (LAB) Zero-Shot Detection. We solve the problems above by spreading the background boxes over the embedding space by using an Expectation Maximization (EM)-like algorithm. We do so by assigning multiple (latent) classes to the background objects and thus covering a wider range of visual concepts. This is reminiscent of semi-supervised learning algorithms [48]; we have annotated objects for seen classes and unlabeled boxes for the rest of the image regions. At a higher level we encode the knowledge that a background box does not belong to the set of seen classes ($\mathcal{S}$), and could potentially belong to a number of different classes from a large vocabulary set, referred to as background set and denoted as $\mathcal{O}$.

We first train a baseline ZSD model on boxes that belong to the seen classes. We then follow an iterative EM-like training procedure (Algorithm 1), where, in the first of two alternating steps, we assign labels to some randomly sampled background boxes in the training set as classes in $\mathcal{O}$ using our trained model with equation 3. In the second step, we re-train our detection model with the boxes, labeled as above, included. In the next iteration, we repeat the first step for another part of background boxes and retrain our model with the new training data. This proposed approach is also related to open-vocabulary learning where we are not restricted by a fixed set of classes [20,57], and to latent-variable based classification models e.g. [49].

3.3 Densely Sampled Embedding Space (DSES)

The ZSD method, described above, relies on learning a common embedding space that aligns object features with label embeddings. A practical problem in learning such a model with small datasets is that there are only a small number of seen classes, which results in a sparse sampling of the embedding space during training. This is problematic particularly for recognizing unseen classes which, by definition, lie in parts of the embedding space that do not have training examples. As a result the method may not converge towards the right alignment between visual and text modalities. To alleviate this issue, we propose to augment the training procedure with additional data from external sources that contain boxes belonging to classes other than unseen classes, $y_i \in \mathcal{C} - \mathcal{U}$. In other words, we aim to have a dense sampling of the space of object classes during training to improve the alignment of the embedding spaces. We show empirically that, because the extra data being used is from diverse external sources and is distinct from seen and unseen classes, it improves the baseline method.

Algorithm 1. LAB algorithm

Given: `annoData` (annotated data), `bgData` (background/unannotated data), $\mathcal{C}$ (set of all classes), $\mathcal{S}$ (seen classes), $\mathcal{U}$ (unseen classes), $\mathcal{O}$ (background set), `initModel` (pre-trained network)
`currModel` ← train(`initModel`, `annoData`)
for $i = 1$ to `niters` **do**
  `currBgData` ← ϕ
  for b in `bgData` **do**
    // distribute background boxes over open vocabulary minus seen classes
    b_{new} ←predict(b, `currModel`, $\mathcal{O}$)
    // $\mathcal{O} = \mathcal{C} \setminus (\mathcal{S} \cup \mathcal{U})$
    `currBgData` ← `currBgData` $\cup\{b_{new}\}$
  `currAnnoData` ← `annoData` ∪ `currBgData`
  `currModel`←train(`currModel`,`currAnnoData`)
return `currModel`

4 Experiments

We first describe the challenging public datasets we use to validate the proposed approaches, and give the procedure for creating the novel training and test splits[1]. We then discuss the implementation details and the evaluation protocol. Thereafter, we give the empirical performance for different models followed by some ablation studies and qualitative results to provide insights into the methods.

MSCOCO [30] We use training images from the 2014 training set and randomly sample images for testing from the validation set.

VisualGenome (VG) [25] We remove non-visual classes from the dataset; use images from part-1 of the dataset for training, and randomly sample images from part-2 for testing.

OpenImages (OI) [24] We use this dataset for densely sampling the label space as described in Sect. 3.3. It contains about 1.5 million images containing 3.7 million bounding boxes that span 545 object categories.

Procedure for Creating Train and Test Splits: For dividing the classes into seen (train) and unseen (test) classes, we use a procedure similar to [3]. We begin with word-vector embeddings for all classes and cluster them into K clusters using cosine similarity between the word-vectors as the metric. We randomly select 80% classes from each cluster and assign these to the set of seen classes. We assign the remaining 20% classes from each cluster to the test set. We set the number of clusters to 10 and 20 for MSCOCO and VisualGenome respectively. Out of all the available classes, we consider only those which have a synset associated with them in the WordNet hierarchy [36] and also have a word vector available. This gives us 48 training classes and 17 test classes for

[1] Visit http://ankan.umiacs.io/zsd.html.

MSCOCO and 478 training classes and 130 test classes for VisualGenome. For MSCOCO, to avoid taking unseen categories as background boxes, we remove all images from the training set which contain any object from unseen categories. However, we can not do this for VG because the large number of test categories and dense labeling results in most images being eliminated from the training set. After creating the splits we have 73,774 training and 6,608 test images for MSCOCO, and 54,913 training and 7,788 test images for VG.

4.1 Implementation Details

Preparing Datasets for Training: We first obtain bounding box proposals for each image in the training set. We construct the training datasets by assigning each proposal a class label from seen classes or the "background" class based on its IoU (Intersection over Union) with a ground truth bounding box. Since, majority of the proposals belong to background, we only include a part of the background boxes. Any proposal with $0 < \text{IoU} < 0.2$ with a ground truth bounding box is included as a background box in the training set. Apart from these, we also include a few randomly selected background boxes with $\text{IoU} = 0$ with any ground truth bounding boxes. Any proposal with an $\text{IoU} > 0.5$ with a ground-truth box is assigned to the class of the ground-truth box. Finally, we get 1.4 million training boxes for MSCOCO and 5.8 million training boxes for VG. We use these boxes for training the two background aware models. As previously mentioned, we only use boxes belonging to seen classes for training the baseline ZSD model. In this case, we have 0.67 million training boxes for MSCOCO and about 2.6 million training boxes for VG. We train our model on these training sets and test them on the test sets as described above.

Baseline ZSD Model: We build our ZSD model on the RCNN framework that first extracts region proposals, warps them, and then classifies them. We use the Edge-Boxes method [62] with its default parameters for generating region proposals and then warp them to an image of size 224×224. We use the (pre-trained) Inception-ResNet v2 model [51] as our base CNN for computing deep features. We project image features from a proposal box to the 300 dimensional semantic text space by adding a fully-connected layer on the last layer of the CNN. We use the Adam optimizer [22] with a starting learning rate of 10^{-3} for the projection matrix and 10^{-5} for the lower layers. The complete network, including the projection layer, is first pre-trained on the MSCOCO dataset with the test classes removed for different models and datasets. For each algorithm, we perform end-to-end training while keeping the word embeddings fixed. The margin for ranking loss was set to 1 and the reconstruction loss was added to max-margin loss after multiplying it by a factor of 10^{-3}. We provide algorithm specific details below.

Static Background Based ZSD: In this case, we include the background boxes obtained as described above in the training set. The single background class is assigned a fixed label vector $[1, \ldots, 0]$ (this fixed background vector was chosen so as to have norm one similar to the other class embeddings).

LAB: We first create a vocabulary ($\mathcal{C}$) which contains all the words for which we have word-vectors and synsets in the WordNet hierarchy [36]. We then remove any label from seen and unseen classes from this set. The size of the vocabulary was about 82 K for VG and about 180 K for MSCOCO. In the first iteration, we use our baseline ZSD model to obtain labels from the vocabulary set for some of the background boxes. We add these boxes with the newly assigned labels to the training set for the next iteration (see Algorithm 1). We fine-tune the model from the previous iteration using this new training set for about one epoch. During our experiments we iterate over this process five times. Our starting learning rates were the same as above and we decreased them by a factor of 10 after every 2 iterations.

Dense Sampling of the Semantic Space: To increase the label density, we use additional data from OI to augment the training sets for both VG and MSCOCO. We remove all our test classes from OI and add the boxes from remaining classes to the training sets. This led to an addition of 238 classes to VG and 330 classes to MSCOCO during training. This increases the number of training bounding boxes for VG to 3.3 million and to 1 million for MSCOCO.

4.2 Evaluation Protocol

During evaluation we use Edge-Boxes for extracting proposals for each image and select only those proposals that have a proposal score (given by Edge-Boxes) greater than 0.07. This threshold was set based on trade-offs between performance and evaluation time. We pass these proposals through the base CNN and obtain a score for each test class as outlined in Sect. 3.1. We apply greedy non-maximal suppression [17] on all the scored boxes for each test class independently and reject boxes that have an IoU greater than 0.4 with a higher scoring box. We use recall as the main evaluation metric for detection instead of the commonly used mean average precision (mAP). This is because, for large-scale crowd-sourced datasets such as VG, it is often difficult to exhaustively label bounding box annotations for all instances of an object. Recall has also been used in prior work on detecting visual relationships [33] where it is infeasible to annotate all possible instances. The traditional mAP metric is sensitive to missing annotations and will count such detections as false positives. We define Recall@K as the recall when only the top K detections (based on prediction score) are selected from an image. A predicted bounding box is marked as true positive only if it has an IoU overlap greater than a certain threshold t with a ground truth bounding box and no other higher confidence predicted bounding box has been assigned to the same ground truth box. Otherwise it is marked as a false positive. For MSCOCO we also report the mAP since all object instances in MSCOCO are annotated.

4.3 Quantitative Results

We present extensive results (Recall@100) for different algorithms on MSCOCO and VG datasets in Table 1 for three different IoU overlap thresholds. We also

Table 1. $|\mathcal{S}|$, $|\mathcal{U}|$, and $|\mathcal{O}|$ refer to the number of seen, unseen and the average number of active background classes considered during training respectively. BG-aware means background-aware representations. This table shows Recall@100 performance for the proposed zero-shot detection approaches (see Sect. 3) on the two datasets at different IoU overlap thresholds with the ground-truth boxes. The numbers in parentheses are mean average precision (mAP) values for MSCOCO. The number of test (unseen) classes for MSCOCO and VisualGenome are 17 and 130 respectively.

		MSCOCO						**Visual Genome**					
ZSD Method	BG-aware	#classes			IoU			#classes			IoU		
		$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{U}\rvert$	$\lvert\mathcal{O}\rvert$	0.4	0.5	0.6	$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{U}\rvert$	$\lvert\mathcal{O}\rvert$	0.4	0.5	0.6
Baseline		48	17	0	34.36	22.14 (0.32)	11.31	478	130	0	8.19	5.19	2.63
SB	✓	48	17	1	34.46	24.39 (0.70)	12.55	478	130	1	6.06	4.09	2.43
DSES		378	17	0	**40.23**	**27.19** (0.54)	**13.63**	716	130	0	7.78	4.75	2.34
LAB	✓	48	17	343	31.86	20.52 (0.27)	9.98	478	130	1673	**8.43**	**5.40**	**2.74**

show the number of seen, unseen, and background classes for each case. During our discussion we report Recall@100 at a threshold of IoU ≥ 0.5 unless specified otherwise.

On the VG dataset the baseline model achieves 5.19% recall and the static background (SB) model achieves a recall of 4.09%. This marked decline in performance is because all the background boxes are being mapped to a single vector. In VG some of these background boxes might actually belong to the seen (train) or unseen (test) categories. This leads to the SB model learning sub-optimal visual embeddings. However, for MSCOCO we observe that the SB model increases the recall to 24.39% from the 22.14% achieved by the baseline model. This is because we remove all images that contain any object from unseen classes from the training set for MSCOCO. This precludes the possibility of having any background boxes belonging to the test classes in the training set. As a result, the SB model is not corrupted by non-background objects and is thus more robust than the baseline.

When we densely sample the embedding space and augment the training classes with additional data, the recall for MSCOCO increases significantly from 22.14% (for baseline) to 27.19%. This shows that dense sampling is beneficial for predicting unseen classes that lie in sparsely sampled parts of the embedding space. With dense sampling, the number of train classes in MSCOCO are expanded by a factor of 7.8 to 378. In contrast, VG *a priori* has a large set of seen classes (478 versus 48 in MSCOCO), and the classes expand only by a factor of 1.5 (716) when using DSES. As a result dense sampling is not able to improve the embedding space obtained by the initial set of categories. In such scenarios it might be beneficial to use more sophisticated methods for sampling additional classes that are not represented well in the training set [15,29,40].

The latent assignment based (LAB) method outperforms the baseline, SB, and DSES on VG. It achieves a recall of 5.40% compared to 5.19%, 4.09% and 4.75% achieved by baseline, SB, and DSES respectively. The consistent

improvement across all IoUs compared to SB, that uses a static background, confirms the benefits of spreading background objects over the embedding space. However, LAB gives a lower performance compared to the baseline for MSCOCO (20.52% by LAB versus 22.14% by baseline). This is not surprising since the iterations for LAB initialize with a larger set of seen classes for VG as compared to MSCOCO, resulting in an embedding that covers a wider spectrum of visual space. As a result, LAB is able to effectively spread the background boxes over a larger set of classes for VG leading to better detections. On the other hand, for MSCOCO a sparsely sampled embedding space restricts the coverage of visual concepts leading to the background boxes being mapped to a few visual categories. We also see this empirically in the average number of background classes (set $\mathcal{O}$) assigned to the background boxes during iterations for LAB, which were 1673 for VG versus 343 for MSCOCO. In the remainder of the paper we focus on LAB method for VG and SB for MSCOCO due to their appropriateness for the respective datasets.

We observe that the relative class-wise performance trends are similar to object detection methods, such as Faster RCNN[2] trained on fully supervised data. For example, classes such as "bus" and "elephant" are amongst the best performing while "scissors" and "umbrella" rank amongst the worst in performance. In addition to these general trends, we also discover some interesting findings due to the zero-shot nature of the problem. For example, the class "cat", which generally performs well with standard object detectors, did not perform well with SB. This results from having an insufficient number of semantically related categories for this class in the training set which does not allow the model to effectively capture the appearance of class "cat" during testing. For such cases we find dense sampling to be useful during training. The class "cat" is one of the top performing categories with DSES. Based on such cases we infer that for ZSD the performance is both a function of appearance characteristics of the class as well as its relationship to the seen classes. For VG, the best performing classes, such as "laptop", "car", "building", "chair", seem to have well defined appearance characteristics compared to bad performing classes, such as "gravel", "vent", "garden", which seem to be more of"stuff" than "things". We also observe that the model is unable to capture any true positive for the class "zebra" and is instead detecting instances of "zebra" as either "cattle" or "horse". This is because the model associates a "zebra" with a "giraffe", which is close in the semantic space. The model is able to adapt the detector for the class "giraffe" to the class "zebra" but fails to infer additional knowledge needed for a successful detector that a zebra differs from a giraffe in having white stripes, lower height, and has a body structure similar to a horse. Finally, we also observe that compared to the baseline, LAB achieves similar or better performance on 104 of 130 classes on VG. While for MSCOCO, SB and DSES achieve better or similar performance on 12 and 13 classes respectively out of 17 classes, highlighting the advantages of the proposed models.

[2] http://cocodataset.org/#detections-leaderboard.

4.4 Generalized Zero-Shot Detection (GZSD)

The generalized zero-shot learning setting is more realistic than the previously discussed zero-shot setting [53] because both seen and unseen classes are present during evaluation. This is more challenging than ZSD because it removes the prior knowledge that the objects at test time belong to unseen classes only. We use a simple novelty detection step which does not need extra supervision. Given a test bounding box, b_i, we first find the most probable train and test classes (see (3)) ($\hat{y}_i^s$ and $\hat{y}_i^u$ respectively) and the corresponding similarity scores (s_i and u_i). As the novelty detection step, we check if u_i is greater than some threshold n_t. We assign the given bounding box to class $\hat{y}_i^u$ if $u_i \geq n_t$, otherwise to $\hat{y}_i^s$. For MSCOCO, DSES gives the best performance in the GZSD setting too. At $n_t = 0.2$, DSES achieves a Recall@100 of 15.02% for seen classes and 15.32% for unseen classes (harmonic mean (HM) 15.17% [53]) at $IoU \geq 0.5$ compared to 14.54% and 10.57% (HM 12.24%) for the LAB model and 16.93% and 8.91% (HM 11.67%) for baseline.

4.5 Ablation Studies

We compare results when considering different number, K, of high-confidence detections. We define $K = All$ as the scenario where we consider all boxes returned by the detector with a confidence score greater than the threshold for evaluation. We compare LAB and the SB models for VG and MSCOCO respectively, with the corresponding baseline models in Table 2.

Table 2. Ablation studies on background-aware approaches for ZSD. We highlight results where the performance is higher for background-aware approaches compared to the corresponding baseline. For MSCOCO, the values in parentheses are mAP values.

MSCOCO

	Baseline			SB		
K↓ IoU→	0.3	0.4	0.5	0.3	0.4	0.5
All	47.91	37.86	24.47 (0.22)	43.79	35.58	**25.12** (0.64)
100	43.62	34.36	22.14 (0.32)	42.22	**34.46**	**24.39** (0.70)
80	41.69	32.64	21.01 (0.38)	41.47	**33.98**	**24.01** (0.72)
50	36.19	27.37	17.05 (0.50)	**39.82**	**32.6**	**23.16** (0.81)

VisualGenome

Baseline			LAB		
0.3	0.4	0.5	0.3	0.4	0.5
13.88	9.98	6.45	12.75	9.61	6.22
11.34	8.19	5.19	11.20	**8.43**	**5.40**
10.41	7.55	4.75	**10.45**	**7.86**	**5.06**
7.98	5.79	3.68	**8.54**	**6.44**	**4.14**

The difference in performance between the cases $K = All$ and $K = 100$ is small, in general, for the background-aware algorithms unlike the baseline. For example, on MSCOCO the recall for SB falls by an average (across IoUs) of 1.14% points, compared to a fall of 3.37% for the baseline. This trend continues further down to $K = 80$ and $K = 50$ with a gradual decline in performance as K decreases. This shows that the high confidence detections produced by our model are of high quality.

We observe that the background-aware models give better quality detections compared to baselines. The Recall@K for the corresponding background-aware

models are better than the baseline at lower K and higher IoU threshold values for both datasets. This region represents higher quality detections. This shows that incorporating knowledge from background regions is an important factor for improving detection quality and performance for ZSD.

4.6 Qualitative Results

Figure 2 shows output detections by the background aware models, i.e. LAB on VisualGenome (first two rows) and SB on MSCOCO (last row). Blue boxes show correct detections and red boxes show false positives. These examples confirm that the proposed models are able to detect unseen classes without observing any samples during training. Further, the models are able to successfully detect multiple objects in real-world images with background clutter. For example, in the image taken in an office (1^{st} row 3^{rd} column), the model is able to detect object classes such as "writing", "chair", "cars". It is also interesting to note that our approach understands and detects "stuff" classes such as "vegetation", and "floor". As discussed in Sect. 4.3, we have shown a failure case "zebra", that results from having limited information regarding the fine-grained differences between seen and unseen classes.

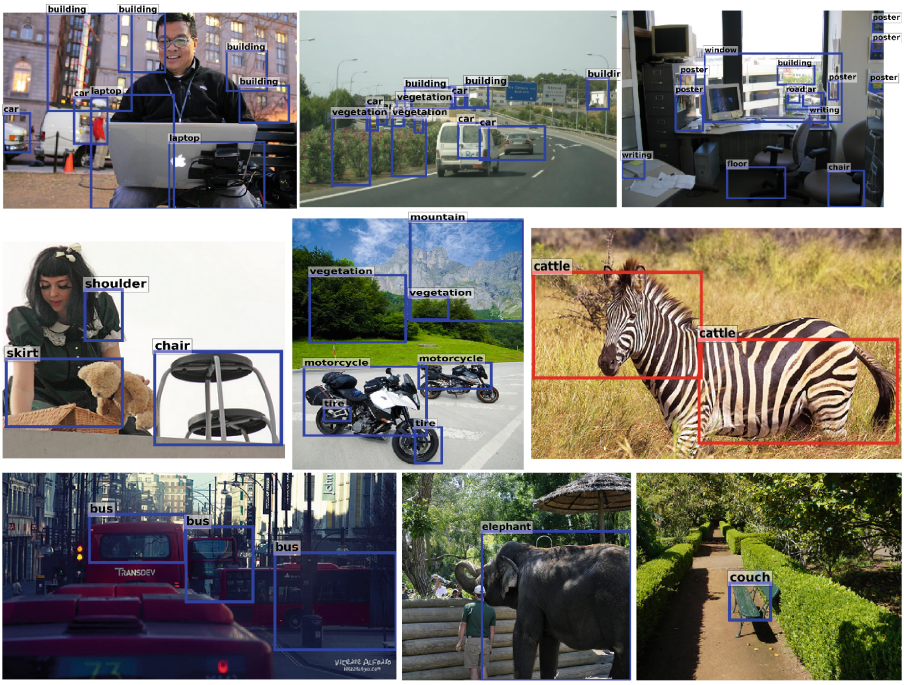

Fig. 2. This figure shows some detections made by the background-aware methods. We have used Latent Assignment Based model for VisualGenome (rows 1–2) and the Static Background model (row 3) for MSCOCO. Reasonable detections are shown in blue and two failure cases in red. (Color figure online)

5 Discussion and Conclusion

We used visual-semantic embeddings for ZSD and addressed the problems associated with the framework which are specific for ZSD. We proposed two background-aware approaches; the first one uses a fixed background class while the second iteratively assigns background boxes to classes in a latent variable framework. We also proposed to improve the sampling density of the semantic label space using auxiliary data. We proposed novel splits of two challenging public datasets, MSCOCO and VisualGenome, and gave extensive quantitative and qualitative results to validate the methods proposed.

Some of the limitations of the presented work, and areas for future work, are as follows. It is important to incorporate some lexical ontology information ("is a" and "is part of" relationships) during training and testing for learning models on large vocabularies. Most current object detection frameworks ignore the hierarchical nature of object classes. For example, a "cat" object should incur a lower loss when predicted as "animal" vs. when predicted as "vehicle". Although a few works have tried to address this issue [18,44], we believe further work in this direction would be beneficial for zero-shot detection. We also feel that additional work is needed to generalize bounding-box regression and hard-negative mining for new objects.

Acknowledgements. This project is sponsored by the Air Force Research Laboratory (AFRL) and Defense Advanced Research Projects Agency (DARPA) under the contract number USAF/AFMC AFRL FA8750-16-C-0158. **Disclaimer**: The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

The work of AB and RC is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DOI/IBC) contract number D17PC00345. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes not withstanding any copyright annotation thereon. **Disclaimer**: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied of IARPA, DOI/IBC or the U.S. Government.

We would like to thank the reviewers for their valuable comments and suggestions.

References

1. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for attribute-based classification. In: CVPR, pp. 819–826. IEEE (2013)
2. Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B.: Evaluation of output embeddings for fine-grained image classification. In: CVPR, pp. 2927–2936. IEEE (2015)
3. Anne Hendricks, L., Venugopalan, S., Rohrbach, M., Mooney, R., Saenko, K., Darrell, T., Mao, J., Huang, J., Toshev, A., Camburu, O., et al.: Deep compositional captioning: Describing novel object categories without paired training data. In: CVPR, pp. 1–10. IEEE (2016)

4. Aytar, Y., Vondrick, C., Torralba, A.: See, hear, and read: Deep aligned representations. arXiv preprint arXiv:1706.00932 (2017)
5. Bendale, A., Boult, T.E.: Towards open set deep networks. In: CVPR, pp. 1563–1572. IEEE (2016)
6. Changpinyo, S., Chao, W.L., Gong, B., Sha, F.: Synthesized classifiers for zero-shot learning. In: CVPR, pp. 5327–5336. IEEE (2016)
7. Elhoseiny, M., Saleh, B., Elgammal, A.: Write a classifier: Zero-shot learning using purely textual descriptions. In: CVPR, pp. 2584–2591. IEEE (2013)
8. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88**(2), 303–338 (2010)
9. Faghri, F., Fleet, D.J., Kiros, J.R., Fidler, S.: Vse++: Improved visual-semantic embeddings. arXiv preprint arXiv:1707.05612 (2017)
10. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Pose search: retrieving people using their pose. In: CVPR, pp. 1–8. IEEE (2009)
11. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T.: Devise: A deep visual-semantic embedding model. In: NIPS, pp. 2121–2129 (2013)
12. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S.: Attribute learning for understanding unstructured social activity. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 530–543. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33765-9_38
13. Fu, Y., Xiang, T., Jiang, Y.G., Xue, X., Sigal, L., Gong, S.: Recent advances in zero-shot recognition. arXiv preprint arXiv:1710.04837 (2017)
14. Fu, Y., Yang, Y., Hospedales, T., Xiang, T., Gong, S.: Transductive multi-label zero-shot learning. arXiv preprint arXiv:1503.07790 (2015)
15. Gavves, S., Mensink, T., Tommasi, T., Snoek, C., Tuytelaars, T.: Active transfer learning with zero-shot priors: reusing past datasets for future tasks. In: ICCV, pp. 2731–2739. IEEE (2015)
16. Girshick, R.: Fast R-CNN. In: ICCV, pp. 1440–1448. IEEE (2015)
17. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Region-based convolutional networks for accurate object detection and segmentation. TPAMI **38**(1), 142–158 (2016)
18. Gupta, T., Shih, K., Singh, S., Hoiem, D.: Aligned image-word representations improve inductive transfer across vision-language tasks. arXiv preprint arXiv:1704.00260 (2017)
19. Hoiem, D., Chodpathumwan, Y., Dai, Q.: Diagnosing error in object detectors. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part III. LNCS, vol. 7574, pp. 340–353. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33712-3_25
20. Jain, L.P., Scheirer, W.J., Boult, T.E.: Multi-class open set recognition using probability of inclusion. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part III. LNCS, vol. 8691, pp. 393–409. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10578-9_26
21. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016)
22. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
23. Kodirov, E., Xiang, T., Gong, S.: Semantic autoencoder for zero-shot learning. arXiv preprint arXiv:1704.08345 (2017)
24. Krasin, I., et al.: Openimages: A public dataset for large-scale multi-label and multi-class image classification. Dataset https://github.com/openimages (2017)

25. Krishna, R., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. arXiv preprint arXiv:1602.07332 (2016)
26. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR, pp. 951–958. IEEE (2009)
27. Lampert, C.H., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. TPAMI **36**(3), 453–465 (2014)
28. Li, Z., Gavves, E., Mensink, T., Snoek, C.G.M.: Attributes make sense on segmented objects. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VI. LNCS, vol. 8694, pp. 350–365. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10599-4_23
29. Lim, J.J., Salakhutdinov, R.R., Torralba, A.: Transfer learning by borrowing examples for multiclass object detection. In: NIPS, pp. 118–126 (2011)
30. 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: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
31. Liu, J., Kuipers, B., Savarese, S.: Recognizing human actions by attributes. In: CVPR, pp. 3337–3344. IEEE (2011)
32. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016, Part I. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
33. Lu, C., Krishna, R., Bernstein, M., Fei-Fei, L.: Visual relationship detection with language priors. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016, Part I. LNCS, vol. 9905, pp. 852–869. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_51
34. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
35. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 3111–3119 (2013)
36. Miller, G.A.: Wordnet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
37. Norouzi, M., Mikolov, T., Bengio, S., Singer, Y., Shlens, J., Frome, A., Corrado, G.S., Dean, J.: Zero-shot learning by convex combination of semantic embeddings. arXiv preprint arXiv:1312.5650 (2013)
38. Parikh, D., Kovashka, A., Parkash, A., Grauman, K.: Relative attributes for enhanced human-machine communication. In: AAAI (2012)
39. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. EMNLP **14**, 1532–1543 (2014)
40. Qi, G.J., Aggarwal, C., Rui, Y., Tian, Q., Chang, S., Huang, T.: Towards cross-category knowledge propagation for learning visual concepts. In: CVPR, pp. 897–904. IEEE (2011)
41. Qiao, R., Liu, L., Shen, C., Hengel, A.v.d.: Visually aligned word embeddings for improving zero-shot learning. arXiv preprint arXiv:1707.05427 (2017)
42. Rahman, S., Khan, S., Porikli, F.: Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. arXiv preprint arXiv:1803.06049 (2018)
43. Rahman, S., Khan, S.H., Porikli, F.: A unified approach for conventional zero-shot, generalized zero-shot and few-shot learning. arXiv preprint arXiv:1706.08653 (2017)

44. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: CVPR, pp. 779–788. IEEE (2016)
45. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. arXiv preprint arXiv:1612.08242 (2016)
46. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: NIPS, pp. 91–99 (2015)
47. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
48. Seeger, M.: Learning with labeled and unlabeled data. Technical report (2000)
49. Sharma, G., Jurie, F., Schmid, C.: Expanded parts model for semantic description of humans in still images. TPAMI **39**(1), 87–101 (2017)
50. Socher, R., Ganjoo, M., Manning, C.D., Ng, A.: Zero-shot learning through cross-modal transfer. In: NIPS, pp. 935–943 (2013)
51. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-ResNet and the impact of residual connections on learning. In: AAAI, pp. 4278–4284 (2017)
52. Xian, Y., Akata, Z., Sharma, G., Nguyen, Q., Hein, M., Schiele, B.: Latent embeddings for zero-shot classification. In: CVPR, pp. 69–77. IEEE (2016)
53. Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. arXiv preprint arXiv:1707.00600 (2017)
54. Xu, B., Fu, Y., Jiang, Y.G., Li, B., Sigal, L.: Heterogeneous knowledge transfer in video emotion recognition, attribution and summarization. TPAMI **9**, 255–270 (2016)
55. Xu, H., Lv, X., Wang, X., Ren, Z., Bodla, N., Chellappa, R.: Deep regionlets for object detection. CoRR abs/1712.02408 (2017)
56. Yu, R., Chen, X., Morariu, V.I., Davis, L.S.: The role of context selection in object detection. arXiv preprint arXiv:1609.02948 (2016)
57. Zhang, H., Shang, X., Yang, W., Xu, H., Luan, H., Chua, T.S.: Online collaborative learning for open-vocabulary visual classifiers. In: CVPR, pp. 2809–2817. IEEE (2016)
58. Zhang, Y., Yuan, L., Guo, Y., He, Z., Huang, I.A., Lee, H.: Discriminative bimodal networks for visual localization and detection with natural language queries. arXiv preprint arXiv:1704.03944 (2017)
59. Zhang, Z., Saligrama, V.: Zero-shot learning via joint latent similarity embedding. In: CVPR, pp. 6034–6042. IEEE (2016)
60. Zhang, Z., Saligrama, V.: Zero-shot recognition via structured prediction. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016, Part VII. LNCS, vol. 9911, pp. 533–548. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_33
61. Zhu, P., Wang, H., Bolukbasi, T., Saligrama, V.: Zero-shot detection. arXiv preprint arXiv:1803.07113 (2018)
62. Zitnick, C.L., Dollár, P.: Edge boxes: locating object proposals from edges. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 391–405. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_26