Beyond Attributes: High-order Attribute Features for Zero-shot Learning

Xiao-Bo Jin[1*] Guo-Sen Xie[2] Kaizhu Huang[1] Jianyu Miao[3]
Qiufeng Wang[1]
[1]Xi'an Jiaotong-Liverpool University, China [2]Inception Institute of Artificial Intelligence, UAE
[3]Henan University of Technology, China [*]Xiaobo.Jin@xjtlu.edu.cn

Abstract

In this paper, SeeNet with the high-order attribute features (SeeNet-HAF) is proposed to solve the challenging zero-shot learning (ZSL) task. The high-order attribute features aims to discover a more elaborate, discriminative high-order semantic vector for each class and can distill the correlation between the class attributes embedding into modeling. SeeNet-HAF consists of two branches. The upper stream is capable of dynamically localizing some discriminative object region, and then the high-order attribute supervision is incorporated to characterize the relationship between the class attributes. Meanwhile, the bottom stream discovers complementary object regions by erasing its discovered regions from the feature maps. In addition, we propose a fast hyperparameter search strategy. It takes both the breadth and precision of the search into account. Experiments on four standard benchmark datasets demonstrate the superiority of the SeeNet-HAF framework.

1. Introduction

The zero-shot learning (ZSL) task, which was first proposed in [40, 32] as a popular problem, is currently regaining widespread attention [2, 55, 4]. In contrast to supervised classification tasks, where the label set of the test images is the same as that of the training images, the label sets of the training and test images are disjoint with each other in ZSL, e.g., given images of zebras and tigers for training, while the test images are of giraffes. To make ZSL possible, the descriptions w.r.t. the training/test classes should be collected, where it is desirable that some common information (concepts), such as attributes [15], are extracted and served as the bridge for connecting the training and test classes. Other widely used descriptions include word2vector [48] and sentences [44]. Among these descriptions, attribute is the most widely used one. In this paper, we leverage attribute descriptions for evaluation.

By further projecting these descriptions onto the semantic space, we can obtain the semantic vector of each class, and then the semantic vectors serve as the prototype for subsequent classification on test images. A typical scenario for ZSL is thus focusing on establishing the correlation between the training/test class images and the corresponding semantic vectors. To learn this image-semantic mapping (embedding), existing works generally design a complex optimization objective equipped with various regularizations. This series of representative methods are based on matrix optimization [28, 58, 36, 62, 61, 45, 29, 42]. Moreover, motivated by the success of convolutional neural network (CNN) models [21] on the ImageNet [30] classification task, some recent approaches have turned to CNN models to find solutions for ZSL. Li et al. [34] proposed adopting Zoom-Net [18] for discovering the global object bounding box, and other CNN-based methods [38, 34, 13, 17, 59] also take the global images as input. In addition, some specific network regularizations, such as semantically consistent regularization [38], are incorporated into the CNN training phase.

Most ZSL methods learn a projection function from a visual feature space to a semantic embedding space using a training set. Such processes can be divided into three groups: (1) learning a projection function from a visual feature space to a semantic space by a regression or ranking method [31, 4, 48, 17, 26]; (2) choosing the reverse projection direction, such as from the semantic space to the visual feature space [47, 28]; and (3) learning an intermediate space onto which both the visual feature and the semantic space are projected [62, 10].

For the first type of approach, semantic output code (SOC) classifier [40] searches the nearest class embedding vector after mapping the image features into the semantic space. Attribute label embedding (ALE) [3] introduces a function that measures the compatibility between an image and a label embedding. Deep visual semantic embedding (DeViSE) [17] presents a deep visual-semantic embedding model trained to identify visual objects, where the semantic information can be exploited to achieve reasonable predictions. Structured joint embedding (SJE) [4] learns a compatibility function such that matching embeddings are as-

signed a higher score than mismatching embeddings. Embarrassingly simple ZSL (ESZSL) [45] uses a square loss with L_2 regularization to learn the bilinear form on the visual features and the class attributes. Bucher et al. [9] optimizes a metric discriminating capacity and accuracy attribute prediction, both of which associate two types of sub-task constraints. Semantic auto-encoder (SAE) [29] presents a semantic auto linear encoder to regularize the model by enforcing the reconstruction from the image feature space into the semantic space.

For the second type of approach, zero-shot learning through cross-modal transfer (CMT) [48] uses a neural network with two hidden layers to learn a non-linear projection from the image feature space to the word2vec space. Latent embedding method (LatEm) [53] extends the learning of a single bilinear map to a collection of maps with the selection by introducing a latent variable for the current image-class pair. Ba et al. [6] use text features to predict the output weights of both the convolutional and fully connected layers. Deep embedding model (DEM) [59] regard the visual space as the embedding space instead of embedding into a semantic space. Changpinyo et al. [11] utilize the clustering structure in the semantic embedding space by imposing a structural constraint.

For the final type of approach, ZSL via semantic similarity embedding (SSE) [62] views each source or target data as a mixture of observed class proportions and assumes that the mixture patterns from the same unseen class should be similar. Joint latent similarity embedding (JLSE) [61] develop a joint discriminative learning framework based on dictionary learning to jointly learn the model parameters in both the source and target domains. Synthesized classifiers (SYNC) [10] aligns the semantic space to the model space and introduces a set of "phantom" object classes that live in both spaces. In our previous work [56], we propose an attentive region embedding network to adapt it into ZSL task.

Although most of current work is capable of transferring the model from the seen classes to the unseen classes according to the given class attribute semantic space, there is no practical guarantee that the dimension correlations of the class attributes can be effectively captured with current optimization techniques. In this paper, we propose a new architecture for the ZSL problems by integrating the high-order feature attributes, which can distill the correlations between the classes attributes embedding into modeling. Specifically, as shown in Figure 1, we propose an end-to-end ZSL framework with the high-order attribute features (ZSL-HAF), which is designed based on user-defined attributes. ZSL-HAF aims to discover a more elaborate, diverse and discriminative high-order semantic vector for each class under the framework of the self-erasing network (SeeNet). The construction of the high-order semantic vector is simple yet effective. Specifically, given an input semantic vector $\boldsymbol{x} \in R^{C\times 1}$ (quantized from attributes), we first calculate the high-order correlation matrix as $M = \boldsymbol{x} \times \boldsymbol{x}^T \in R^{C\times C}$; then, Gaussian random projection (GRP) is leveraged to project M onto the high-order attribute space (Figure 2). It explicitly captures the pairwise correlations between the embedding dimensions and represent their high-order dimension correlations. Finally, We propose a fast hyperparameter search strategy. It takes both the breadth and precision of the search into account.

2. Related Works

Zero-shot Learning. The direct attribute prediction (DAP) model, a seminal work for ZSL, was proposed by Lampert et al. [32]. In DAP, the probabilistic attribute classifiers are first learned for each attribute, and then the posteriors of the test classes are calculated for a given image. The final class is obtained by maximizing the posterior estimation. Meanwhile, a multi-class classifier is trained on seen classes for indirect attribute prediction (IAP) [32]. According to the scores of these seen classes, the attribute posteriors are determined. Both DAP and IAP ignore the correlations between different attributes, and a random forest approach was further introduced by [24].

For latent attribute learning, only several linear transformation methods exist, including joint learning of semantic and latent attributes (JSLA) [41], LDF [34] and latent attribute dictionary (LAD) learning [25], all of which are obtained by directly/indirectly regulating the inter-class and intra-class distances, and they are first-order attribute methods.

Generalized ZSL. If images from both seen and unseen classes are considered during the testing phase, ZSL becomes generalized ZSL (GZSL), as first proposed by [46]. Then, a new split for the training and test data for GZSL was proposed by [55]. Following the new split, samples from both seen and unseen classes are utilized to conduct GZSL evaluation [64, 27]. Many work [23, 33] focus on taking advantage of generative adversarial network to assist in completing GZSL tasks.

Adversarial Erasing Learning. Adversarial erasing aims to discover irregular object locations, and it was first proposed in [52] for semantic segmentation tasks and has been successfully applied to related fields, such as object detection [60]. Motivated by the ability of adversarial erasing learning for discovering irregular objects, we adopt adversarial erasing to leverage ZSL, which is the first attempt to use erasing learning for ZSL.

3. Proposed Approach

We are given a set of source classes $C_S = \{l_1, l_2, \cdots, l_s\}$ and N labelled source samples $D = \{(\boldsymbol{I}_i, y_i)\}_{i=1}^{N}$ for training, where $\boldsymbol{I}_i$ is the i-th training im-

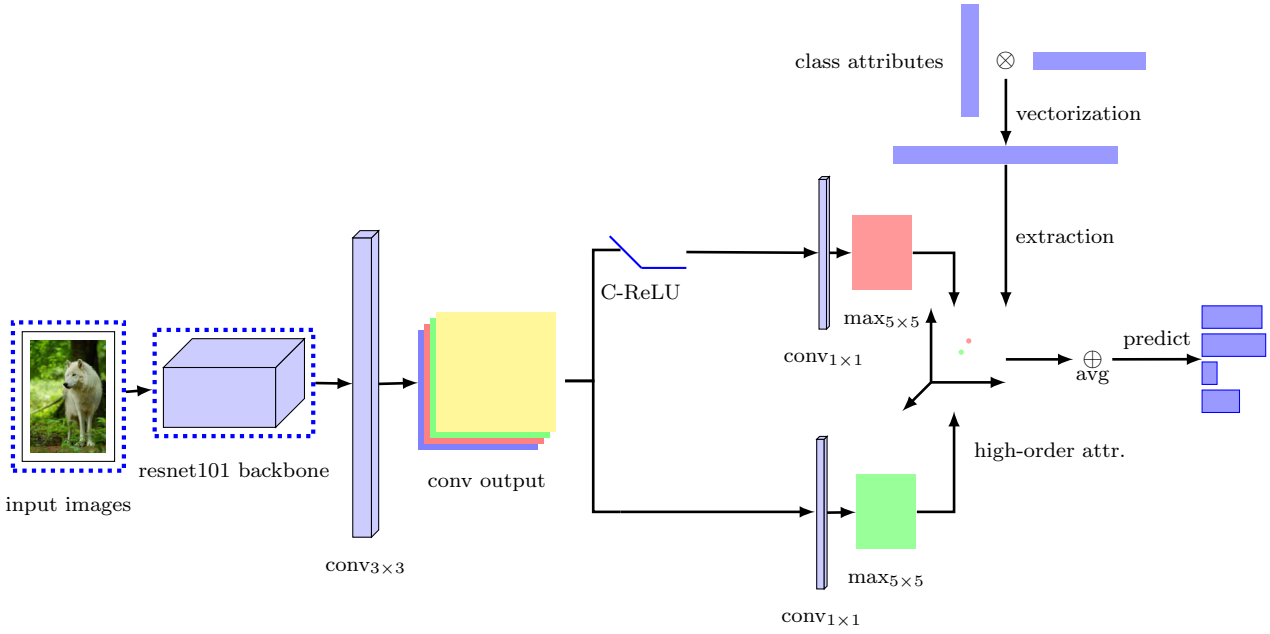

Figure 1: Overview of the proposed SeeNet-HAF approach. SeeNet consists of two branches after a shared backbone network (e.g. resnet101). The structure of the lower branch is a convolutional network with the size 1×1 followed by a maximum pooling layer, and the upper branch is similar to that of the bottom one, except a C-ReLU la yer which is inserted in front. The outputs of both branches are projected onto the space spanned by the high-order attributes.

age and $y_i (y_i \in C_S)$ is its label. Given a new test image $\boldsymbol{I}_j$, the goal of ZSL is to assign it to an unseen class label from $C_U = \{l_{s+1}, \cdots, l_{s+u}\}$. Note that the label sets from the training (seen) classes and the test (unseen) classes are disjoint from each other, i.e., $C_S \cap C_U = \phi$. Each class label y (both seen/unseen classes) is associated with a predefined semantic vector $\varphi(y)$.

3.1. Self-Erasing Network Embedding

Self-erasing network (SeeNet) [22] is an extension of class activation maps (CAM) [63], where fully connected layers can aggregate the features of the last convolutional layer for localization purposes. SeeNet contains two branches: One branch dynamically localizes some discriminative object region; the other branch discovers complementary object regions by erasing its discovered regions from the feature maps, which can assist the ZSL tasks; thus, we embed SeeNet for ZSL tasks, which is an end-to-end network framework (SeeNet-ZSL).

SeeNet (as shown in Figure 1) consists of two branches after a shared backbone network (e.g., ResNet101). The structure of the lower branch is a convolutional network with a size of 1×1 followed by a maximum pooling layer, and the upper branch is similar to that of the bottom one, except for a C-ReLU layer that is inserted in front.

We consider a fully convolutional network (FCN) and denote the last convolutional feature maps as $S_{K \times H \times H}$, where $H \times H$ is the spatial size and K is the number of channels. Given the feature map S, we add a convolutional layer of C channels with the kernel size of 1×1, stride 1 on top of the feature maps S. We aggregate the feature map S with C groups of weights to obtain C weighted feature maps $W_{k,c}$ called the localization map $L_c, c = 0, 1, \cdots, C-1$, which can be computed as

$$L_c = \sum_{k=0}^{K-1} S_k \cdot W_{k,c}, \tag{1}$$

where S_k is the k-th channel of a feature map with a size of $H \times H$. The above localization can be implemented by a convolutional layer with a kernel size of 1×1 (see $\text{conv}_{1\times1}$ unit of Figure 1).

As shown in the red block diagram in Figure 1, we introduce the erase operation to learn to highlight the attention map, where the C-ReLU [22] function merges a binary mask with the ReLU function. C-ReLU is defined as

$$\text{C-ReLU}(x) = \max(x, 0) \cdot \theta_\delta(x), \tag{2}$$

where $\theta_\delta(x)$ is a binary mask: $\theta_\delta(x) = 1$ if $x \geq \delta$, and $\theta_\delta(x) = -1$ otherwise. In our work, we set a parameter δ_k for each channel S_k $(k = 0, 1, \cdots, K-1)$ of the feature map.

3.2. Extraction of High-order Attribute Features

Most previous works on learning latent attributes in ZSL focus on the class attribute itself or its linear/non-linear transformation, such as the form of two-layer neural network.

However, in many vision tasks, the relationship between the class attributes carries the relevant information, which is helpful for ZSL. We use the outer product to encode the relations between the class attributes as (predefined semantic vector as $\varphi(y)$)

$$L_y = \text{vec}\left(\varphi(y) \cdot \varphi(y)^T\right). \tag{3}$$

Each element in the matrix $\varphi(y) \cdot \varphi(y)^T$ will constitute evidence for exactly one type of shift and detect the coincidences [35], acting as AND-gates (Figure 2).

For faster processing times and smaller model sizes, we need an efficient way to remove the unimportant attribute relations. Random projections show appealing properties of preserving the distance quite well. The projection onto a random lower-dimensional subspace yields comparable results to PCA but with less computational expense [8]. The original d-dimensional data use a random $r \times d$ matrix W^{RP} whose rows have unit lengths.With the projection matrix W^{RP}, the input is mapped onto r dimensions of subspace with a time complexity of $O(rdn)$. GRP [1] projects the original input X onto the reduced subspace with the random matrix, whose components are selected from the Gaussian distribution $N(0, 1/r)$.

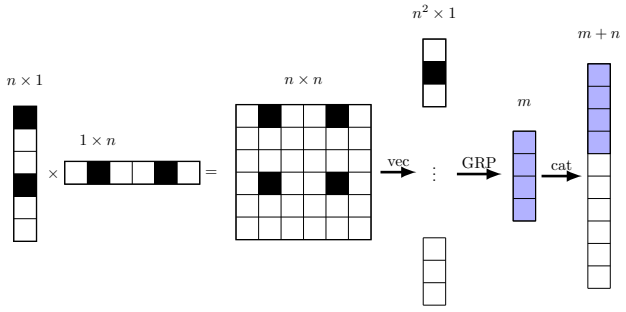

Figure 2: Merge of the high-order and original class attributes: the outer product of the class attribute is vectorized by row and then project into a reduced space to obtain a compact high-order representation.

3.3. SeeNet with High-order Attribute Features (SeeNet-HAF)

After the class activation map in both of the branches ($\text{conv}_{1\times1}$ in Figure 1), we add 5×5 max pooling layers (the green square in Figure 1), and then we project them into the new class attribute semantic space.

Our ZSL model aims to learn the relation between the visual feature space and the semantic space. Formally,

$$F(\boldsymbol{I}_i; W) = \phi(\boldsymbol{I}_i)^T W \varphi(y) \tag{4}$$

where W is a linear projection matrix to learn in a fully connected layer and $\phi(\boldsymbol{I}_i)$ is the deep learning representation of the image $\boldsymbol{I}_i$. It is similar to the classification score in traditional object recognition tasks, where the sum of the cross-entropy loss of two branches can be used as the compatibility loss function.

At the test stage, an unseen image I_u can be assigned to the most matched class $y^* \in C_U$

$$y^* = \underset{l \in C_U}{\arg\max}\, \phi(\boldsymbol{I}_u)^T W \varphi(l) \tag{5}$$

Discussion. Learning of high-order feature was shown to yield good results in a variety of recognition and classification tasks [43]. Joint recurrent learning of context and correlation [51] is proven to improve attribute recognition given some sized training data with bad quality images. An alternative interpretation for why the inclusion of higher-order features works well is, that they are better at representing real-valued data. The learning of higher-order features amounts to learning on a basis-expansion of the feature inputs [43]. The hierarchy and exclusion graphs [14] allows encoding of flexible relations between labels, especially in the case of the overlap and subsumption of the labels. The raw classification attributes is important and noisy, but there is few work to handle it. In our work, the low-dimensional pre-projections of the class attributes can be defined naturally to reflect the correlations between the attributes.

4. Experiments

4.1. Datasets and settings

Datasets. We select two fine-grained (CUB and SUN) and two coarse-grained datasets (AWA2 and aPY).

CUB (Caltech-UCSD Birds-200-2011) is a medium-scale dataset with respect to the number of classes and images. We follow the class split of CUB with 150 training (50 validation classes) and 50 test classes. **SUN** contains 14340 images from 717 types of scenes annotated with 102 attributes, where 645 classes (65 classes for validation) are chosen for training and 72 classes are chosen for testing. **AwA2** contains 37,322 images of the same 50 classes of animals for training (13 classes for validation) and another 10 classes for testing, which is an extension of **AwA1** . Finally, **aPY** contains 32 classes with 64-dimensional attribute vectors, including 20 Pascal classes for training and 12 Yahoo classes for testing.

Implementation details. We conduct the experiments under two types of ZSL settings, including the standard splitting (SS) and the proposed splitting (PS). In addition,

we also provide the results in the generalized ZSL, where the test samples may come from either the training classes or test classes.

For aPY, we crop the images from bounding boxes because there are multiple objects in each image. Our image embedding vectors correspond to the 2048-dimensional top-layer pooling units of the ResNet-101 network. We use the original ResNet-101 that is pre-trained on ImageNet with 1000 classes. Most of the previous ZSL methods adopt fixed pre-trained features, but we believe that it is inappropriate to regulate the image representation with fixed image features. In general, an end-to-end framework will lead to better performance [59]. We initialize the final fully connected linear layer with the attribute matrix and fix them during the training process.

SGD is used to optimize our model with a minibatch size of 64. An initial learning rate is randomly taken from the real range $[0.0001, 0.01]$. For our SGD algorithm, we use the cyclic learning rate strategy, where the starting cycle is set to 10 epochs and then multiplied by a factor 2 ($T_{mul} = 2$). Other training parameters, such as the dropout rate, momentum and weight decay, are set to 0.4, 0.9 and 0.0005, respectively. For the threshold used in the erase network, we set the threshold δ to ξ times the maximum value of each channel of the attention map input to the C-ReLU layer, where ξ is taken from the range $[0.001, 0.1]$. For the extraction of high-order features, we set the reduced dimension to γ times the dimension of the original attributes, where γ is a float number chosen from $\{0.3, 4\}$.

4.2. Fast hyperparameter search

Random search [7] is able to find models that are as good as those found by the grid search but with less computational cost. For each configuration, the training of deep learning on large-scale datasets is the main computational bottleneck: several days are often required to obtain reasonable results.

The cyclic learning rate [37] can help us achieve better performance and a faster convergence rate than the constant learning rate within few epochs. In the following, we use the cyclic learning rate strategy to search for a best parameter for the ZSL problems, which simulates a new restart of SGD after T_i epochs are implemented. During T_i epochs, the learning value is varying from its maximum to minimum (e.g., 0). Formally, the learning rate with a cosine annealing is computed as

$$\alpha = \frac{\alpha_{max}}{2}\left(1 + \cos\left(\frac{T_{cur}}{T_i}\pi\right)\right), \tag{6}$$

where α_{max} is the max learning rate, and T_{cur} is accumulating epochs from the last restart. Note that each batch has its own learning rate since T_{cur} is updated during each batch iteration. Meanwhile, we increase T_i by a factor of T_{mul} at every restart.

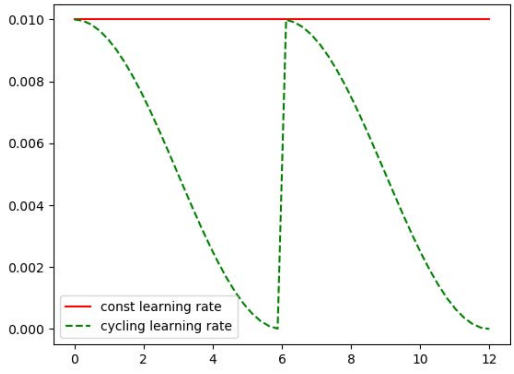

Figure 3: Cycling learning rate and const learning rate

Given a large group of candidate parameters (e.g., 100) randomly chosen from a user-defined range, we run one epoch for each candidate parameter. According to the performance on the validation dataset, we select the top ten parameter configurations and run ten epochs to choose the best parameter configuration from these ten groups. Finally, we report the final results by running another 30 epochs on the test dataset.

Figure 4 presents a comparison in terms of the accuracy in the first ten epochs for the constant learning rate and the cyclic learning rate with different configurations. The length and multiplier of the cycle vary from $\{2, 10\}$ and $\{1, 1.1, 1.5, 2\}$, respectively. After three epochs, the constant learning rate begins to catch up with the cyclic learning rate. However, at the 6th epoch, the cyclic learning rate surpasses the constant learning rate. In practice, the increasing period may slow the decay speed of the learning rate. As shown in Figure 4, we can obtain the best performance with the cycle multipliers 2 and 1.5. Our proposed algorithm achieves the highest accuracy in the case of $cycle\_len = 10$ and $cycle\_mul = 2$, which verifies that it is a good empirical setting in the deep learning [37].

Table 1 presents the experimental results, from which, we can conclude that the cycling learning rate (SeeNet-HAF) achieves better performance than the constant learning rate (SeeNet-HAF*) on multiple dataset splits. For example, SeeNet-HAF obtains 72.2% on CUB-PS, which has improved SeeNet-HAF* up to 5%. In other cases, SeeNet-HAF model performs slightly worse than SeeNet-HAF* on both of AWA2-PS and aPY-PS, which shows that using a simple constant learning rate on these datasets is enough to search a good model.

In summary, our search strategy takes both the breadth and precision of the search into account. It gradually nar-

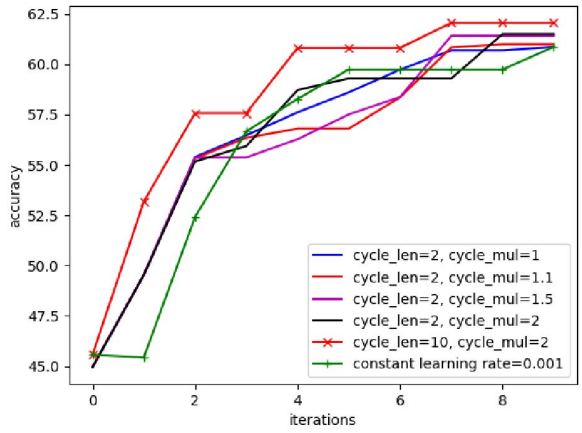

Figure 4: Comparisons between the step learning rate and the cycling learning rate where the initial learning rate is 0.001 with the settings of different cycle lengths and the cycle multipliers

rows the scope of the search and improves the precision of the search during the search process.

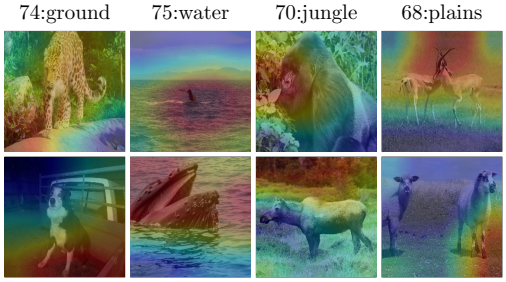

Figure 5: Class attributes activation map of AWA2 dataset (numbering the class attributes from zero): the maps highlights the object regions related to the class attributes, e.g. ground, water, jungle and plains.

4.3. Comparisons with benchmarks

To demonstrate the effectiveness of our SeeNet-HAF and SeeNet-ZSL, we compare them with dozens of existing ZSL methods in Tables 1 and 2, among which the results of 13 methods are the baselines reported in [55].

Comparisons in conventional ZSL. In the conventional ZSL setting, we follow the experiment and evaluation protocol as in [55] and report the results on four benchmarks for both the standard split (SS) and the proposed split (PS). The first 13 baselines are from [55], and the next two are taken from [13, 5]. We obtain our results following identical settings for the fairness of comparisons. As shown, our SeeNet-ZSL and SeeNet-HAF algorithms outperform other

Table 1: Zero-shot learning results on SUN,CUB,AWA2 and aPY. SS = Standard Split, PS = Proposed Split. The best result is marked in red and the second best in blue.

Method	SUN		CUB		AWA2		aPY	
	SS	PS	SS	PS	SS	PS	SS	PS
DAP [31]	38.9	39.9	37.5	40.0	58.7	46.1	35.2	33.8
IAP [31]	17.4	19.4	27.1	24.0	46.9	35.9	22.4	36.6
CONSE [39]	44.2	38.8	36.7	34.3	67.9	44.5	25.9	26.9
CMT [48]	41.9	39.9	37.3	34.6	66.3	37.9	26.9	28.0
SSE [62]	54.5	51.5	43.7	43.9	67.5	61.0	31.1	34.0
LATEM [53]	56.9	55.3	49.4	49.3	68.7	55.8	34.5	35.2
ALE [3]	59.1	58.1	53.2	54.9	80.3	62.5	30.9	39.7
DEVISE [17]	57.5	56.5	53.2	52.0	68.6	59.7	35.4	39.8
SJE [4]	57.1	53.7	55.3	53.9	69.5	61.9	32.0	32.9
ESZSL [45]	57.3	54.5	55.1	53.9	75.6	58.6	34.4	38.3
SYNC [10]	59.1	56.3	54.1	55.6	71.2	46.6	39.7	23.9
SAE [29]	42.4	40.3	33.4	33.3	80.7	54.1	8.3	8.3
GFZSL [50]	62.9	60.6	53.0	49.3	79.3	63.8	51.3	38.4
PSR [5]	—	61.4	—	56	—	63.8	—	38.4
SP-AEN [13]	—	59.2	—	55.4	—	58.5	—	24.1
QFSL$^-$ [29]	58.9	56.2	58.5	58.8	72.6	63.5	—	—
DEM [59]	—	40.3	—	51.7	—	67.1	—	35.0
LDF [34]	—	—	67.1	—	83.4	—	—	—
RN [57]	—	—	—	55.6	—	64.2	—	—
UDA [28]	—	—	39.5	—	—	—	—	—
TMV [19]	61.4	—	51.2	—	—	—	—	—
SMS [20]	60.5	—	59.2	—	—	—	—	—
QFSL [49]	61.7	58.3	69.7	72.1	84.8	79.7	—	—
AREN [56]	61.7	60.6	70.7	71.8	86.7	67.9	44.1	39.2
SeeNet-ZSL	61.5	60.1	70.8	73.5	81.5	64.4	43.2	37.2
SeeNet-HAF*	59.8	58.5	66.1	67.2	82.7	67.9	44.9	38.7
SeeNet-HAF	63.5	62.5	68.4	72.2	87.1	67.2	45.7	38.3

*: SeeNet-HAF with the common learning rate strategy.

state-of-the-art algorithms on most datasets. For example, SeeNet-ZSL outperforms SYNC by 16.7% on the SS split of the CUB dataset (CUB-SS), where SYNC achieves the best result among the compared methods. In addition, in the PS split of the CUB dataset (CUB-PS), SeeNet-ZSL surpasses the PSR algorithm by 17.3%. On the AWA2 dataset, SeeNet-HAF exceeds the best results by 6.4% and 3.4% for SS and PS, respectively. In CUB and SUN datasets, our inductive methods are on par with and even overpass the leading transductive methods such as QFSL. These results demonstrate that for image recognition in a complex background, the extraction of a irregular segmentation discriminating region is very beneficial for migrating from the training classes to the test ones.

When exploring the effects of the high-order class attributes, we find that the simple off-line-extracted high-order attributes help further improve our algorithm by 2% in most cases. With an exception, we achieve an approximately 6% increase on the SS split of the AWA2 dataset when comparing SeeNet-HAF with SeeNet-ZSL. We also observe that there is a slight performance decrease on the CUB dataset, which may be attributed to the images in CUB containing a single object and simple background, and there may be no such interaction between the class attributes. From the above analysis, we verify the validity of the high-order attributes for ZSL problems.

Comparisons in generalized ZSL. The training accu-

Table 2: Generalized Zero-Shot Learning on Proposed Split (PS) measures including the training accuracy, test accuracy and harmonic mean. CS means the Calibrated Stacking approach. The best number is marked in bold.

Method	SUN			CUB			AWA2			aPY		
	tr	te	H	tr	te	H	tr	te	H	tr	te	H
DAP [31]	4.2	25.1	7.2	1.7	67.9	3.3	0.0	84.7	0.0	4.8	78.3	9.0
IAP [31]	1.0	37.8	1.8	0.2	**72.8**	0.4	0.9	87.6	1.8	5.7	65.6	10.4
CONSE [39]	6.8	39.9	11.6	1.6	72.2	3.1	0.5	90.6	1.0	0.0	**91.2**	0.0
CMT [48]	8.1	21.8	11.8	7.2	49.8	12.6	0.5	90.0	1.0	1.4	85.2	2.8
SSE [62]	2.1	36.4	4.0	8.5	46.9	14.4	8.1	82.5	14.8	0.2	78.9	0.4
LATEM [53]	14.7	28.8	19.5	15.2	57.3	24.0	11.5	77.3	20.0	0.1	73.0	0.2
ALE [3]	21.8	33.1	26.3	23.7	62.8	34.4	14.0	81.8	23.9	4.6	73.7	8.7
DEVISE [17]	16.9	27.4	20.9	23.8	53.0	32.8	17.1	74.7	27.8	4.9	76.9	9.2
SJE [4]	14.7	30.5	19.8	23.5	59.2	33.6	8.0	73.9	14.4	3.7	55.7	6.9
ESZSL [45]	11.0	27.9	15.8	12.6	63.8	21.0	5.9	77.8	11.0	2.4	70.1	4.6
SYNC [10]	7.9	43.3	13.4	11.5	70.9	19.8	10.0	90.5	18.0	7.4	66.3	13.3
SAE [29]	8.8	18.0	11.8	7.8	54.0	13.6	1.1	82.2	2.2	0.4	80.9	0.9
GFZSL [50]	0.0	39.6	0.0	0.0	45.7	0.0	2.5	80.1	4.8	0.0	83.3	0.0
PSR [5]	20.8	37.2	26.7	24.6	54.3	33.9	20.7	73.8	32.3	13.5	51.4	21.4
SP-AEN [13]	—	—	24.9	—	—	34.7	—	—	23.3	—	—	13.7
DEM [59]	20.5	34.3	25.6	19.6	57.9	29.2	30.5	86.4	45.1	11.1	75.1	19.4
RN [57]	—	—	—	38.1	61.4	47.0	30.0	**93.4**	45.3	—	—	—
QFSL [29]	30.9	18.5	23.1	33.3	48.1	39.4	52.1	72.8	60.7	—	—	—
f-CLSWGAN [54]	42.6	36.6	39.4	43.7	57.7	49.7	57.9	61.4	59.6	—	—	—
cycle-CLSWGAN [16]	49.4	33.6	40.0	45.7	61.0	52.3	59.6	63.4	59.8	—	—-	—
cycle-(U)WGAN [16]	47.2	33.8	39.4	47.9	59.3	53.0	59.6	63.4	59.8	—	—	—
AREN [56]	40.3	32.3	35.9	63.2	69.0	66.0	54.7	79.1	64.7	30.0	47.9	**36.9**
SeeNet-ZSL + CS	33.1	40.7	36.5	72.5	64.7	**68.4**	81.6	52.5	63.9	51.0	27.0	35.4
SeeNet-HAF + CS	33.8	**46.4**	**39.1**	67.9	62.9	65.3	82.0	55.6	**66.3**	**55.0**	26.5	35.7

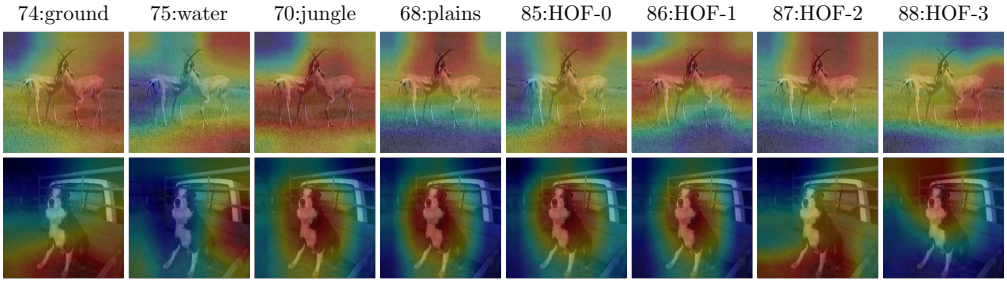

Figure 6: First-order and high-order class attribute activation maps of the AWA2 dataset: the class attribute before and after 85 is the first-order and the high-order class attributes, respectively. We can see that the higher-order and first-order attributes complement each other. The high-order attributes also can guide the convolution map to find the discriminant region where the first-order attribute may ignore.

racy, test accuracy and their harmonic mean [55] is taken as the evaluation criterion for the model comparisons under GZSL settings.

We observe that the output in the training and test classes is not comparable. When the output from the training classes dominates, the classification performance of the training classes is higher than that of the test classes, and vice versa. We argue that the performance of the training classes is not necessarily better than one of the test classes, which can be found in Chao's work [12].

We follow the settings of the generalized ZSL problem [55] to report the results with the trained model on the PS split of four datasets. We find that the classification performance is biased in the training and test classes for the listed algorithms. The reason for this phenomenon is that no instances from the test classes are observed during the

Figure 7: Average attribute activation map of SeeNet-HAF on the AWA2 dataset: the images in the upper row and the lower row is the original ones and its attention maps. We can see that our approach is able to discover the irregular segmentation discriminative region of the object.

training process; thus, the outputs of the training and test classes are independent of each other and not comparable during the testing stage.

With the calibration stacking (CS) strategy [12], we can well overcome the bias of the mode outputs on the training and test classes. We observe that the harmonic accuracy of our algorithms is greatly improved. On the CUB dataset, the harmonic accuracy increases from 44.8% to 68.4%. As another example, the harmonic accuracy of SeeNet-HAF has been greatly improved from 6.7% to 66.3%. Of course, with this strategy, the harmonic accuracy of our algorithm is far greater than the other algorithms listed in the table. Due to the unavailability of codes for some compared methods and space limitation, we only conducted CS on SeeNet-ZSL and SeeNet-HAF.

4.4. Attention of the SeeNet-HAF algorithm

The 1×1 convolutional layer generates maps with d channels, where d is the dimension of the class attributes. We sample some images from the AWA2 dataset and visualize the attention map related to some attributes to obtain a class attributes activation map (Figure 5). It is surprising in the ZSL problem that our SeeNet-HAF can relate the semantic objects of the image to the corresponding class attributes. For example, in Figure 5, the ground where the tiger and the dog stand and the plains where the antelopes and sheep live are marked as deeper red (attention regions). However, before training, we do not associate the position of the specific attribute of the image with the class attribute. We only use the text attribute to describe whether there is such an attribute in the image or how likely it possesses such an attribute. Our algorithm is able to accurately mark the locations of the class attributes in the image, which will aid us in deeply understanding how our ZSL algorithm works.

To investigate how the high-order attribute activates the feature map, we show the comparison of the feature activation map of the first-order and high-order attributes on two images in Figure 6. As shown, the high-order features focus on different parts of the image, and these parts may be ignored by the first-order features. To some extent, higher-order features complement and enhance the effects of the first-order features.

Finally, we weight the feature maps corresponding to the class attributes to obtain the average activation map of the class attributes, as shown in Figure 7, where the weights of the class attributes are softmax values [63] of the class attributes matrix. As shown, SeeNet-HAF can accurately find the discriminating area of the target. For example, for the rhinoceros (the 5th picture in the image), we identify whether an animal is a rhinoceros or not through its mouth rather than the body; thus, the colour of the head of the rhinoceros appears deeper than the body in Figure 7. The rightmost picture in the image shows that an adult is holding a horse on which a little girl is riding. SeeNet-HAF deepens the colour of the first half of the horse rather than the little girl or the adult because the class of the image is labelled as a horse.

5. Conclusions

In this paper, an adversarial erasing embedding network guided by high-order attributes (SeeNet-HAF) is proposed to solve the challenging ZSL/GZSL task. The high-order attribute features can distill the correlations between the class attributes embedding into modeling, which is simple yet effective to compute. To the best of our knowledge, this work is the first to seriously consider the high-order features for the predefined class attributes. SeeNet-HAF consists of two branches. The upper stream is capable of erasing some initially discovered regions, and then the high-order attributes followed by Gaussian random projection is incorporated to represent the relationship between the class attributes. Meanwhile, the bottom stream is trained by using the current background regions to train the same attribute. We propose a fast hyperparameter search strategy. It takes both the breadth and precision of the search into account. A class attribute activation map is proposed to visually show the relationship between the class attribute features and attention map. Experiments on four standard benchmark datasets demonstrate the superiority of the SeeNet-HAF framework.

References

[1] D. Achlioptas. Database-friendly random projections: Johnson-lindenstrauss with binary coins. 66(4):671–687. 4

[2] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for attribute-based classification. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 819–826. 1

[3] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for image classification. 38(7):1425–1438. 1, 6, 7

[4] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele. Evaluation of output embeddings for fine-grained image classification. pages 2927–2936. 1, 6, 7

[5] Y. Annadani and S. Biswas. Preserving semantic relations for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7603–7612. 6, 7

[6] J. L. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *ICCV 2015*, pages 4247–4255. 2

[7] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. 13:281–305. 5

[8] E. Bingham and H. Mannila. Random projection in dimensionality reduction: applications to image and text data. pages 245–250. ACM Press. 4

[9] M. Bucher, S. Herbin, and F. Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *ECCV 2016*, Lecture Notes in Computer Science, pages 730–746. Springer International Publishing. 2

[10] S. Changpinyo, W.-L. Chao, B. Gong, and F. Sha. Synthesized classifiers for zero-shot learning. pages 5327–5336. 1, 2, 6, 7

[11] S. Changpinyo, W.-L. Chao, and F. Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *ICCV 2017*. 2

[12] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *ECCV 2016*. 7, 8

[13] L. Chen, H. Zhang, J. Xiao, W. Liu, and S.-F. Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. 1, 6, 7

[14] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-scale object classification using label relation graphs. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 48–64. Springer International Publishing. 4

[15] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE. 1

[16] R. Felix, B. G. V. Kumar, I. Reid, and G. Carneiro. Multi-modal cycle-consistent generalized zero-shot learning. 7

[17] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. DeViSE: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems 26*, pages 2121–2129. 1, 6, 7

[18] J. Fu, H. Zheng, and T. Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *CVPR*, volume 2, page 3. 1

[19] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong. Transductive multi-view zero-shot learning. 37(11):2332–2345. 6

[20] Y. Guo, G. Ding, X. Jin, and J. Wang. Transductive zero-shot recognition via shared model space learning. In *AAAI*, volume 3, page 8. 6

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. 1

[22] Q. Hou, P.-T. Jiang, Y. Wei, and M.-M. Cheng. Self-erasing network for integral object attention. 3

[23] H. Huang, C. Wang, P. S. Yu, and C.-D. Wang. Generative dual adversarial network for generalized zero-shot learning. 2

[24] D. Jayaraman and K. Grauman. Zero-shot recognition with unreliable attributes. In *Advances in neural information processing systems*, pages 3464–3472. 2

[25] H. Jiang, R. Wang, S. Shan, Y. Yang, and X. Chen. Learning discriminative latent attributes for zero-shot classification. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4233–4242. 2

[26] X.-B. Jin, G.-G. Geng, G.-S. Xie, and K. Huang. Approximately optimizing NDCG using pair-wise loss. 453:50–65. 1

[27] X.-B. Jin, G.-S. Xie, K. Huang, H. Cao, and Q.-F. Wang. Discriminant zero-shot learning with center loss. 2

[28] E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Unsupervised domain adaptation for zero-shot learning. In *ICCV 2015*, pages 2452–2460. 1, 6

[29] E. Kodirov, T. Xiang, and S. Gong. Semantic autoencoder for zero-shot learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4447–4456. 1, 2, 6, 7

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105. 1

[31] C. H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. 36(3):453–465. 1, 6, 7

[32] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 951–958. IEEE. 1, 2

[33] J. Li, M. Jin, K. Lu, Z. Ding, L. Zhu, and Z. Huang. Leveraging the invariant side of generative zero-shot learning. 2

[34] Y. Li, J. Zhang, J. Zhang, and K. Huang. Discriminative learning of latent features for zero-shot recognition. 1, 2, 6

[35] T.-Y. Lin, A. RoyChowdhury, and S. Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 1449–1457. 4

[36] Y. Long, L. Liu, F. Shen, L. Shao, and X. Li. Zero-shot learning using synthesised unseen visual data with diffusion regularisation. pages 1–1. 1

[37] I. Loshchilov and F. Hutter. SGDR: Stochastic gradient descent with warm restarts. 5

[38] P. Morgado and N. Vasconcelos. Semantically consistent regularization for zero-shot recognition. In *CVPR*, volume 9, page 10. 1

[39] 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*. 6, 7

[40] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 2009*, pages 1410–1418. 1

[41] P. Peng, Y. Tian, T. Xiang, Y. Wang, and T. Huang. Joint learning of semantic and latent attributes. In *European Conference on Computer Vision*, pages 336–353. Springer. 2

[42] R. Qiao, L. Liu, C. Shen, and A. van den Hengel. Less is more: zero-shot learning from online textual documents with noise suppression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2249–2257. 1

[43] M. Ranzato and G. E. Hinton. Modeling pixel means and covariances using factorized third-order boltzmann machines. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2551–2558. 4

[44] S. Reed, Z. Akata, H. Lee, and B. Schiele. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–58. 1

[45] B. Romera-Paredes and P. H. S. Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, ICML'15, pages 2152–2161. JMLR.org. 1, 2, 6, 7

[46] W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult. Toward open set recognition. 35(7):1757–1772. 2

[47] Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, and Y. Matsumoto. Ridge regression, hubness, and zero-shot learning. In A. Appice, P. P. Rodrigues, V. Santos Costa, C. Soares, J. Gama, and A. Jorge, editors, *ECML PKDD 2015*, Lecture Notes in Computer Science, pages 135–151. Springer. 1

[48] R. Socher, M. Ganjoo, C. D. Manning, and A. Y. Ng. Zero-shot learning through cross-modal transfer. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'13, pages 935–943. 1, 2, 6, 7

[49] J. Song, C. Shen, Y. Yang, Y. Liu, and M. Song. Transductive unbiased embedding for zero-shot learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1024–1033. IEEE. 6

[50] V. K. Verma and P. Rai. A simple exponential family framework for zero-shot learning. 6, 7

[51] J. Wang, X. Zhu, S. Gong, and W. Li. Attribute recognition by joint recurrent learning of context and correlation. 4

[52] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *IEEE CVPR*, volume 1, page 3. 2

[53] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele. Latent embeddings for zero-shot classification. In *CVPR 2016*. 2, 6, 7

[54] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata. Feature generating networks for zero-shot learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5542–5551. IEEE. 7

[55] Y. Xian, B. Schiele, and Z. Akata. Zero-shot learning - the good, the bad and the ugly. In *CVPR 2017*. 1, 2, 6, 7

[56] G.-S. Xie, L. Liu, X. Jin, F. Zhu, Z. Zhang, J. Qin, Y. Yao, and L. Shao. Attentive region embedding network for zero-shot learning. pages 9384–9393. 2, 6, 7

[57] F. S. Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. 6, 7

[58] M. Ye and Y. Guo. Zero-shot classification with discriminative semantic representation learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1

[59] L. Zhang, T. Xiang, and S. Gong. Learning a deep embedding model for zero-shot learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3010–3019. 1, 2, 5, 6, 7

[60] X. Zhang, Y. Wei, J. Feng, Y. Yang, and T. S. Huang. Adversarial complementary learning for weakly supervised object localization. page 10. 2

[61] Z. Zhang and V. Saligrama. Zero-shot learning via joint latent similarity embedding. pages 6034–6042. 1, 2

[62] Z. Zhang and V. Saligrama. Zero-shot learning via semantic similarity embedding. In *ICCV 2015*, pages 4166–4174. 1, 2, 6, 7

[63] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. 3, 8

[64] Y. Zhu, M. Elhoseiny, B. Liu, X. Peng, and A. Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2