Improving Generalized Zero-Shot Learning by Exploring the Diverse Semantics from External Class Names

Yapeng Li[1], Yong Luo[1,2], Zengmao Wang[1] *, Bo Du[1,2]*
[1]National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence,
Hubei Key Laboratory of Multimedia and Network Communication Engineering,
School of Computer Science, Wuhan University, Wuhan, China.
[2] Hubei Luojia Laboratory, Wuhan, China.

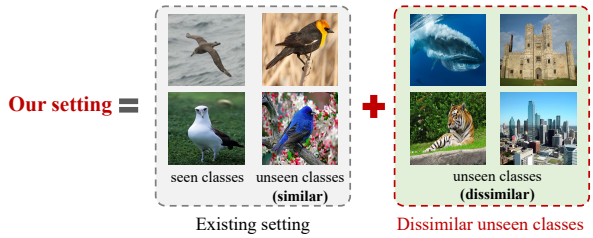
https://github.com/li-yapeng/DSECN

Abstract

Generalized Zero-Shot Learning (GZSL) methods often assume that the unseen classes are similar to seen classes, and thus perform poor when unseen classes are dissimilar to seen classes. Although some existing GZSL approaches can alleviate this issue by leveraging additional semantic information from test unseen classes, their generalization ability to dissimilar unseen classes is still unsatisfactory. This motivates us to study GZSL in the more practical setting, where unseen classes can be either similar or dissimilar to seen classes. In this paper, we propose a simple yet effective GZSL framework by exploring diverse semantics from external class names (DSECN), which is simultaneously robust on the similar and dissimilar unseen classes. This is achieved by introducing diverse semantics from external class names and aligning the introduced semantics to visual space using the classification head of pre-trained network. Furthermore, we show that the design idea of DSECN can easily be integrate into other advanced GZSL approaches, such as the generative-based ones, and enhance their robustness for dissimilar unseen classes. Extensive experiments in the practical setting including both similar and dissimilar unseen classes show that our method significantly outperforms the state-of-the-art approaches on all datasets and can be trained very efficiently.

1. Introduction

Due to the high cost of annotation and the complexity of real-world test scenarios, the presence of unseen classes is often inevitable [34, 43]. Unfortunately, traditional machine learning models are unable to handle samples from classes that have not been covered by the training data [8].



(a) Our Setting

Mode	CN [38]	TransZero [9]	ZLA [4]	DGZ [5]	Ours (Δ)
Similar	42.50	47.99	42.65	43.16	**48.45** (+0.46)
Dissmilar	5.09	2.47	8.52	12.30	**47.44** (+35.14)

(b) Results

Figure 1. Setting illustration and results: (a) Comparison with the existing setting, our setting takes into account the scenario where the unseen classes are dissimilar to seen classes; (b) Performance of existing GZSL methods on similar and dissimilar unseen classes on the CUB dataset. The similar unseen classes were obtained from the same CUB dataset, while the dissimilar unseen classes come from AWA2 and SUN dataset.

To tackle this challenge, zero-shot learning (ZSL) has been proposed to recognize new classes via transferring knowledge obtained from seen classes with the help of semantic information [14, 15, 23, 52]. In contrast, Generalized ZSL (GZSL) is a more challenging task which handle test samples from both seen and unseen classes [3, 25, 29, 34].

The core idea of GZSL is to introduce auxiliary semantic information and establish the connection between semantic and visual space for the recognition of unseen classes [34]. Since unseen samples are not available during training, existing GZSL models often misclassify unseen class samples into seen classes during test (known as the bias issue) [21, 22]. To alleviate this issue, several strategies have been introduced. A common strategy is to utilize a backbone pre-trained on the ImageNet-1K dataset [13] to extract visual features [7, 24, 26, 36, 50], which is beneficial to improve the generalization ability of visual features. However,

*Corresponding authors: Zengmao Wang, Bo Du

the rich semantics that implicitly contained in the backbone classification heads are simply ignored in these approaches, and thus their ability to exploit diverse semantics is limited. Generative-based GZSL methods [4–6, 16, 51] alleviate the bias problem by introducing the semantics from test unseen classes, and some semantic augmentation approaches focus on enhancing the semantics of each class by leveraging the textual documents [30], large language model [31], etc [19, 50]. However, these approaches share a common limitation that they heavily rely on the semantics and visual features from seen classes to build the relations between visual and semantic space. This makes these models perform poorly on dissimilar unseen classes (see Fig. 1b), as little information can be transferred from seen to dissimilar unseen classes. That is, these methods can only deal with the unrealistic setting that the unseen classes are similar with seen classes (see existing setting of Fig. 1a). However, it is inevitable that unseen classes may be quite different from seen classes in the real-world applications, and hence it is desirable to enhance GZSL models to effectively handle dissimilar unseen classes.

To achieve this goal, we need to address three main challenges: (I) how to obtain the information that can be transferred to dissimilar unseen classes; (II) how to enable effective information transfer without labeled training data; and (III) how to reduce costs as much as possible while ensuring effectiveness. If the cost of the solution exceed the cost of collecting the data and retraining the model, then such a solution would be meaningless. Therefore, we propose a simple yet effective GZSL framework by exploring Diverse Semantics from External Class Names (DSECN), which is robust on both the similar and dissimilar unseen classes. Specifically, we introduce the diverse semantics from external (not test unseen classes) class names as the bridge to reduce the gap between the seen and unseen classes, which is beneficial for the recognition of unseen classes (challenge I). Then we utilize the classification head pre-trained on large-scale dataset, *e.g.*, ImageNet-1K [13], to align the introduced semantics to the visual space (challenge II). Finally, the hierarchical taxonomy of WordNet [28] for the classes in large-scale dataset is introduced to further improve the diversity of semantics from class names. Since the class name and pre-trained classification head are quite easy to collect, the cost of our method is quite low (challenge III).

To summarize, the main contributions of this paper are:

- To the best of our knowledge, we are the first that explicitly study the realistic GZSL setting that both similar and dissimilar unseen classes exist (see our setting in Fig. 1a).
- We propose a GZSL method that explores the diverse semantics from external class names (DSECN), which is simultaneously robust on the similar and dissimilar unseen classes.
- We show that the proposed idea can be easily integrated into other GZSL approaches, such as generative-based ones, and improve their robustness for dissimilar unseen classes.

We conduct extensive experiments on diverse real-world datasets. The results show that in the practical setting including both similar and dissimilar unseen classes, the harmonic mean accuracies of our method significantly outperform all counterparts. Besides, our model can be trained within one minute on all three datasets.

2. Related Work

2.1. Generalized Zero-shot Learning

Existing GZSL methods can be broadly categorized into embedding-based [9, 18, 25, 38] and generative-based ones [2, 5, 12, 16]. Early global embedding-based methods [15, 18, 25, 38, 45] align global visual features with corresponding category semantic information into a common embedding space, which enables knowledge transfer from seen to unseen classes. Recently proposed local embedding-based methods [9–11, 42, 48, 49] utilize attribute descriptions as guidance to discover the discriminative local features between seen classes and unseen classes. Besides, generative-based methods [4, 5, 12, 37, 38, 47] introduce the semantic from test unseen classes to alleviate the bias problem in GZSL [34]. However, these methods focus on the unrealistic setting where unseen classes are assumed to be similar to seen classes, and perform poorly in identifying dissimilar unseen classes. In contrast, our work focuses on a realistic GZSL setting that includes both similar and dissimilar unseen classes, and the proposed DSECN can robustly identify similar and dissimilar unseen classes.

2.2. Semantic Information for GZSL

Semantic information is a bridge that transfers knowledge from seen classes to unseen class recognition and is crucial to GZSL [30, 34, 50]. Most of prior works rely on human-annotated attributes [2, 32, 49] or word vectors [20, 39] as the semantic information. While attributes are accurate, they are costly to annotate and are not suitable for a large-scale problem [20, 50]. In contrast, word vectors require less human labor and are suitable for large-scale datasets [34]. However, word vectors often do not reflect visual similarities, thus limiting the performance [30, 49]. Xu et al. [49] propose a visually-grounded semantic embedding (VGSE) network to discover semantic embeddings containing discriminative visual properties. Naeem et al. [30] propose a transformer-based model I2DFormer that learns semantic embeddings from raw online textual documents. However, these methods only consider semantic enhancement within a single category. When an unseen class is dissimilar to seen classes, the transferable semantic infor-

mation from seen classes is still little, which limits their performance. In contrast, we introduce diverse semantic information from external category names and align the introduced semantics into visual space using the classification head pre-trained on a large-scale dataset , thus assisting in the identification of dissimilar unseen categories.

3. Proposed Method

Problem Formulation. In ZSL and GZSL, we define two sets of classes: seen classes in Y^s and unseen classes in Y^u. The seen classes Y^s and the unseen classes Y^u are disjoint, *i.e.*, $Y^s \cap Y^u = \emptyset$ and $Y^s \cup Y^u = \mathcal{Y}$. The unseen classes may be similar or dissimilar to seen classes in practice. Hence, in this work, the unseen classes Y^u contain similar unseen classes Y^u_s and dissimilar unseen classes Y^u_d, *i.e.*, $Y^u = Y^u_s \cup Y^u_d$. The seen data are expressed as $D^s = \{(x^s_i, y^s_i)\}$, where $x^s_i \in X$ indicates the i-th sample features extracted by the pretrained backbone network, *e.g.*, ResNet101 [17], and $y^s_i \in Y^s$ is its class label. The D^s is split into a training set D^s_{tr} and a testing set D^s_{te}. On the other hand, the unseen data are denoted as $D^u = \{(x^u_i, y^u_i)\}$, where $x^u_i \in X^u$ and $y^u_i \in Y^u$ are the sample features of unseen classes and the corresponding ground-truth label for evaluation, respectively. The goal of ZSL is to learn a classifier for classifying test samples from unseen classes, i.e., $X \xrightarrow{f_{ZSL}} Y^u$. In contrast to ZSL, the goal of GZSL is to learn a classifier for classifying test samples from both seen and unseen classes, i.e., $X \xrightarrow{f_{GZSL}} Y^s \cup Y^u$. In ZSL and GZSL, the auxiliary semantic information A is obtained by transforming the class labels $\mathcal{Y}$ using human-annotated attributes [9] or language models [27, 35].

Overview of framework. As shown in Fig. 2, the framework contains three components: visual flow (§ 3.1), diverse semantic enhancement (DSE, § 3.2), and hierarchy taxonomy enhancement (HTE, § 3.3). The visual flow aligns seen class semantics into visual space. Diverse semantic and hierarchy taxonomy enhancement are proposed to enhance the diversity of semantics available to the model, thereby assisting the identification of dissimilar unseen classes.

3.1. Visual Flow

The visual flow is designed to classify visual objects from both seen and unseen classes by transferring knowledge from the seen classes to the unseen ones with the help of semantic information. The visual flow contains two parts: semantic-to-visual sub-network ($S2V$) and visual classifier. The $S2V$ sub-network is a learnable multilayer perceptron (MLP), and is used to link the semantic and visual representation. This enables the model to transfer the knowledge from seen classes to the unseen classes through semantic information. The visual classifier uses the relationship between the visual sample and all categories to obtain the classification result of the sample. Because the visual flow requires paired visual samples and category labels, in the training phase, the visual flow can only be trained on the paired visual feature and label dataset D^s_{tr}. Specifically, the $S2V$ sub-network takes the seen semantic feature A^s as input to generate the class-level visual prototype of seen classes V^s. The process to generate the class-level visual prototype of seen classes V^s can be expressed by

$$A^s \in \mathbb{R}^{C^s \times d^a} \xrightarrow{S2V} V^s \in \mathbb{R}^{C^s \times d^v}, \tag{1}$$

where C^s, d^a, d^v denote the number of seen classes, the dimensionality of semantic representation and the dimensionality of visual representation, respectively. A^s signifies the semantic representations of seen classes and is extracted by language models [27, 35].

Next, the visual classifier (VC) computes the scaled cosine similarity between the visual feature of seen classes $X^s_{tr} \in \mathbb{R}^{N^s_{tr} \times d^v}$ and the class-level visual feature of seen classes $V^s \in \mathbb{R}^{C^s \times d^v}$ as logits $l^s \in \mathbb{R}^{N^s_{tr} \times C^s}$, where N^s_{tr} is the number of seen class samples in the training set. Formally, the logits l^s can be obtained as follows:

$$l^s = \gamma^2 \frac{X^s_{tr} V^{s\top}}{\|X^s_{tr}\| \, \|V^s\|}, \tag{2}$$

where γ is the scale factor.

Finally, we adopt the Cross-Entropy (CE) loss to update the visual flow:

$$\mathcal{L}_V = \mathcal{L}_{CE}(\sigma(l^s), Y^s_{tr}), \tag{3}$$

where σ denotes the $softmax$ function.

3.2. Diverse Semantic Enhancement

Motivation of DSE. When the unseen classes are dissimilar with seen classes, the visual flow can transfer only little information from seen classes to unseen classes, resulting in poor recognition performance of unseen classes. Motivated by this, we propose the diverse semantic enhancement module, which introduces the semantic information from external class names to help the recognition of unseen classes. To successfully transfer semantic information to unseen classes, establishing a substantial linkage between the semantics inherent in class names and their corresponding visual features is imperative. Nonetheless, the visual feature of the class name is unknown, which poses the primary obstacle in leveraging semantics from class names to enhance Generalized Zero-Shot Learning (GZSL). Considering the visual representation X obtained through the pretrained backbone, *e.g.* ResNet101, on the large-scale dataset, we use the classification head corresponding to pretrained backbone as semantic classifier (SC) to constrain the

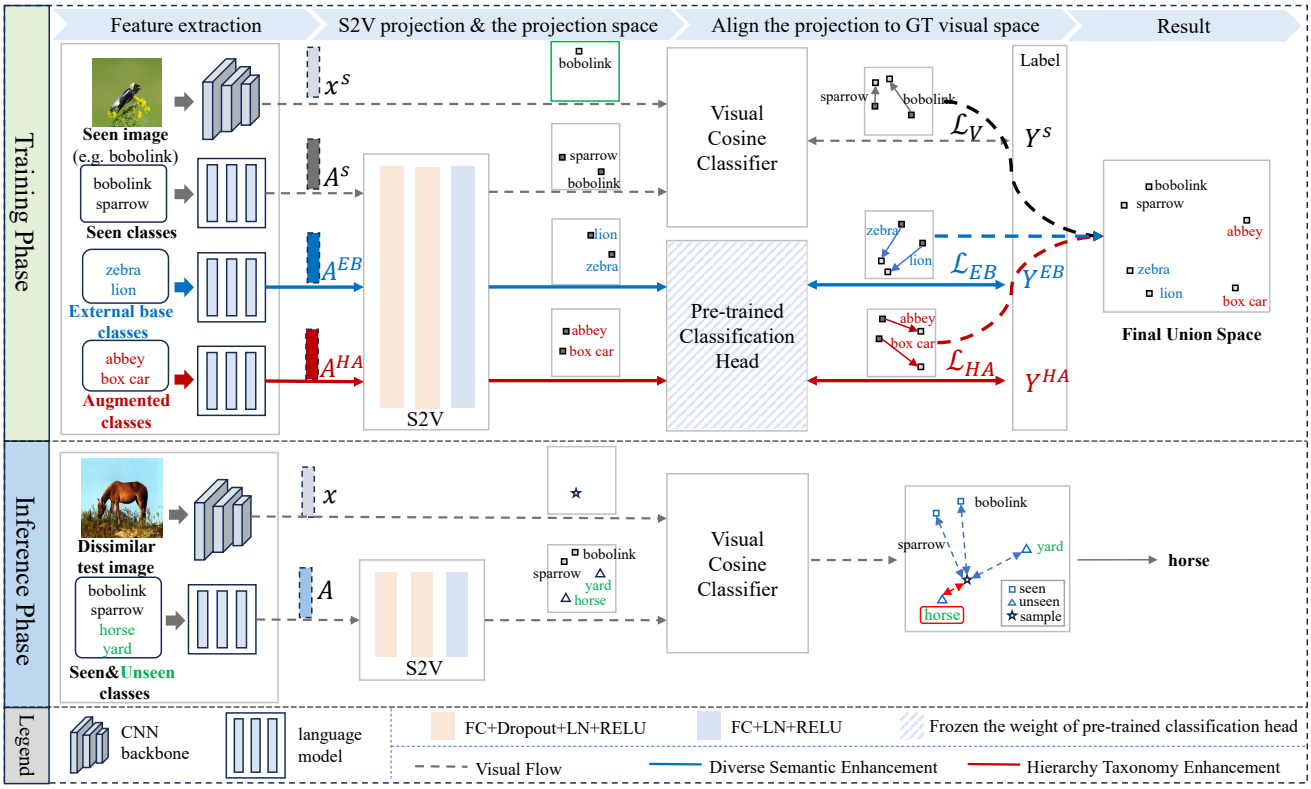


Figure 2. Overview of the proposed DSECN framework. During the training phase, our method contains three components: visual flow, diverse semantic enhancement, and hierarchy taxonomy enhancement. In the visual flow, we employ the visual cosine classifier to align the semantics and visual features of seen classes. Then the diverse semantic enhancement and hierarchy taxonomy enhancement introduce more diverse information from the external base class names and the augmented class names. The introduced diverse semantics are aligned to the visual space using pre-trained classification head, which enables our model to learn a robust union space for both similar and dissimilar unseen classes. During the inference phase, benefiting from the learned union space that contains visual-semantic correspondence of diverse classes, the test image from dissimilar unseen classes can be accurately classified.

generated class-level visual prototype of class names to visual space. Besides, the introduced semantics from class names should be diverse. Hence, we choose the class names of large-scale dataset to extract diverse semantics.

Construction of DSE. We first introduce the class names CN^{EB} of large-scale dataset corresponding to pretrained network and extract the semantic information $A^{EB} \in \mathbb{R}^{C^{EB} \times d^a}$ of CN^{EB} using the Language Model (LM), *i.e.*,

$$A^{EB} = LM(CN^{EB}). \tag{4}$$

C^{EB} denotes the number of external base class names from the large-scale dataset. Then the semantic information A^{EB} are taken as the input of $S2V$ sub-network to generate class-level visual prototype $V^{EB} \in \mathbb{R}^{C^{EB} \times d^v}$, *i.e.*,

$$V^{EB} = S2V(A^{EB}). \tag{5}$$

Next, the generated visual prototype V^{EB} is fed into the semantic classifier (SC) to get the logits $l^{EB} \in \mathbb{R}^{C^{EB} \times C^{PR}}$, *i.e.*,

$$l^{EB} = SC(V^{EB}), \tag{6}$$

where SC is the frozen classification head from pretrained network. C^{PR} is the class number of pretraining dataset.

Due to the backbone pretrained in large-scale dataset, we can obtain the label of CN^{EB} as follows:

$$Y^{EB} = CN2Y(CN^{EB}), \tag{7}$$

where $CN2Y$ is the dictionary mapping from class names to labels in the large-scale dataset corresponding to the pretrained backbone.

Finally, the Cross-Entropy (CE) Loss is adopted to update the diverse semantic enhancement module:

$$\mathcal{L}_{EB} = \mathcal{L}_{CE}(\sigma(l^{EB}), Y^{EB}). \tag{8}$$

Comparison with Analogous Methods. The prevailing GZSL approach depends on transferring semantic knowledge from seen to unseen classes. However, these techniques falter in a realistic GZSL setting that encompasses both similar and dissimilar unseen classes. Our method introduces varied semantics derived from external class names and aligns them with the visual domain through a pre-trained classification head. This process allows our

model to leverage newly introduced semantics as a bridge connecting seen and dissimilar unseen classes, thus enhancing the identification of dissimilar unseen classes. Notably, the procurement cost of class names and pre-trained classification heads is low, rendering our approach more feasible in practice.

3.3. Hierarchy Taxonomy Enhancement

Chihuahua is a kind of pet dog, and the dog is also a kind of animal. That is, there is a hierarchical structure between different categories. Therefore, we can utilize the hierarchy structure of classes to augment more classes. Based on the assumption, we propose a class name augment method based on hierarchy structure of classes. Specifically, we first employ WordNet [28] to extract the subclass names of external base class names (CN^{EB}) as hierarchically augmented class names (CN^{HA}).

$$CN^{HA} = hyponyms(CN^{EB}), \tag{9}$$

where $hyponyms$ denotes the mapping function for extracting hyponyms using WordNet [28]. The semantic information $A^{HA} \in \mathbb{R}^{C^{HA} \times d^a}$ of the CN^{HA} can be extracted by the language model, *i.e.*, $A^{HA} = LM(CN^{HA})$. C^{HA} denotes the number of the augmented class names.

Then we can obtain the predicted probability $l^{HA} \in \mathbb{R}^{C^{HA} \times C^{PR}}$ of the augmented classes as follows:

$$l^{HA} = SC(S2V(A^{HA})), \tag{10}$$

where $S2V$ and SC is the semantic-to-visual ($S2V$) subnetwork and the semantic classifier (SC), respectively. Considering that the subclass is one of the parent classes, we assign the label of the corresponding parent class as the label of the subclass. For example, fire ant is the subclass of ant, and thus the label of fire ant $y^{fire-ant}$ is assigned as the label of ant in the pretraining large-scale dataset, *i.e.*, $y^{fire-ant} = y^{ant} = CN2Y(ant)$. Based on the idea, the label Y^{HA} of CN^{HA} can be obtained as follows:

$$Y^{HA} = CN2Y(hypernyms(CN^{HA})), \tag{11}$$

where $hypernyms$ denotes the mapping function for extracting hypernyms using WordNet [28], $CN2Y$ is the dictionary mapping from class names to labels in the large-scale dataset corresponding to the pretrained backbone.

Finally, the Cross-Entropy (CE) Loss is adopted to obtain the loss of hierarchy taxonomy enhancement:

$$\mathcal{L}_{HA} = \mathcal{L}_{CE}(\sigma(l^{HA}), Y^{HA}). \tag{12}$$

3.4. Training and Inference

The proposed $DSECN$, which contains three components, is trained in an end-to-end manner, and the total loss function is given as follows:

$$\mathcal{L}_{total} = \mathcal{L}_V + \lambda_{EB}\mathcal{L}_{EB} + \lambda_{HA}\mathcal{L}_{HA}, \tag{13}$$

Algorithm 1: Training Process of DSECN

Input: Training seen data $\{X^s_{tr}, CN^s, Y^s_{tr}\}$, external base class names CN^{EB} and hierarchy augmented class names CN^{HA}
Output: The final S2V model

```
// Generate Semantics and Labels
```
$A^s, A^{EB}, A^{HA} = LM(CN^s, CN^{EB}, CN^{HA})$,
$Y^{EB} \leftarrow \{CN^{EB}\}$ in Eq.(7),
$Y^{HA} \leftarrow \{CN^{HA}\}$ in Eq.(11).
for $e = 1, 2, \ldots, E$ **do**
```
    // Visual Flow
```
    $V^s = S2V(A^s)$,
    $l^s \leftarrow \{V^s, X^s_{tr}\}$ in Eq.(2),
    $\mathcal{L}_V \leftarrow \{l^s, Y^s_{tr}\}$ in Eq.(3).
```
    // Diverse Semantic Enhancement
```
    $V^{EB} = S2V(A^{EB})$,
    $l^{EB} = SC(V^{EB})$,
    $\mathcal{L}_{EB} \leftarrow \{l^{EB}, Y^{EB}\}$ in Eq.(8).
```
    // Hierarchy Taxonomy Enhancement
```
    $V^{HA} = S2V(A^{HA})$,
    $l^{HA} = SC(V^{HA})$,
    $\mathcal{L}_{HA} \leftarrow \{l^{HA}, Y^{HA}\}$ in Eq.(12).
```
    // Compute Total Loss
```
    $\mathcal{L}_{total} = \mathcal{L}_V + \mathcal{L}_{EB} + \mathcal{L}_{HA}$.
```
    // Update Parameters
```
    Update the parameters of S2V using Adam.
end

where λ_{EB} and λ_{HA} are trade-off hyper-parameters. The training pseudo-code is presented in Algorithm 1. It is noteworthy that we only update the parameters of $S2V$.

In the inference phase, given a visual feature x and the semantic feature A of all classes, we apply the visual flow to obtain the final class prediction $\hat{y}$. Specifically, the semantic features $A = A^s \cup A^u$ of all classes are first fed into $S2V$ sub-network to generate the class-level visual prototype, *i.e.*, $V = S2V(A)$. Then the final class prediction can be obtained according to:

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} (softmax(\gamma^2 \frac{xV^\top}{\|x\| \, \|V\|})), \tag{14}$$

where $\mathcal{Y} = Y^s \cup Y^u$ is the union of seen classes Y^s and unseen classes Y^u.

3.5. Integrating into Existing GZSL Methods

The proposed DSECN can be easily integrated into other mainstream GZSL approaches to improve their robustness for dissimilar unseen classes. The GZSL method essentially models the relationship between visual and semantic features, so that semantic features can be used as a bridge to accurately classify visual features of unseen classes. DSECN can effectively build such relationship between "diverse" semantic and visual features at low cost, and thus signif-

icantly improving the performance of existing GZSL approaches. Taking generative-based methods as an example, the generator can generate visual features of the introduced diverse semantic from external class names. Then the pre-trained classification head is used to align the generated visual features to visual space. This enables the generator to adapt to more classes and generate more accurate visual features for unseen classes. Finally, the refined visual features of unseen classes are used to train the GZSL classifier, thus improving the model performance. More details on the integration of our idea with the generative-based methods and other types of GZSL approaches can be found in the supplementary material A.

4. Experiments

4.1. Experimental Setting

Datasets. We conduct our experiments on three widely used ZSL benchmark datasets, which are AWA2 [46], CUB [40] and SUN [33]. Since the main focus of this work is to study the more practical GZSL setting that includes both similar and dissimilar unseen classes, the learned GZSL model needs to recognize the dissimilar unseen classes from other datasets. Hence, we do not use the human-annotated attributes, but rather use language models, *e.g.*, W2V [27] and CLIP [35], to extract the semantic embedding of classes.

Evaluation. Unlike most existing methods that only consider similar setting, we comprehensively evaluate the GZSL model on three testing settings, including similar, dissimilar and practical settings. To better describe these settings, we define the set of datasets as $Set = \{AWA2, CUB, SUN\}$. The dataset of seen classes and unseen classes are respectively expressed as ds and du, where $ds \subseteq Set$ and $du \subseteq Set$. To simulate the similar scene, the dataset ds of seen classes is same as the dataset du of unseen classes, *i.e.*, $ds = du$. To simulate the dissimilar scene, we use the complementary set of ds as du, *i.e.*, $du = \complement_{Set} ds$. For practical scene, the unseen classes may either be similar or dissimilar to the seen classes. Therefore, in our practical setting, we use all datasets to evaluate unseen classes, *i.e.*, $du = Set$. For all three settings, we follow [46] to evaluate model for both ZSL and GZSL. In the ZSL, the average per-class top-1 accuracy (T) on unseen classes is taken as the evaluation metric. In the GZSL, the evaluation metric is the harmonic mean H between seen classes accuracy S and unseen class accuracy U, *i.e.*, $H = (2 \times U \times S)/(U + S)$.

Implementation Details. The implementation details are provided in the supplementary material B.

Fair-comparison Guarantee. All methods are trained on seen class images of the training set D^s_{tr}, ensuring that the labeled training data are the same for different approaches. They all adopt the same backbone weights pretrained on large-scale dataset, which means that the pre-training data implicitly utilized by the model is completely consistent. They all use the same language model to extract semantic features, ensuring the fairness of the semantic extraction method. Therefore, the same data are utilized for different methods, and the fairness of comparison is guaranteed. Notably, introducing additional information is a common strategy to mitigate bias issue in GZSL. For example, generative-based GZSL methods [2, 16] introduce semantic information of test unseen classes to mitigate bias. The transductive GZSL methods [1, 41, 44] utilize test unseen class samples to mitigate bias. Unlike these methods, our method does not utilize any semantic or sample information of test unseen classes, thus making our method more suitable for practical GZSL.

4.2. Comparison with State-of-the-Arts

Performance. We comprehensively compare our method with several SOTA GZSL approaches on the similar, dissimilar, and practical settings. Tab. 1 shows that: 1) the performance of all counterparts in identifying dissimilar unseen classes is very poor, which proves that existing models have difficulty in identifying common dissimilar unseen classes in the real world; 2) our method achieves the SOTA performance on all datasets, semantic embeddings and settings, which demonstrates the effectiveness of our method. In particular, our method significantly outperforms all counterparts under dissimilar and practical settings. Take the CUB dataset under dissimilar setting as an example, our method outperforms the most competitive counterpart by 21.76% and 35.14% on the W2V [27] and CLIP text embedding [35], respectively.

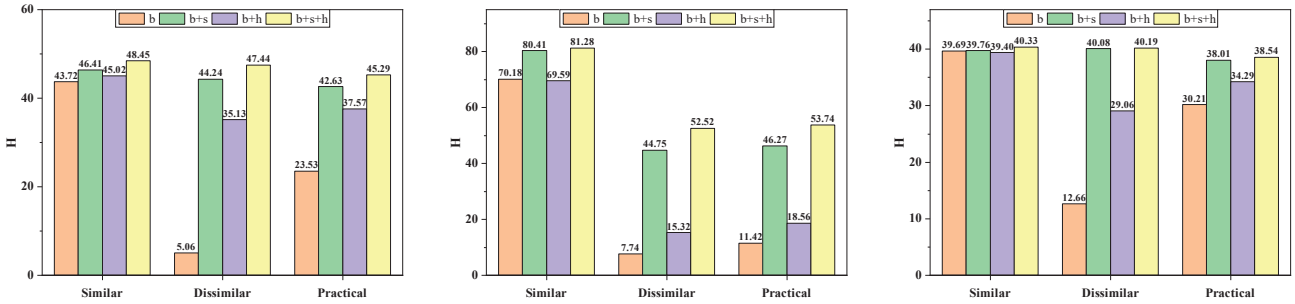
Training Efficiency. We conduct training efficiency comparison experiments on all datasets using CLIP [35] text embedding. Tab. 2 reports the training time of each method. Benefiting from the simple network design, the training costs of CN and the proposed method are much lower than other approaches. Although our method is slightly slower than the CN method because of the introducing semantics from external class names, our method can be trained within one minute.

4.3. Ablation Study

To evaluate the benefits of tackling GZSL with the diverse semantic enhancement (DSE) and the hierarchy taxonomy enhancement (HTE), we conduct an ablation study on all semantics, datasets and settings. The results using CLIP [35] and W2V [27] semantic embedding are reported in Fig. 3 and Fig. S3 of supplementary material C, respectively. From these results, we can see that: 1) regardless of W2V or CLIP semantic representation, the baseline model b has poor recognition performance on dissimilar unseen classes.

Train Data	Methods	W2V						CLIP					
		Similar		Dissimilar		Practical		Similar		Dissimilar		Practical	
		ZSL	GZSL	ZSL	GZSL	ZSL	GZSL	ZSL	GZSL	ZSL	GZSL	ZSL	GZSL
CUB	ZLA† [IJCAI22] [4]	17.24	16.03	3.35	6.00	7.41	11.04	46.56	42.65	4.66	8.52	19.32	24.89
	DGZ† [AAAI23] [5]	18.00	18.42	3.24	5.67	5.47	7.42	47.12	43.16	7.15	12.30	20.61	26.60
	CN‡ [ICLR21] [38]	18.16	22.10	1.12	2.03	5.69	8.74	43.74	42.50	3.82	5.09	17.81	23.25
	TransZero‡ [AAAI22] [9]	17.15	15.65	0.90	0.86	6.49	6.41	49.77	47.99	3.56	2.47	19.18	24.81
	MSDN‡ [CVPR22] [10]	18.30	18.46	1.73	3.33	4.84	6.44	49.17	47.69	2.65	2.79	18.60	23.40
	HASZSL‡ [MM23] [11]	13.46	16.91	1.04	2.03	5.26	6.75	42.51	43.64	2.17	2.81	16.26	17.12
	DSECN (Ours)	**18.80**	**23.40**	**17.73**	**27.76**	**17.81**	**23.63**	**50.70**	**48.45**	**35.54**	**47.44**	**40.85**	**45.29**
AWA2	ZLA† [IJCAI22] [4]	49.78	54.65	3.41	6.21	4.67	8.20	76.59	75.09	5.15	9.32	9.25	14.87
	DGZ† [AAAI23] [5]	54.59	57.02	3.12	5.87	4.84	8.54	78.27	76.59	7.03	12.59	10.30	17.26
	CN‡ [ICLR21] [38]	41.12	49.62	1.66	2.76	3.47	6.30	79.45	73.38	5.48	9.95	8.60	14.72
	TransZero‡ [AAAI22] [9]	39.70	42.90	1.07	1.94	3.34	5.30	69.58	59.78	2.11	3.94	2.67	4.79
	MSDN‡ [AAAI22] [9]	45.90	48.52	1.63	2.52	3.47	5.05	64.42	64.21	3.66	6.03	5.89	9.46
	HASZSL‡ [MM23] [11]	36.59	45.04	0.88	0.98	1.26	1.06	60.34	56.06	6.76	7.73	37.95	37.25
	DSECN (Ours)	**69.23**	**70.71**	**11.21**	**19.88**	**14.37**	**23.79**	**86.97**	**81.28**	**37.01**	**52.52**	**40.00**	**53.74**
SUN	ZLA† [IJCAI22] [4]	33.54	23.90	9.12	13.06	18.70	18.86	54.65	36.70	8.52	11.79	31.59	28.31
	DGZ† [IJCAI22] [4]	31.32	25.08	9.31	13.76	17.53	17.85	53.54	37.73	11.87	17.49	31.61	28.31
	CN‡ [ICLR21] [38]	33.82	26.27	5.17	7.05	18.80	18.14	55.56	38.87	9.69	13.64	32.80	30.00
	TransZero‡ [AAAI22] [9]	30.42	21.81	3.68	2.66	16.69	14.63	54.51	35.86	5.28	2.89	29.98	25.39
	MSDN‡ [AAAI22] [9]	28.13	16.23	1.51	2.78	9.96	7.58	53.33	31.73	5.57	8.20	28.31	22.56
	HASZSL‡ [MM23] [11]	19.51	11.41	3.72	0.12	10.69	1.41	36.53	21.15	4.57	0.12	20.02	8.34
	DSECN (Ours)	**37.85**	**26.81**	**10.54**	**16.47**	**24.19**	**21.69**	**60.28**	**40.33**	**37.11**	**40.19**	**49.10**	**38.54**

Table 1. Comparison with the state-of-the-art GZSL approaches in the three settings, † and ‡ signify the generative-based method and embedding-based approach. Please see details in § 4.2.



(a) CUB (b) AWA2 (c) SUN

Figure 3. Effect of diverse semantic enhancement (s) and hierarchy taxonomy enhancement (h) for GZSL. We remove these two components as our baseline (b). In the ablation study, we add DSE (s) and HTE (h) step by step to show their effect on GZSL. Refer to § 4.3.

Methods	CUB	AWA2	SUN	Mean
ZLA* [4]	4.4 hours	9.5 hours	7.7 hours	7.2 hours
DGZ* [5]	8.3 hours	12.2 hours	16.1 hours	12.2 hours
CN* [38]	8.6 sec	25.8 sec	12.2 sec	15.5 sec
TransZero◇ [9]	40.7 min	2.0 hours	1.6 hours	1.4 hours
MSDN◇ [10]	4.4 min	21.9 min	14.5 min	13.6 min
HAS• [11]	5.4 hours	17.3 hours	7.4 hours	10.0 hours
DSECN* (Ours)	13.9 sec	37.9 sec	19.7 sec	23.8 sec

Table 2. Training time for the recent GZSL methods that made their official implementations publicly available. $*$ and $\diamond$ respectively indicate that the model uses the 2048-dimensional vector and $14 \times 14 \times 2048$-dimensional feature map extracted by pretrained Resnet101 as input. • denotes that the model takes original image as input and finetunes the pretrained backbone. See § 4.2.

This indicates that the problem of identifying dissimilar unseen classes cannot be solved by enhancing the semantic information contained in seen classes; 2) after adding DSE s and HTE h to the baseline, the $b+s+h$ model achieves the best performance on all semantics, datasets and settings. This proves that both the DSE s and the HTE h are critical to GZSL and complementary to each other on all three test settings; 3) the less similar the unseen classes are to the seen classes in the test setting, the greater the performance improvement gain brought by the proposed module s and h. Taking the CUB dataset with CLIP semantic embedding as an example, $b+s+h$ improves the harmonic mean accuracy by 42.38%, 21.76%, and 4.73% on dissimilar, practical and similar settings, respectively. This may be because when the unseen classes are dissimilar to seen classes, the baseline model can only transfer little information from seen classes to unseen classes, while $b+s+h$ introduces a variety of semantic information from class names that can be transferred to unseen classes to improve the recognition performance of

Methods	CUB		AWA2		SUN	
	W2V	CLIP	W2V	CLIP	W2V	CLIP
CN [38]	2.03	5.09	2.76	9.95	7.05	13.64
CN+DSECN	**22.08**	**37.68**	**14.35**	**33.04**	**12.64**	**29.11**
TransZero [9]	0.86	2.47	1.94	3.94	2.66	2.89
TransZero+DSECN	**2.41**	**4.44**	**2.51**	**5.41**	**3.59**	**3.02**
DGZ [5]	5.67	12.30	5.87	12.59	13.76	17.49
DGZ+DSECN	**26.34**	**48.39**	**17.02**	**47.68**	**17.76**	**39.11**

Table 3. Effect of integrating DSECN into existing GZSL methods under dissimilar setting. CN [38], TransZero [9], and DGZ [5] belong to the global embedding-based, local embedding-based, and generation-based GZSL methods respectively. See details in § 4.4.

Semantics	Methods	CUB		AWA2		SUN	
		1K	21K	1K	21K	1K	21K
W2V	b	2.96	1.96	5.51	2.82	11.17	5.50
	b+s+h	27.76	56.97	19.88	40.68	16.47	28.24
	↑	**24.80**	**55.01**	**14.37**	**37.86**	**5.30**	**22.74**
CLIP	b	5.06	6.87	7.74	7.94	12.66	15.55
	b+s+h	47.44	77.60	52.52	71.25	40.19	61.59
	↑	**42.38**	**70.73**	**44.78**	**63.31**	**27.53**	**46.04**

Table 4. Effect of the number of external class names. ↑signifies the performance improvement. Please see § 4.5 for details.

unseen classes. Furthermore, we also qualitatively analyzed the reasons why the proposed module is effective in the supplementary materials D.

4.4. Integration with other GZSL Approaches

We have integrated DSECN into three mainstream GZSL methods and analysed the effect of integrating DSECN. From the results in Tab. 3, we can observe that: 1) benefiting from the introduction of diverse semantic information from class names (DSECN), the performance of global embedding-based (CN) and generative-based methods (DGZ) is significantly improved. For example, when unseen classes are not similar to seen classes, DSECN improves DGZ's GZSL performance by 20.67% in the CUB dataset using W2V [27] embedding. This proves that the DSECN can be integrated into existing global embedding-based and generative-based GZSL methods to improve the model's robustness to dissimilar unseen classes; 2) the improvement with the local embedding-based GZSL method (TransZero) is smaller than that of global embedding-based and generative-based methods. This may be attributed to the fact that the performance improvement of integrating DSECN is limited by the accuracy of the generated visual feature maps. It is challenging to accurately generate not visible visual feature maps from semantic vectors, which is why existing generative-based GZSL methods mostly generate global visual features rather than local visual features. Overall, the proposed DSECN can easily be integrated into existing mainstream GZSL methods to improve the robustness for dissimilar unseen classes.

4.5. Effect of the External Class Name Number

To investigate the effect of the ECN number on identifying dissimilar unseen classes, we introduce the class names from ImageNet-1K and ImageNet-21K separately as the external base class names, and then remove the unseen class names from the external class names. Notably, we use the ResNet101 pretrained on ImageNet-21K when introducing ImageNet-21K class names. The results are shown in Tab. 4. We can observe that as the introduced ECN increases, performance of the model improves significantly. This proves that increasing the ECN is important to improve the performance of dissimilar unseen class recognition.

5. Conclusion

In this paper, we introduce and investigate the practical GZSL setting, where unseen classes can be either similar and dissimilar to seen classes. We empirically show that existing GZSL methods are difficult in identifying dissimilar unseen classes, and propose a simple yet effective method, which exploits diverse semantics from external class names (DSECN), and is simultaneously robust for both similar and dissimilar unseen classes. From the results, we mainly conclude that: 1) the semantics contained in the external unseen class names are quite helpful to improve the generalization ability of GZSL approach; 2) It is critical to align the augmented semantics to their corresponding visual features. In the future, we intend to incorporate the proposed idea into more GZSL approaches and improve their performance.

Potential Impacts. The existing GZSL methods perform poorly when unseen classes are dissimilar to seen classes, which hinders the practical application of the existing GZSL methods. The paper will help attract researchers' attention to dissimilar unseen classes and help the GZSL field develop in a more practical direction. In addition, the proposed DSECN is simultaneously robust on similar and dissimilar unseen classes, and can easily be integrated into other GZSL methods to improve their robustness for dissimilar unseen classes. This capability is beneficial for improving the practicality of the GZSL model. From an evaluation perspective, existing methods assess using visible and invisible classes from the same dataset. This leads to excessive focus on similar unseen classes during evaluation, thereby overestimating the generalizability of existing GZSL methods. In contrast, the cross-dataset evaluation protocol proposed in the paper can more comprehensively reflect the performance of existing GZSL methods on similar, dissimilar and practical settings.

Acknowledgement. This work was supported by the National Key Research and Development Program of China 2023YFC2705700, National Natural Science Foundation of China under Grants 62225113 and 62276195, and Special Fund of Hubei Luojia Laboratory under Grant 220100014.

References

[1] Liu Bo, Qiulei Dong, and Zhanyi Hu. Hardness sampling for self-training based transductive zero-shot learning. In *CVPR*, pages 16499–16508, 2021. 6

[2] Samet Cetin, Orhun Buğra Baran, and Ramazan Gokberk Cinbis. Closed-form sample probing for learning generative models in zero-shot learning. In *ICLR*, 2022. 2, 6

[3] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *ECCV*, pages 52–68. 1

[4] Dubing Chen, Yuming Shen, Haofeng Zhang, and Philip H.S. Torr. Zero-shot logit adjustment. In *IJCAI*, pages 813–819, 2022. Main Track. 1, 2, 7

[5] Dubing Chen, Yuming Shen, Haofeng Zhang, and Philip HS Torr. Deconstructed generation-based zero-shot model. In *AAAI*, pages 295–303, 2023. 1, 2, 7, 8

[6] Shiming Chen, Wenjie Wang, Beihao Xia, Qinmu Peng, Xinge You, Feng Zheng, and Ling Shao. Free: Feature refinement for generalized zero-shot learning. In *ICCV*, pages 122–131, 2021. 2

[7] Shiming Chen, Guosen Xie, Yang Liu, Qinmu Peng, Baigui Sun, Hao Li, Xinge You, and Ling Shao. Hsva: Hierarchical semantic-visual adaptation for zero-shot learning. In *NeurIPS*, pages 16622–16634, 2021. 1

[8] Shiming Chen, Ziming Hong, Wenjin Hou, Guo-Sen Xie, Yibing Song, Jian Zhao, Xinge You, Shuicheng Yan, and Ling Shao. Transzero++: Cross attribute-guided transformer for zero-shot learning. *IEEE TPAMI*, pages 1–17, 2022. 1

[9] Shiming Chen, Ziming Hong, Yang Liu, Guo-Sen Xie, Baigui Sun, Hao Li, Qinmu Peng, Ke Lu, and Xinge You. Transzero: Attribute-guided transformer for zero-shot learning. In *AAAI*, pages 330–338, 2022. 1, 2, 3, 7, 8

[10] Shiming Chen, Ziming Hong, Guo-Sen Xie, Wenhan Yang, Qinmu Peng, Kai Wang, Jian Zhao, and Xinge You. Msdn: Mutually semantic distillation network for zero-shot learning. In *CVPR*, pages 7612–7621, 2022. 7

[11] Zhi Chen, Pengfei Zhang, Jingjing Li, Sen Wang, and Zi Huang. Zero-shot learning by harnessing adversarial samples. In *ACM MM*, pages 4138–4146, 2023. 2, 7

[12] Yu-Ying Chou, Hsuan-Tien Lin, and Tyng-Luh Liu. Adaptive and generative zero-shot learning. In *ICLR*, 2021. 2

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 1, 2

[14] Shay Deutsch, Soheil Kolouri, Kyungnam Kim, Yuri Owechko, and Stefano Soatto. Zero shot learning via multi-scale manifold regularization. In *CVPR*, 2017. 1

[15] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc' Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NeurIPS*, 2013. 1, 2

[16] Zongyan Han, Zhenyong Fu, Shuo Chen, and Jian Yang. Contrastive embedding for generalized zero-shot learning. In *CVPR*, pages 2371–2381, 2021. 2, 6

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[18] Huajie Jiang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Transferable contrastive network for generalized zero-shot learning. In *ICCV*, pages 9765–9774, 2019. 2

[19] Michael Kampffmeyer, Yinbo Chen, Xiaodan Liang, Hao Wang, Yujia Zhang, and Eric P Xing. Rethinking knowledge graph propagation for zero-shot learning. In *CVPR*, pages 11487–11496, 2019. 2

[20] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *CVPR*, pages 3174–3183, 2017. 2

[21] Xia Kong, Zuodong Gao, Xiaofan Li, Ming Hong, Jun Liu, Chengjie Wang, Yuan Xie, and Yanyun Qu. En-compactness: Self-distillation embedding & contrastive generation for generalized zero-shot learning. In *CVPR*, pages 9306–9315, 2022. 1

[22] Gukyeong Kwon and Ghassan Al Regib. A gating model for bias calibration in generalized zero-shot learning. *IEEE TIP*, 2022. 1

[23] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958. IEEE, 2009. 1

[24] Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *CVPR*, pages 7402–7411, 2019. 1

[25] Shichen Liu, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Generalized zero-shot learning with deep calibration network. In *NeurIPS*, 2018. 1, 2

[26] Yang Liu, Lei Zhou, Xiao Bai, Yifei Huang, Lin Gu, Jun Zhou, and Tatsuya Harada. Goal-oriented gaze estimation for zero-shot learning. In *CVPR*, pages 3794–3803, 2021. 1

[27] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 3, 6, 8

[28] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 2, 5

[29] Ashish Mishra, Shiva Krishna Reddy, Anurag Mittal, and Hema A. Murthy. A generative model for zero shot learning using conditional variational autoencoders. In *CVPRW*, 2018. 1

[30] Muhammad Ferjad Naeem, Yongqin Xian, Luc V Gool, and Federico Tombari. I2dformer: Learning image to document attention for zero-shot image classification. In *NeurIPS*, pages 12283–12294, 2022. 2

[31] Muhammad Ferjad Naeem, Muhammad Gul Zain Ali Khan, Yongqin Xian, Muhammad Zeshan Afzal, Didier Stricker, Luc Van Gool, and Federico Tombari. I2mvformer: Large language model generated multi-view document supervision for zero-shot image classification. In *CVPR*, pages 15169–15179, 2023. 2

[32] Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *ECCV*, pages 479–495. Springer, 2020. 2

[33] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, pages 2751–2758, 2012. 6

[34] Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Peng Lim, Xi-Zhao Wang, and Q. M. Jonathan Wu. A review of generalized zero-shot learning methods. *IEEE TPAMI*, 45(4):4051–4070, 2023. 1, 2

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 3, 6

[36] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero- and few-shot learning via aligned variational autoencoders. In *CVPR*, 2019. 1

[37] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *CVPR*, pages 8247–8255, 2019. 2

[38] Ivan Skorokhodov and Mohamed Elhoseiny. Class normalization for (continual)? generalized zero-shot learning. In *ICLR*, 2021. 1, 2, 7, 8

[39] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *NeurIPS*, 2013. 2

[40] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 6

[41] Ziyu Wan, Dongdong Chen, Yan Li, Xingguang Yan, Junge Zhang, Yizhou Yu, and Jing Liao. Transductive zero-shot learning with visual structure constraint. In *NeurIPS*, 2019. 6

[42] Chaoqun Wang, Shaobo Min, Xuejin Chen, Xiaoyan Sun, and Houqiang Li. Dual progressive prototype network for generalized zero-shot learning. In *NeurIPS*, 2021. 2

[43] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2):1–37, 2019. 1

[44] Jiamin Wu, Tianzhu Zhang, Zheng-Jun Zha, Jiebo Luo, Yongdong Zhang, and Feng Wu. Self-supervised domain-aware generative network for generalized zero-shot learning. In *CVPR*, pages 12767–12776, 2020. 6

[45] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *CVPR*, pages 69–77, 2016. 2

[46] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE TPAMI*, 41 (9):2251–2265, 2018. 6

[47] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, 2018. 2

[48] Guo-Sen Xie, Li Liu, Xiaobo Jin, Fan Zhu, Zheng Zhang, Jie Qin, Yazhou Yao, and Ling Shao. Attentive region embedding network for zero-shot learning. In *CVPR*, pages 9384–9393, 2019. 2

[49] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for zero-shot learning. In *NeurIPS*, pages 21969–21980, 2020. 2

[50] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Vgse: Visually-grounded semantic embeddings for zero-shot learning. In *CVPR*, pages 9316–9325, 2022. 1, 2

[51] Zihan Ye, Fuyuan Hu, Fan Lyu, Linyan Li, and Kaizhu Huang. Disentangling semantic-to-visual confusion for zero-shot learning. *IEEE TMM*, 24:2828–2840, 2021. 2

[52] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *CVPR*, pages 6034–6042, 2016. 1