Bridging the Projection Gap: Overcoming Projection Bias Through Parameterized Distance Learning

Chong Zhang[1], Mingyu Jin[1], Qinkai Yu[2], Haochen Xue[1]
Shreyank N Gowda[3], and Xiaobo Jin[1] †

[1] Xi'an Jiaotong-Liverpool University
[2] University of Liverpool,
[3] University of Oxford

Abstract. Generalized zero-shot learning (GZSL) aims to recognize samples from both seen and unseen classes using only seen class samples for training. However, GZSL methods are prone to bias towards seen classes during inference due to the projection function being learned from seen classes. Most methods focus on learning an accurate projection, but bias in the projection is inevitable. We address this projection bias by proposing to learn a parameterized Mahalanobis distance metric for robust inference. Our key insight is that the distance computation during inference is critical, even with a biased projection. We make two main contributions - (1) We extend the VAEGAN (Variational Autoencoder & Generative Adversarial Networks) architecture with two branches to separately output the projection of samples from seen and unseen classes, enabling more robust distance learning. (2) We introduce a novel loss function to optimize the Mahalanobis distance representation and reduce projection bias. Extensive experiments on four datasets show that our approach outperforms state-of-the-art GZSL techniques with improvements of up to 3.5 % on the harmonic mean metric.

Keywords: Generalized zero-shot learning · Mahalanobis distance · Projection bias

1 Introduction

Deep learning (DL) models have achieved recent advances in computer vision and have gained widespread popularity due to their ability to provide end-to-end solutions from feature extraction to classification. Despite their success, traditional deep learning models require extensive labeled data for each category. However,

† Corresponding author: Xiaobo Jin. Email: xiaobo.jin@xjtlu.edu.cn
This work was partially supported by Research Development Fund with No. RDF-22-01-020, the top talent award project RDF-TP-0019, and the "Qing Lan Project" in Jiangsu universities and National Natural Science Foundation of China under Grant U1804159.

collecting large-scale datasets is a challenging problem due to the time and expenses related to it. Zero-shot learning (ZSL) [25,29] technology provides a good solution to this challenge. ZSL aims to train a model that can classify images and realize knowledge transfer from seen classes (source classes) to unseen classes (target domain) through semantic information, which is leveraged to bridge the gap between seen and unseen classes. In real-world scenarios, data samples from

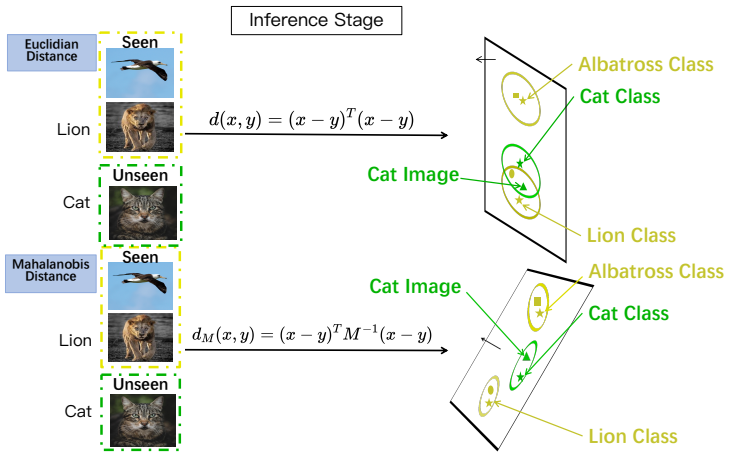

Fig. 1: Demonstration of how the Mahalanobis distance compensates for the biased nature of GZSL in the projection space: when image instances and class descriptions are biased in the projection space, an image from the Cat class (indicated by the green triangle) will be misclassified into the Lion class (deep yellow pentagram) according to the Euclidean distance (the top part); however, the image will be correctly classified into the Cat class according to the Mahalanobis distance (the bottom part).

seen classes often outnumber those from unseen classes. The generalized zero-shot learning (GZSL) paradigm addresses this, aiming to classify both seen and unseen class samples concurrently. The crux of most GZSL techniques is to determine an embedding or projection function that links seen class visual features to their respective semantic vectors. This function then aids in classifying test samples based on their proximity to unseen class semantic vectors. In doing so, GZSL methods bridge the gap between the high-dimensional visual space and the semantic attribute space, facilitating the transfer of learned knowledge to accurately classify unseen classes by exploiting their semantic relationships.

Most GZSL methods learn embedding/projection functions to associate seen low-level visual feature classes with their corresponding semantic vectors. The learned function is used to compute the distance between the semantic representation of the class and the projected representation of the sample and classify them to the nearest class. Since each entry of an attribute vector represents a description of that class, class descriptions with similar features are expected to contain similar attribute vectors in the semantic space. However, in visual space,

classes with similar properties can be quite different. Therefore, finding a precise and suitable embedding space is a challenging task. Otherwise, it may lead to the ambiguity problem of visual semantics.

There is a problem of projection domain bias in the embedding methods used for the Generalized Zero-Shot Learning (GZSL) task. On one hand, vision and semantics are located in two distinct spaces; on the other hand, the samples of seen and unseen classes do not intersect, leading to potential differences in their distributions. Consequently, without appropriate adjustments to the embedding space for unseen classes, a problem of projection domain bias may arise [10, 20, 50]. Since GZSL methods must recognize both seen and unseen categories during inference but only have access to visual features of seen categories during training, they are usually biased toward seen categories. To address this issue, inductive methods incorporate additional constraints or information about the seen classes, whereas transductive methods leverage available information to mitigate the problem of projection domain shifts [5, 13, 17, 31].

Typical GZSL algorithms perform the following three steps during the inference process: 1) Project the image into the space where the class semantic vector is located; 2) Calculate the Euclidean distance between the projection vector and each semantic vector; 3) Classify the image to the class with the closest distance. We argue that Mahalanobis distance takes into account the correlation of features, unlike Euclidean distance. This is critical for addressing projection bias, especially when the class distribution changes significantly. Therefore, in our work, we will learn the Mahalanobis distance to influence the decision in step 3 to mitigate the impact of the projection bias in step 1.

Building on this foundation, we explore the integration of the Mahalanobis distance within the VAEGAN framework [40], which synergizes the strengths of VAE and WGAN models [39] for sampling from data distributions. Our extension introduces a dual-branch structure within the VAEGAN model, specifically designed to accommodate the learning of Mahalanobis distance. The upper branch is tasked with generating unseen class images from the seen class via a generative network, simulating the inference stage for classifying images of unseen classes. Concurrently, the lower branch focuses on directly learning projective representations for seen class images. To optimize this architecture, we propose a novel loss function based on Mahalanobis distance, aimed at minimizing the distance for projections within the same branch while maximizing the separation for projections across different branches. Our key contributions can be summarized as follows:

- The introduction of a Mahalanobis distance metric to GZSL, aiming to counteract the performance degradation due to projection bias. Alongside this, we propose a novel loss function leveraging this distance metric for optimization.
- A novel adaptation of the VAEGAN architecture featuring two discriminative modules, which is designed to address the GZSL challenge where training predominantly encounters seen class samples.

– Robust experimental evidence demonstrating the superiority of our method over existing state-of-the-art techniques on four benchmark datasets.

2 Related Work

Generalized zero-shot learning (GZSL) [3, 34] has garnered significant attention in the computer vision community, as it holds promise in recognizing novel categories without explicit training samples. This section highlights the foundational works in GZSL and the methodologies employed to address the bias towards seen classes, culminating in the motivation for our approach.

2.1 GZSL Foundations and Bias Challenges

Early GZSL approaches leveraged semantic embedding spaces derived from attributes or word vectors to bridge the gap between seen and unseen classes [2, 10, 39, 42]. However, many of these methods were hindered by the challenge of bias towards seen classes, as the projection functions often relied heavily on seen data distributions [11, 45]. The bias problem was later formally analyzed [47], revealing the projection's intrinsic limitations. In contrast to these methods, our work addresses the bias by incorporating a novel loss function that leverages the Mahalanobis distance, offering a more balanced treatment of seen and unseen classes.

2.2 Projection Optimization

A significant portion of GZSL research has centered on optimizing the projection function. For instance, methods such as [13, 48] proposed complex mapping strategies to embed both seen and unseen samples into a shared semantic space. Techniques that integrate auxiliary information or adopt transductive settings to alleviate the bias have also been explored [5]. Our method diverges by enhancing the projection optimization through a dual-branch VAEGAN architecture, which directly addresses the challenge of training predominantly with seen class samples.

2.3 Distance Metric Learning

While optimizing the projection function remains a popular strategy, a few works have identified the importance of distance computation. During the training phase, some methods use Euclidean distance as a constraint to maintain the relationship between the generated visual features and the real semantic representation or use non-Euclidean embedding spaces [32] based on graph networks or manifold learning to maintain the relationship between data samples. However, the traditional Euclidean distance is still used in the inference stage to search for the nearest neighbor class of a given test sample. Our contribution uniquely focuses on the inference stage, where we apply the Mahalanobis distance to improve classification accuracy, distinguishing our approach from previous distance metric learning efforts.

2.4 Generative Models in GZSL

Generative-models-based GZSL methods classify unseen samples using semantic representations. Zero-Sample Learning Semantic Embeddings (SE-ZSL) [9] uses category embeddings, while Generative Zero-Sample Learning with Balanced Semantic Embeddings (LBSE-ZSL) [43] addresses category imbalance. Generative Zero Sample Learning using Visible and Invisible Semantic Relationships (LsrGAN) [36] improves unseen category representation by leveraging class relationships. Guo et al. [14] proposed an image-specific prompt learning (IPL) method, producing a more precise adaptation for each cross-domain image pair, enhancing the generator's flexibility. The integration of generative models like GANs and VAEs has shown promise in generating synthetic unseen class samples. For example, the VAEGAN model [28] fuses the strengths of VAEs and GANs to enhance generation quality. However, a unified architecture that robustly models seen, and unseen data distributions remained an open challenge until our contribution. Our novel adaptation of the VAEGAN architecture introduces a mechanism for learning the Mahalanobis distance, setting our work apart by directly tackling the issue of data distribution modeling for both seen and unseen classes.

The bias towards seen classes in GZSL has been a consistent challenge, with the majority of the efforts focusing on refining the projection function. Our work diverges from this trend, emphasizing the significance of the distance metric during inference and extending the VAEGAN model to learn from both seen and unseen distributions adaptively. This positions our method distinctively in the GZSL landscape, as validated by our experimental results.

3 Method

In this section, we first clarify the problem we aim to solve. Next, we detail the integration of the Mahalanobis distance into the VAEGAN framework and propose a new loss function to facilitate learning the optimal Mahalanobis distance metric. We here use VAEGAN as a Baseline to implement our specific method. Finally, we demonstrate how the Mahalanobis distance is utilized for classification during the inference stage.

3.1 Problem Formulation

The main goal of GZSL is to build a classifier based only on samples $\mathcal{X}^s$ of seen classes $\mathcal{C}^s$ that can simultaneously distinguish samples $\mathcal{X}^u$ from seen classes $\mathcal{C}^s$ and unseen classes $\mathcal{C}^u$, where unseen classes only appear in the test set, e.g. $\mathcal{C}^s \cap \mathcal{C}^u = \phi$. In addition to class labels, current existing methods fully use class-level semantic labels $\mathcal{S}$ (such as attributes or word2vec) to bridge the gap between seen and unseen classes. To this end, we define the training set as $\mathcal{D}^{tr} = \{I_i, \boldsymbol{s}_i, y_i) | I_i \in \mathcal{X}^s, \boldsymbol{s}_i \in \mathcal{S}, y_i \in \mathcal{C}^s\}$, where I_i and $\boldsymbol{s}_i$ represent the image of the i-th sample and its semantic vector, respectively. Similarly, we can represent the test set by $\mathcal{D}^{te} = \{(I_i, \boldsymbol{s}_i, y_i) | I_i \in \mathcal{X}^s \cup \mathcal{X}^u, \boldsymbol{s}_i \in \mathcal{S}, y_i \in \mathcal{C}^s \cup \mathcal{C}^u\}$, where $I_i, \boldsymbol{y}_i$ either belong to the seen classes or belong to the unseen classes.

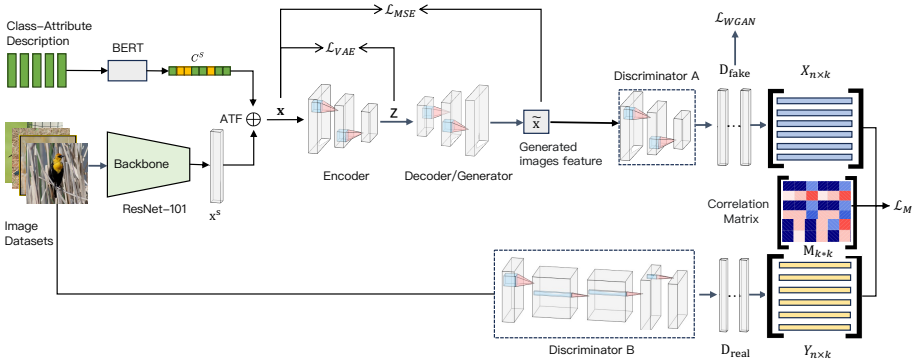

Fig. 2: Framework of VAEGAN with Mahalanobis distance, featuring two branches: the upper branch generates images of unseen classes from seen classes using a generative network to simulate classification of unseen class images during the inference phase; the lower branch directly learns the projective representations of seen class images. The newly proposed Mahalanobis distance-based loss function aims to minimize the distance between projection outputs within the same branch, while maximizing the distance between projections from different branches.

3.2 Framework of VAEGAN

VAEGAN [40] combines the power of VAE models and WGAN models [39] to learn the data distribution of unlabeled samples by sharing the decoder in VAE and the generator in WGAN. We adopt this model to simulate scenarios where samples in the inference stage may come from unseen categories. During the training phase, the generated samples can be seen as coming from some fake unseen class, although these unseen classes are somewhat similar to seen classes.

Feature Extraction Under the framework of VAEGAN, image and text pairs $(I_i, \boldsymbol{s}_i)$ are processed through ResNet101 [16] or ViT-B [8] and BERT [27] to obtain their initial representation such as

$$\bar{\boldsymbol{x}}_i = \text{ResNet101}(I_i) \in R^d, \quad \bar{\boldsymbol{s}}_i = \text{BERT}(\boldsymbol{s}_i) \in R^k. \tag{1}$$

We use the Affine Transformation Fusion (ATF) [35] to replace the common vector concatenation operation to better fuse the two multimodal information while keeping the dimensions unchanged, the schematic of ATF is shown in supplementary. We adopt two MLPs $\alpha(\cdot)$ and $\theta(\cdot)$ to predict the scale parameter and offset parameter of the affine transformation, respectively, as follows

$$\boldsymbol{x}_i = \alpha(\bar{\boldsymbol{s}}_i)\bar{\boldsymbol{x}}_i + \theta(\bar{\boldsymbol{s}}_i) \in R^d, \tag{2}$$

where $\alpha(\cdot)$ will output a scalar, and $\theta(\cdot)$ will output a d-dimensional vector, resulting in a fused representation of image $\bar{\boldsymbol{x}}_i$ and semantics $\bar{\boldsymbol{s}}_i$. In the following description, unless otherwise specified, we will omit the subscript i.

VAEGAN A variational autoencoder (VAE) [22] is a deep generative model capable of learning complex density model variables from latent data. Given a nonlinear generative model $p_\phi(\boldsymbol{x}|\boldsymbol{z})$, where $\boldsymbol{x}$ is the input of the network, the latent variable $\boldsymbol{z}$ comes from the prior distribution $p_0(\boldsymbol{z})$. The goal of a VAE is to approximate the posterior probability distribution of the latent variable $\boldsymbol{z}$ by maximizing the following variational lower bound through an inference network $q_\tau(\boldsymbol{z}|\boldsymbol{x})$

$$\mathcal{L}_{\phi,\tau} = \mathbb{E}_{q_\tau(\boldsymbol{z}|\boldsymbol{x})}[\log p_\phi(\boldsymbol{x}|\boldsymbol{z})] - \mathrm{KL}(q_\tau(\boldsymbol{z}|\boldsymbol{x})||p_0(\boldsymbol{z})). \tag{3}$$

With the above consideration, we minimize the following the VAE loss [40] with the input $\boldsymbol{x}$

$$\mathcal{L}_{\mathrm{VAE}} = \mathrm{KL}(q_\tau(\boldsymbol{z}|\boldsymbol{x})||p_0(\boldsymbol{z})) - \mathbb{E}_{q_\tau(\boldsymbol{z}|\boldsymbol{x})}[\log p_\phi(\boldsymbol{x}|\boldsymbol{z})], \tag{4}$$

where $q_\tau(\boldsymbol{z}|\boldsymbol{x})$ is an encoder $E(x)$, which encodes an input $\boldsymbol{x}$ to a latent variable $\boldsymbol{z}$, $p_\phi(\boldsymbol{x}|\boldsymbol{z})$ is a decoder, which reconstructs the input $\boldsymbol{x}$ from the latent $\boldsymbol{z}$ and the prior distribution $p_0(\boldsymbol{z})$ is assumed to be a standard normal distribution $\mathcal{N}(0,1)$.

It is worth noting that in VAEGAN, VAE's decoder $p_\phi(\boldsymbol{x}|\boldsymbol{z})$ and GAN's generator $G(\boldsymbol{z})$ share a network structure, so we use the discriminator $D_A(\boldsymbol{x})$ to distinguish real and fake samples, where the discriminator $D_A(\boldsymbol{x})$ will be optimized by minimizing the loss function

$$\mathcal{L}_{\mathrm{WGAN}} = \mathbb{E}[D_A(\boldsymbol{x})] - \mathbb{E}[D_A(\tilde{\boldsymbol{x}})] - \lambda\mathbb{E}[(\|\nabla_{\hat{\boldsymbol{x}}} D_A(\hat{\boldsymbol{x}})\|_2 - 1)^2],$$

where $\tilde{\boldsymbol{x}} = G(\boldsymbol{z}) \sim p_\phi(\boldsymbol{x}|\boldsymbol{z})$, $\hat{\boldsymbol{x}} = \alpha\boldsymbol{x} + (1-\alpha)\tilde{\boldsymbol{x}}$ and $\alpha \sim U(0,1)$.

Different from the literature [40], we effectively fuse the semantic information in the input and condition in VAEGAN through the learning of affine transformation (see Eqn.(2)). In addition, to ensure that the generated samples do not deviate too far from the real samples, we introduce the following MSE loss

$$\mathcal{L}_{\mathrm{MSE}} = \mathbb{E}(\boldsymbol{x} - \tilde{\boldsymbol{x}})^2. \tag{5}$$

3.3 Metric Learning with Stochastic Gradient Descent

The root cause of projection bias is that in the projection space, the samples of the seen class and the samples of the unseen class are too close to each other (see Fig. 1). When the samples are classified according to the distance from the class description vector, the samples from the unseen class are likely to be classified into seen classes. The Mahalanobis distance offers significant advantages over Euclidean distance. It accounts for the correlation between data features and automatically adjusts their scales using a covariance matrix, which minimizes the impact of differing scales. To this end, we extend the traditional Euclidean distance metric to a general Mahalanobis distance metric so that under this distance metric, samples from unseen classes will be far away from the class vectors of seen classes, thereby improving the classification performance of GZSL.

Given two vectors X, Y from projected space, we calculate the Mahalanobis distance between X and Y by the following formula

$$d_M^2(X,Y) = (X-Y)^T M(X-Y), \tag{6}$$

where M is a positive definite matrix, which can represent the correlation between the various components of the vector.

For the output $\tilde{\boldsymbol{x}}$ of VAEGAN and the image I, we simulate the projection output of unseen class samples and seen class samples respectively by two discriminators

$$X = D_A(\tilde{\boldsymbol{x}}) \in R^k, \tag{7}$$

$$Y = D_B(I) \in R^k. \tag{8}$$

We stitch N samples by row to get a $2N \times k$ matrix $\tilde{X}$ through the output X and Y of the two branches. In order to learn the optimal matrix M under the framework of gradient descent, we represent M in the following form

$$M = [\mathrm{cov}(\tilde{X}) + \epsilon I]^+, \tag{9}$$

where R^+ represents the generalized inverse matrix of the matrix R, and here since $\mathrm{cov}(\tilde{X})$ may be a singular matrix, a small constant ϵ (10^{-8}) is added to correct the covariance matrix. Note that M is a function of network structure parameters. It should be a symmetrical positive definite distance, thus ensuring that the distance (6) is effective.

Given a batch of samples, we propose a new loss function below so that the Mahalanobis distances of the projection outputs of the different branches are as far as possible, and the Mahalanobis distances of the projection outputs from the same branch are as close as possible

$$\mathcal{L}_M = -\log \sum_{i \neq j} \left(d_M^2(X_i, Y_j) - d_M^2(X_i, X_j) \right), \tag{10}$$

where X_i and Y_j are calculated by Eqn. (7) and (8), respectively. Intuitively, when we use the Mahalanobis distance, minimizing the loss function $\mathcal{L}_M$ will increase the distance between the real (seen) samples and the generated (unseen) samples, and decrease the distance between the generated (unseen) samples.

Ultimately, our algorithm minimizes the following loss function via stochastic gradient descent (as shown in Alg. 1)

$$\mathcal{L} = \mathcal{L}_{\mathrm{WGAN}} + \lambda_{\mathrm{VAE}} \mathcal{L}_{\mathrm{VAE}} + \lambda_{\mathrm{MSE}} \mathcal{L}_{\mathrm{MSE}} + \lambda_{\mathrm{M}} \mathcal{L}_{\mathrm{M}}, \tag{11}$$

where λ_{VAE}, λ_{MSE} and λ_{M} are hyperparameters. Note that in optimizing the two discriminant models D_A and D_B, our loss function is fundamentally different from one in f-VAEGAN-D2 [40]: we optimize them by defining $\mathcal{L}_M$ loss, which is a simple minimization problem; but in f-VAEGAN-D2, their optimization is a classic min-max problem in GAN.

It is worth noting that, since the matrix M depends on the outputs of the two branches, M is constantly updated during model iteration. We use M^* obtained in the last iteration as the optimal metric for the inference phase.

When $M = I$, then the distance metric $d_M^2(X, Y)$ becomes an Euclidean distance. Since M is a positive definite distance, it has the following Cholesky decomposition form $M = LL^T$, and thus we have

$$d_M^2(X, Y) = \|L^T(X - Y)\|_2^2. \tag{12}$$

Algorithm 1 VAEGAN with Mahalanobis Metric

Input: A batch of images and class-attributes pairs $\langle I_i, \boldsymbol{s}_i, y_i \rangle$
for $i = 1, 2, \cdots$ **do**
  $\bar{\boldsymbol{x}}_i = \text{ResNet-101}(I_i)$
  $\bar{\boldsymbol{s}}_i = \text{BERT}(\boldsymbol{s}_i)$
  $\boldsymbol{x}_i = \alpha(\bar{\boldsymbol{s}}_i)\bar{\boldsymbol{x}}_i + \theta(\bar{\boldsymbol{s}}_i)$
  Encode: $\boldsymbol{z}_i = E(\boldsymbol{x}_i)$
  Decode: $\tilde{\boldsymbol{x}}_i = G(\boldsymbol{z}_i)$
  Branch A: $X_i = D_A(\tilde{\boldsymbol{x}}_i)$
  Branch B: $Y_i = D_B(I_i)$
end for
$\tilde{X} = \text{cat}((X_1, X_2, \cdots, Y_1, Y_2, \cdots), \text{dim} = 0)$
$M = [\text{cov}(\tilde{X}) + \epsilon I]^+$
Compute the loss function $\mathcal{L}$ with $\mathcal{L} = \mathcal{L}_{\text{WGAN}} + \lambda_{\text{VAE}}\mathcal{L}_{\text{VAE}} + \lambda_{\text{MSE}}\mathcal{L}_{\text{MSE}} + \lambda_{\text{M}}\mathcal{L}_{\text{M}}$
return $\mathcal{L}$

Unlike the projection bias problem in which the projection to vector X is learned, our network optimizes the projection matrix of the difference $X - Y$ of any two vectors X and Y. It is worth noting that the weight matrix in the Mahalanobis distance does not introduce additional parameters and only depends on the structural parameters of the network, which limits the complexity of the model and reduces the risk of model overfitting.

3.4 Inference with Mahalanobis Metric

In the inference stage (as shown in Alg. 2), for any image I, it and all class description text $s \in \mathcal{X}^s \cup \mathcal{X}^u$ are input into the upper branch of the network as multiple pairs, and the semantic representation X of the class prototype is obtained through the discriminator A; at the same time, the image is directly passed through the lower branch of the network to obtain an embedded representation Y of the image. Finally, according to the Mahalanobis distance between the image and the class prototype, the image will be classified into the nearest class.

We first describe four popular public datasets and experimental implementation details. We then describe and compare the experimental implementation details with certain classical approaches. Finally, we conduct an ablation study to test the effectiveness of four important components of the work.

3.5 Datasets and Evaluation Details

We adopted four public datasets including Caltech-UCSD Birds-200-2011 (CUB) [37], Animals with Attribute 1 (AWA1) [24], Animals with Attribute 2 (AWA2) [38], SUN Database (SUN) [41], and four other datasets. The CUB dataset contains 11,788 images of 200 species of birds, with about 60 images from each category. The AWA1 and AWA2 datasets each collect 50 different animal categories with about 40 to 60 images, and the AWA1 and AWA2 datasets have

Algorithm 2 Inference with Mahalanobis Metric

Input: Any given image I
dist = []
Branch B: $Y = D_B(I)$
for $s \in \mathcal{X}^s \cup \mathcal{X}^u$ **do**
  $\bar{\boldsymbol{x}} = \text{ViT-B}(I)$
  $\bar{\boldsymbol{s}} = \text{BERT}(\boldsymbol{s})$
  $\boldsymbol{x} = \alpha(\bar{\boldsymbol{s}})\bar{\boldsymbol{x}} + \theta(\bar{\boldsymbol{s}})$
  Encode: $\boldsymbol{z} = E(\boldsymbol{x})$
  Decode: $\tilde{\boldsymbol{x}} = G(\boldsymbol{z})$
  Branch A: $X = D_A(\tilde{\boldsymbol{x}})$
  $d = d^2_{M^*}(X, Y)$
  dist.append(d)
end for
$c = \arg\max(\text{dist})$
return c

30,475 and 37,322 images, respectively. The SUN dataset has images from 717 different scene categories, with about 200 to 500 images per category, totaling about 14,340. In dividing the dataset, we followed the conventional division method of GZSL datasets.

In the evaluation method, we used the harmonic mean H to evaluate the recognition results on both visible and invisible class data simultaneously, which is often used to evaluate the classification performance of the GZSL task and is calculated as follows $H = \frac{2\times U\times S}{U+S}$, where U and S denote the classification accuracy on unseen classes and seen class data, respectively.

3.6 Implementation Details

For basic visual features and visual extraction, we refer to VAEGAN [40]. We use pre-trained ResNet-101 [16] and Bert Tokenizer [6] to extract the visual and semantic features of images and generate 2048-dimensional visual feature vectors and 768-dimensional semantic feature vectors, respectively. We also use ViT-B [8] that generates 768-dimensional visual feature vector for our generalization experiments. The dimension of the generated feature vector follows the initial hidden size of the model to ensure minimum feature loss [7,8]. These two vectors are fused into a $32 \times 3 \times 384 \times 384$ tensor for use in the encoding stage. Subsequently, we obtain a latent semantic representation of size 500. The samples generated by the latent semantic representation pass through the discriminator A to obtain a vector of size 900. Similarly, the original image also gets a 900-dimensional vector through the discriminator B to facilitate our calculation of the Mahalanobis distance. The ADAM [21] optimizer is used in our algorithm, where the learning rate is set to 10^{-3}. We found that simply setting λ_{VAE}, λ_{MSE} and λ_{M} to 1, 1, and 1 gave the best results.

3.7 Ablation Study

Mahalanobis/Euclidean Distance Metric The Mahalanobis distance plays a key role in our algorithm. To verify its effectiveness, we compare it with ordinary Euclidean distance. Since the Euclidean distance contains no parameters, we remove the $\mathcal{L}_M$ loss during training. The comparison results of using Euclidean distance and Mahalanobis distance on the four data sets are shown in Tab. 1. From the experimental results in Tab. 1, the Euclidean distance measure performs very poorly under our model framework because our network uses two discriminator branches for learning Mahalanobis distance, which also shows the important role of Mahalanobis distance. We observe that the baseline architecture of VAEGAN is shown in Tab. 2 also uses Euclidean distance, and the performance degradation of 4 data sets is not very obvious. At the same time, Mahalanobis distance takes into account the interaction between sample attribute features in the projection space. It is continuously optimized in the iterative process to better distinguish seen classes from unseen classes, alleviating the projection offset problem in the GZSL problem.

Table 1: Performance of Our method with Euclidean Distance/Mahalanobis Distance

Model	CUB			AWA1			AWA2			SUN		
	U	S	H	U	S	H	U	S	H	U	S	H
+ Euclidean Distance	16.9	41.6	24.0	18.1	46.3	26.1	19.4	43.2	26.8	9.7	18.4	12.4
+ Mahalanobis Distance	**62.1**	**74.6**	**67.8**	**67.2**	**76.3**	**71.5**	**64.9**	**79.1**	**71.3**	**45.7**	**49.8**	**47.7**

Single-branch/Multi-branch Discriminator(s) In our work, we improve the structure of VAEGAN, in particular, we introduce another branch to better learn discriminative features between seen and unseen classes (see Tab. 2). On the other hand, we utilize two branches to define the loss $\mathcal{L}_M$ to learn the Mahalanobis distance metric. Note that Discriminator B cannot be used alone under our framework.

Table 2: Ablation study on our method with augmented discriminator

Model	CUB			AWA1			AWA2			SUN		
	U	S	H	U	S	H	U	S	H	U	S	H
VAEGAN with Discriminator A	46.4	62.1	53.1	66.3	61.2	63.6	54.1	69.8	61.0	38.0	45.7	41.5
+ Augmented Discriminator B	**62.1**	**74.6**	**67.8**	**67.2**	**76.3**	**71.5**	**64.9**	**79.1**	**71.3**	**45.7**	**49.8**	**47.7**

According to the conducted experiments, our model greatly outperforms the baseline of VAEGAN in all categories except unseen categories in the AWA1 dataset (using the same ResNet features). This suggests that branch B of our

model alleviates the projection bias/shift problem of samples to some extent and helps alleviate the problem related to semantic imbalance.

Table 3: Performance comparison of our method under different information fusion

Model	CUB			AWA1			AWA2			SUN		
	U	S	H	U	S	H	U	S	H	U	S	H
+ Concatenation operation	56.0	**81.6**	66.4	61.0	**82.3**	70.1	61.1	**80.4**	69.4	33.7	**52.2**	41.0
+ Affine transformation fusion	**62.1**	74.6	**67.8**	**67.2**	76.3	**71.5**	**64.9**	79.1	**71.3**	**45.7**	49.8	**47.7**

Affine Transformation on Fusion/Concatenation The fusion of multiple modalities, such as images and texts, is usually fused by the concatenation operation. However, multiple modalities of the same sample are interrelated, and affine transformation fusion makes the change of text representation directly affect the mapping function to the image, realizing a closer information fusion between the two. To verify the effectiveness of the Affine Transformation Fusion (ATF) module under our model framework, we list the results of their comparison in Tab. 3. According to the experimental results, the affine transformation function shows a slight improvement over models that only capture image features and semantic features for all categories except the SUN dataset. As a result, the information fusion effect of this module is equivalent to the general concatenation operation.

Impact of Hyperparameters λ_{VAE},λ_{MSE} and λ_{M} We study the impact of the tradeoff parameters λ_{VAE}, λ_{MSE} and λ_{M} on the performance of our algorithm. Fig. **??** shows the recognition results of our method on seen classes and unseen classes under different parameter values, as well as their harmonic (Tab. 4).

Table 4: Comparison of the influence of different loss functions on model performance: $\mathcal{L}_{\mathrm{M}}$ based on Mahalanobis distance has the greatest impact, and the performance of the model drops the most when $\mathcal{L}_{\mathrm{M}}$ is removed (the comparative performance is marked in red and blue)

Model	CUB			AWA1			AWA2			SUN		
	U	S	H	U	S	H	U	S	H	U	S	H
$\mathcal{L}_{\mathrm{WGAN}}$	7.6	10.1	8.7	5.1	6.3	5.6	4.1	5.4	4.7	3.1	4.0	3.5
$\mathcal{L}_{\mathrm{WGAN}} + \mathcal{L}_{\mathrm{MSE}} + \mathcal{L}_{\mathrm{M}}(\lambda_{\mathrm{VAE}} = 0)$	15.8	25.7	19.6	15.7	27.1	19.9	14.4	26.1	18.6	12.3	17.9	14.6
$\mathcal{L}_{\mathrm{WGAN}} + \mathcal{L}_{\mathrm{VAE}} + \mathcal{L}_{\mathrm{M}}(\lambda_{\mathrm{MSE}} = 0)$	43.7	49.9	46.6	38.8	52.3	44.5	42.3	55.6	48.0	38.8	41.5	40.1
$\mathcal{L}_{\mathrm{WGAN}} + \mathcal{L}_{\mathrm{VAE}} + \mathcal{L}_{\mathrm{MSE}}(\lambda_{\mathrm{M}} = 0)$	8.8	27.3	13.3	6.9	12.7	8.9	7.6	14.7	10.0	5.4	10.1	7.0
$\mathcal{L}_{\mathrm{WGAN}} + \mathcal{L}_{\mathrm{VAE}} + \mathcal{L}_{\mathrm{MSE}} + \mathcal{L}_{\mathrm{M}}$	62.1	74.6	67.8	67.2	76.3	71.5	64.9	79.1	71.3	45.7	49.8	47.7

As can be seen from Tab. 4, the model only using WGAN has weak discriminability. We remove different losses from the total loss to obtain the value of

each loss's influence on the model, where the hyperparameter corresponding to the loss is set to 0. The performance of each loss function is consistent on various data sets. For example, on the CUB data set, the loss $\mathcal{L}_{\mathrm{M}}$ has the greatest influence, which causes the harmonic mean H of the model to drop from 67.8 to 13.3, followed by the loss $\mathcal{L}_{\mathrm{VAE}}$, and the loss $\mathcal{L}_{\mathrm{MSE}}$ has the smallest impact, that is $\mathcal{L}_{\mathrm{M}} > \mathcal{L}_{\mathrm{VAE}} > \mathcal{L}_{\mathrm{MSE}}$. Therefore, since the function of loss $\mathcal{L}_{\mathrm{M}}$ is to learn the Mahalanobis distance, it can be seen that the Mahalanobis distance plays a very important role.

We assign values of 0.1, 0.5, and 0.8 to one of λ_{VAE}, λ_{MSE} and λ_{M} respectively, while keeping the other two parameters set to 1. As part of the ablation, we verified the impact of three parameter settings on algorithm performance on the data sets CUB and AWA2 and finally obtained 18 sets of experimental results, as shown in Fig. **??**. On the one hand, for each loss, when we increase its weight, the performance of the algorithm is improved, especially the harmonic mean accuracy H. But $\mathcal{L}_{\mathrm{MSE}}$ is an exception. We need to set the parameters carefully to avoid some degree of overfitting. On the other hand, under the same weight, λ_{M} and λ_{VAE} have a greater impact on the algorithm. When the parameter increases from 0.1 to 0.8, the harmonic mean increases from about 20% to more than 60%.

3.8 Comparison with State-of-the-Art Methods

In this section, we select recent results of the state of the art in GZSL tasks, including LisGAN [26], f-VAEGAN-D2 [40], CADA-VAE [33], CE-GZSL [15], HSVA [4], DAZLE [18], DEM [49], CADA-VAE [33], DAZLE [18], CMC-GAN [44], CMC-GAN + SPOT [12], DFTN [19], CvDSF [46] and BSeGN [43], f-VAEGAN-D2 [40], ERPL [13], CLIP [30] and ICCE [23], see Tab. 5 for details. Some of the previous SOTA results are put into supplementary for reference, including ALE [1], f-CLSWGAN [39].

Tab. 5 lists the result comparison between our method and other classical methods. Specifically, in our method, the H-score can reach 67.2% on the CUB dataset, 71.6% on AWA1, 70.9% on AWA2, and 45.9% on SUN. Compared with the original VAEGAN model [43]: Our method improves the H-score of the model from 58.0% to 67.2% on the CUB dataset, from 67.4% to 70.9% on the AWA2, and finally, the SUN data set is increased from 42.9% to 45.9%. We attribute the above results to three aspects: 1) Under the generative framework of VAEGAN, we simulate the output from seen samples and the output of unseen samples through two branches and make the samples of the two branches as far as possible through the loss function, at the same time, the samples of the same branch are as close as possible so that the seen class and the (pseudo) unseen class can be separated as much as possible; 2) Mahalanobis distance can help us correct wrong decisions when projection bias occurs, thereby improving classification performance on seen and unseen classes; 3) The method is backbone agnostic, replacing ResNet101 with ViT-B maintains high overall performance, showing the generalization of the approach.

Table 5: Comparison of our method with other state-of-the-art methods on four datasets. Except for LisGAN and ERPL, all baselines use the same 2048-D feature vectors of ResNet101 pretrained on ImageNet.

Model (Year)	CUB			AWA1			AWA2			SUN		
	U	S	H	U	S	H	U	S	H	U	S	H
ERPL (2018)	43.7	37.2	40.2	66.4	50.4	57.4	-	-	-	-	-	-
LisGAN (2019)	46.5	57.9	51.6	52.6	76.3	62.3	54.3	68.5	60.6	42.9	37.8	40.2
f-VAEGAN-D2 (2019)	48.4	60.1	53.6	62.9	63.3	63.5	57.6	70.6	63.5	45.1	38.0	41.3
CADA-VAE (2019)	51.6	53.5	52.4	57.3	72.8	64.1	55.8	75	63.9	47.2	35.7	40.6
CE-GZSL(2021)	54.2	67.2	61.4	65.3	73.4	69.1	63.1	78.6	70.0	48.8	38.6	43.1
HSVA (2021)	52.2	59.7	55.7	61.1	75.2	67.4	57.4	81.1	67.3	48.6	39.0	43.3
CLIP (2021)	55.2	54.8	55.0	**85.2**	59.7	73.9	88.3	93.1	**90.6**	41.2	53.4	46.5
ICCE (2022)	67.3	65.5	66.4	67.4	81.2	73.5	65.3	82.3	72.8	-	-	-
BSeGN (2022)	55.3	60.8	58.0	-	-	-	59.3	78.0	67.4	48.9	38.3	42.9
DAZLE (2023)	56.7	59.6	58.1	-	-	-	60.3	75.7	67.1	**52.3**	24.3	33.2
CMC-GAN (2023)	52.6	65.1	58.2	63.2	70.6	66.7	-	-	-	48.2	40.8	44.2
CMC-GAN + SPOT (2023)	53.1	66.7	59.1	63.3	73.8	68.1	-	-	-	48.9	44.1	46.4
DFTN (2023)	61.8	67.2	64.4	56.3	**83.6**	67.3	61.1	78.5	68.7	-	-	-
CvDSF (2023)	53.7	60.0	56.9	64.5	71.4	67.8	**65.6**	70.4	67.9	49.2	38.0	42.9
Our model + ResNet101	57.1	**81.6**	67.2	62.9	83.1	**71.6**	62.2	**82.3**	70.9	39.6	**52.7**	45.9
Our model + ViT-B	**62.1**	74.6	**67.8**	67.2	76.3	71.5	64.9	79.1	71.3	45.7	49.8	**47.7**

4 Conclusions

The biased projection is an important challenging problem in GZSL. In our work, we introduce the Mahalanobis distance into the VAEGAN framework. To this end, we use two branches to learn the samples of the seen class and the samples of the (pseudo) unseen class, respectively, and propose a new loss function such that the projected space is learned to be more discriminative for samples from unseen and seen classes. In particular, the weight matrix of the Mahalanobis distance does not introduce additional parameters, which limits the expressive ability of the model and avoids the possibility of further overfitting. Finally, our extensive experimental evaluation shows that our proposed method outperforms the state-of-the-art methods on four benchmark datasets. Our contribution has significant implications for advancing zero-shot learning and provides a promising avenue for future research.

References

1. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for image classification. IEEE transactions on pattern analysis and machine intelligence **38**(7), 1425–1438 (2015) 13
2. Changpinyo, S., Chao, W.L., Gong, B., Sha, F.: Synthesized classifiers for zero-shot learning. In: CVPR. pp. 5327–5336 (2016) 4
3. Chen, L., Zhang, H., Xiao, J., Liu, W., Chang, S.F.: Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In: CVPR. pp. 1043–1052 (2018) 4

4. Chen, S., Xie, G., Liu, Y., Peng, Q., Sun, B., Li, H., You, X., Shao, L.: Hsva: Hierarchical semantic-visual adaptation for zero-shot learning. NeurIPS **34**, 16622–16634 (2021) 13
5. Cheraghian, A., Rahman, S., Campbell, D., Petersson, L.: Transductive zero-shot learning for 3d point cloud classification. In: CVPR. pp. 923–933 (2020) 3, 4
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018) 10
7. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR **abs/1810.04805** (2018) 10
8. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 6, 10
9. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., Mikolov, T.: Devise: A deep visual-semantic embedding model. NeurIPS **26** (2013) 5
10. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S.: Transductive multi-view zero-shot learning. IEEE transactions on pattern analysis and machine intelligence **37**(11), 2332–2345 (2015) 3, 4
11. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S.: Transductive multi-view zero-shot learning. IEEE Trans. Pattern Anal. Mach. Intell. **37**(11), 2332–2345 (2015) 4
12. Gowda, S.N.: Synthetic sample selection for generalized zero-shot learning. In: CVPR. pp. 58–67 (2023) 13
13. Guan, J., Zhao, A., Lu, Z.: Extreme reverse projection learning for zero-shot recognition. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. pp. 125–141 3, 4, 13
14. Guo, J., Wang, C., Wu, Y., Zhang, E., Wang, K., Xu, X., Song, S., Shi, H., Huang, G.: Zero-shot generative model adaptation via image-specific prompt learning. In: CVPR. pp. 11494–11503 (June 2023) 5
15. Han, Z., Fu, Z., Chen, S., Yang, J.: Contrastive embedding for generalized zero-shot learning. In: CVPR. pp. 2371–2381 (2021) 13
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016) 6, 10
17. Huo, Y., Ding, M., Zhao, A., Hu, J., Wen, J.R., Lu, Z.: Zero-shot learning with superclasses. In: ICONIP. pp. 460–472. Springer (2018) 3
18. Huynh, D., Elhamifar, E.: Fine-grained generalized zero-shot learning via dense attribute-based attention. In: CVPR. pp. 4483–4493 (2020) 13
19. Jia, Z., Zhang, Z., Shan, C., Wang, L., Tan, T.: Dual-focus transfer network for zero-shot learning. Neurocomputing **541**, 126264 (2023) 13
20. Jia, Z., Zhang, Z., Wang, L., Shan, C., Tan, T.: Deep unbiased embedding transfer for zero-shot learning. IEEE Transactions on Image Processing **29**, 1958–1971 (2019) 3
21. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 10
22. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013) 7
23. Kong, X., Gao, Z., Li, X., Hong, M., Liu, J., Wang, C., Xie, Y., Qu, Y.: Encompactness: Self-distillation embedding & contrastive generation for generalized zero-shot learning. In: CVPR. pp. 9306–9315 (2022) 13

24. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR. pp. 951–958. IEEE (2009) 9
25. Larochelle, H., Erhan, D., Bengio, Y.: Zero-data learning of new tasks. In: Fox, D., Gomes, C.P. (eds.) AAAI 2008. pp. 646–651. AAAI Press (2008) 2
26. Li, J., Jing, M., Lu, K., Ding, Z., Zhu, L., Huang, Z.: Leveraging the invariant side of generative zero-shot learning. In: CVPR. pp. 7402–7411 (2019) 13
27. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019) 6
28. Narayan, S., Gupta, A., Khan, F.S., Snoek, C.G., Shao, L.: Latent embedding feedback and discriminative features for zero-shot classification. In: ECCV. pp. 479–495. Springer (2020) 5
29. Palatucci, M., Pomerleau, D., Hinton, G.E., Mitchell, T.M.: Zero-shot learning with semantic output codes. NeurIPS **22** (2009) 2
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Clip: Learning transferable visual models from natural language supervision. arXiv preprint arXiv:2103.00020 (2021) 13
31. Rahman, S., Khan, S., Barnes, N.: Transductive learning for zero-shot object detection. In: ICCV. pp. 6082–6091 (2019) 3
32. Rezaei, M., Shahidi, M.: Zero-shot learning and its applications from autonomous vehicles to covid-19 diagnosis: A review. Intelligence-Based Medicine **3-4**, 100005 (2020) 4
33. Schonfeld, E., Ebrahimi, S., Sinha, S., Darrell, T., Akata, Z.: Generalized zero-and few-shot learning via aligned variational autoencoders. In: CVPR. pp. 8247–8255 (2019) 13
34. Socher, R., Ganjoo, M., Manning, C.D., Ng, A.: Zero-shot learning through cross-modal transfer. NeurIPS 2024 **26** (2013) 4
35. Tao, M., Tang, H., Wu, F., Jing, X.Y., Bao, B.K., Xu, C.: Df-gan: A simple and effective baseline for text-to-image synthesis. In: CVPR. pp. 16515–16525 (2022) 6
36. Vyas, M.R., Venkateswara, H., Panchanathan, S.: Leveraging seen and unseen semantic relationships for generative zero-shot learning. In: ECCV. pp. 70–86. Springer (2020) 5
37. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: Cub_200_2011. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011) 9
38. Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. IEEE transactions on pattern analysis and machine intelligence **41**(9), 2251–2265 (2018) 9
39. Xian, Y., Lorenz, T., Schiele, B., Akata, Z.: Feature generating networks for zero-shot learning. In: CVPR. pp. 5542–5551 (2018) 3, 4, 6, 13
40. Xian, Y., Sharma, S., Schiele, B., Akata, Z.: f-vaegan-d2: A feature generating framework for any-shot learning. In: CVPR. pp. 10275–10284 (2019) 3, 6, 7, 8, 10, 13
41. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: CVPR. pp. 3485–3492. IEEE (2010) 9
42. Xie, G.S., Liu, L., Jin, X., Zhu, F., Zhang, Z., Qin, J., Yao, Y., Shao, L.: Attentive region embedding network for zero-shot learning. In: CVPR. pp. 9376–9385 (2019) 4

43. Xie, G.S., Zhang, X.Y., Xiang, T.Z., Zhao, F., Zhang, Z., Shao, L., Li, X.: Leveraging balanced semantic embedding for generative zero-shot learning. IEEE Transactions on Neural Networks and Learning Systems (2022) 5, 13
44. Yang, F.E., Lee, Y.H., Lin, C.C., Wang, Y.C.F.: Semantics-guided intra-category knowledge transfer for generalized zero-shot learning. IJCV **131**(6), 1331–1345 (2023) 13
45. Ye, Z., Yang, G., Jin, X., Liu, Y., Huang, K.: Rebalanced zero-shot learning. IEEE Transactions on Image Processing **32**, 4185–4198 (2023) 4
46. Zhai, Z., Li, X., Chang, Z.: Center-vae with discriminative and semantic-relevant fine-tuning features for generalized zero-shot learning. Signal Processing: Image Communication **111**, 116897 (2023) 13
47. Zhang, F., Shi, G.: Co-representation network for generalized zero-shot learning. In: Chaudhuri, K., Salakhutdinov, R. (eds.) ICML. vol. 97, pp. 7434–7443. PMLR (09–15 Jun 2019) 4
48. Zhang, L., Wang, P., Liu, L., Shen, C., Wei, W., Zhang, Y., Van Den Hengel, A.: Towards effective deep embedding for zero-shot learning. IEEE Transactions on Circuits and Systems for Video Technology **30**(9), 2843–2852 (2020) 4
49. Zhang, L., Xiang, T., Gong, S.: Learning a deep embedding model for zero-shot learning. In: CVPR. pp. 2021–2030 (2017) 13
50. Zhao, B., Sun, X., Yao, Y., Wang, Y.: Zero-shot learning via shared-reconstruction-graph pursuit. arXiv preprint arXiv:1711.07302 (2017) 3