FREE: Feature Refinement for Generalized Zero-Shot Learning

Shiming Chen[1], Wenjie Wang[1], Beihao Xia[1], Qinmu Peng[1], Feng Zheng[2], Weiping Ding[3] and Xinge You[1,*]
[1]Huazhong University of Science and Technology (HUST), China
[2]Southern University of Science and Technology (SUSTech), China
[3]Nantong University (NTU), China
{shimingchen,wangwj54,xbh_hust,pengqinmu,youxg}@hust.edu.cn zfeng02@gmail.com ding.wp@ntu.edu.cn

Abstract

*Generalized Zero-Shot Learning (GZSL) has achieved promising progress when many efforts have been mainly contributed to overcome the problems of visual-semantic domain gap and seen-unseen bias. However, most existing methods directly use the feature extraction models which have been trained on ImageNet only, and ignore the cross-dataset bias between ImageNet and GZSL benchmarks. While, such a bias will inevitably result in poor-quality visual features for GZSL task, which potentially limit the recognition performance both in seen and unseen classes. In this paper, we propose a simple yet effective GZSL method, termed **F**eature **RE**fin**E**ment for the generalized zero-shot learning (**FREE**), to tackle the above problem. FREE employs a well-designed Feature Refinement (**FR**) module that incorporates semantic→visual mapping into a unified generative model to refine the visual features of seen and unseen class samples. Furthermore, we propose a Self-Adapt Margin Center Loss (**SAMC-Loss**) cooperated with a semantic cycle-consistency loss to guide FR to learn class- and semantic-relevant representations, and thus the learned visual features can be improved. Extensive experiments on four benchmark datasets demonstrate that the significant performance gain of FREE over current state-of-the-art methods and the compared baseline has been made.*

1. Introduction

A key challenge of tracking artificial intelligence to human-level intelligence is to generalize machine learning models from seen knowledge to unseen scenarios. In fact, Zero-Shot Learning (ZSL) is a typical research topic targeting on this goal [24, 26, 41]. The aim of ZSL is to classify the images of unseen classes by constructing the mapping relationship between semantic and visual domains. It usually bases on a hypothesis that both seen and unseen classes

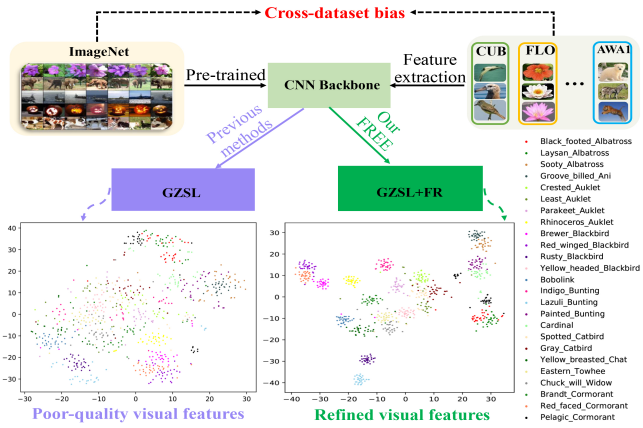


Figure 1. The core idea of our FREE. The cross-dataset bias between ImageNet and GZSL benchmarks (e.g., CUB) is harmful for feature extraction of GZSL benchmarks, which results in poor-quality visual features for unsatisfying performance in GZSL. Our FREE refines the visual features and improves *semantic→visual* mapping using FR in a unified network for GZSL classification.

can be described through a set of semantic vectors in the same semantic space, e.g., word embeddings [36], sentence embeddings [44], and attribute vectors [25]. According to the classification range, ZSL methods can be categorized into conventional ZSL (CZSL) and generalized ZSL (GZSL) [58]. CZSL aims to predict unseen classes, while GZSL can predict both seen and unseen classes. Indeed, recently, GZSL has attracted more attention as it is realistic and more challenging. We are interested in GZSL settings in this paper.

Currently, GZSL has achieved promising progress when many efforts have been contributed to focus on the problems of *visual-semantic domain gap* [25, 1, 2, 52, 51, 61] and *seen-unseen bias* [57, 38, 64, 62, 49, 39, 37, 19]. The semantic embedding methods [33, 7, 30, 64, 34] and the generative methods (e.g., CVAEs [3, 47], GANs [57, 30, 60, 63, 21, 51], VAEGANs [18, 59], and generative flows [49]) are typically

*Corresponding author

applied to mitigate these challenges. GZSL usually extracts visual features of coarse- and fine-grained benchmarks (e.g., AWA1 [25] and CUB [53]) from the CNN backbone (e.g., ResNet-101 [16], GoogleNet [50]) pre-trained on ImageNet [45]. However, there are few GZSL methods to consider the problem of the cross-dataset bias between ImageNet and GZSL benchmarks resulting in poor-quality visual features, which potentially limits their recognition performance both in seen and unseen classes. We attribute it to an open issue in GZSL, i.e., *cross-dataset bias*, which will be tackled in this paper.

We argue that the unsatisfying performance in GZSL comes down to the cross-dataset bias. As shown in Fig. 1, the visual feature instances of different categories are ambiguous, and especially the features of subclasses are more serious. It reveals that the visual features extracted by a pre-trained CNN backbone is unable to effectively represent and distinguish samples for the GZSL task. The inherent reason is that there has a tremendous cross-dataset bias between ImageNet and the GZSL benchmarks (e.g., CUB), especially for fine-grained benchmarks. Thus, it is an unwise way that we directly transfer the knowledge from ImageNet to the new dataset for the GZSL without any further sequential learning. Although fine-tuning may alleviate this problem to a certain degree, it inevitably results in other severer problems including inefficiency and overfitting [17, 27]. Thus, how to essentially circumvent the problem of the cross-dataset bias becomes very necessary.

In light of the above observation, to circumvent these challenges, we propose a novel GZSL method, termed *Feature Refinement for zero-shot learning* (**FREE**), to further boost the performance of GZSL. In essence, FREE refines visual features in a unified generative model, which simultaneously benefits for *semantic→visual* learning, feature synthesis, and classification. Specifically, we take the f-VAEGAN [59] as a baseline to learn *semantic→visual* mapping. To improve the visual features of seen and unseen class samples, We employ a well-designed *Feature Refinement* (**FR**) module, which can jointly optimize with f-VAEGAN and effectively avoid the drawbacks of fine-tuning. Considering that the class label information is available, we introduce a *Self-Adapt Margin Center Loss* (**SAMC-Loss**) to guide FR to learn discriminative class-relevant features. Thus, the distributions of different classes can be easily separated, as shown in Fig. 1. Notably, SAMC-Loss explicitly encourages intra-class compactness and inter-class separability adapting to different datasets, i.e., coarse-grained and fine-grained. To better learn semantic-relevant and more discriminative visual features, a semantic cycle-consistency loss is also added after the restitution of features. From the residual information [16], we further concatenate the discriminative features learned with the respective visual features before the final classification.

To summarize, this paper makes the following salient contributions: **(1)** We propose a novel GZSL method, termed *Feature Refinement for Zero-Shot Learning* (*FREE*), to circumvent the problem of the cross-dataset bias, which can further boost the performance of GZSL. To achieve this goal, a well-designed *feature refinement* (*FR*) module that cooperates with *semantic→visual* mapping in a unified framework is explored and, more importantly, the two modules can be jointly optimized. **(2)** We propose a *Self-Adapt Margin Center Loss* (*SAMC-Loss*) to explicitly encourage intra-class compactness and inter-class separability. Meanwhile, SAMC-Loss cooperates with a semantic cycle-consistency constraint to enable FR to learn more discriminative class- and semantic-relevant representations, especially effective for GZSL. **(3)** The extensive experimental results on four benchmarks, i.e., CUB, FLO, AWA1, and AWA2, clearly demonstrate the advantages of the proposed FREE over current state-of-the-art methods and the compared baseline.

2. Related Work

GZSL for Visual-Semantic Domain Gap. The required knowledge transfer from seen classes to unseen classes for GZSL relies on semantic embedding. One key task is to bridge visual domain and semantic domain [25, 1, 2, 52, 11, 51]. Since visual features in various forms may convey the same concept, the distribution of instances in visual space is often distinct from that of their underlying semantics in semantic space. Thus there is typically a gap between the two domains, which is the visual-semantic domain gap problem of GZSL [52]. Common space learning is a generic methodology towards bridging the visual-semantic gap [6, 65, 56, 52, 15]. This methodology learns a common latent embedding where both visual features and semantic representations are projected for effective knowledge transfer. Consequently, GZSL is achieved in this learned common representation space. Model parameter transfer is another GZSL methodology [5, 12, 20]. It estimates model parameters for unseen classes by combining those model parameters learned from seen classes via mining the inter-class relationship between seen and unseen classes in semantic space. Direct mapping is currently the most popular GZSL methodology to present visual-semantic relations. It learns a mapping function from visual features to semantic representations directly or indirectly [1, 2, 11, 57, 59, 39, 18, 31, 7]. Early mapping works [1, 2] is carried out via either a classifier or a regression model depending upon an adopted semantic representation. Most of recent mapping works are based on generative model (e.g., CVAEs [3, 47, 34], GANs [57, 30, 28, 42, 46, 63], VAEGANs [18, 59, 39], and generative flows [49]), which not only learns visual-semantic mapping, but also it can generate a great number of feature samples of unseen classes for data augmentation.

GZSL for Seen-Unseen Bias. GZSL methods inevitably fall in seen-unseen bias problem [47, 59, 49, 4, 49, 33, 64,

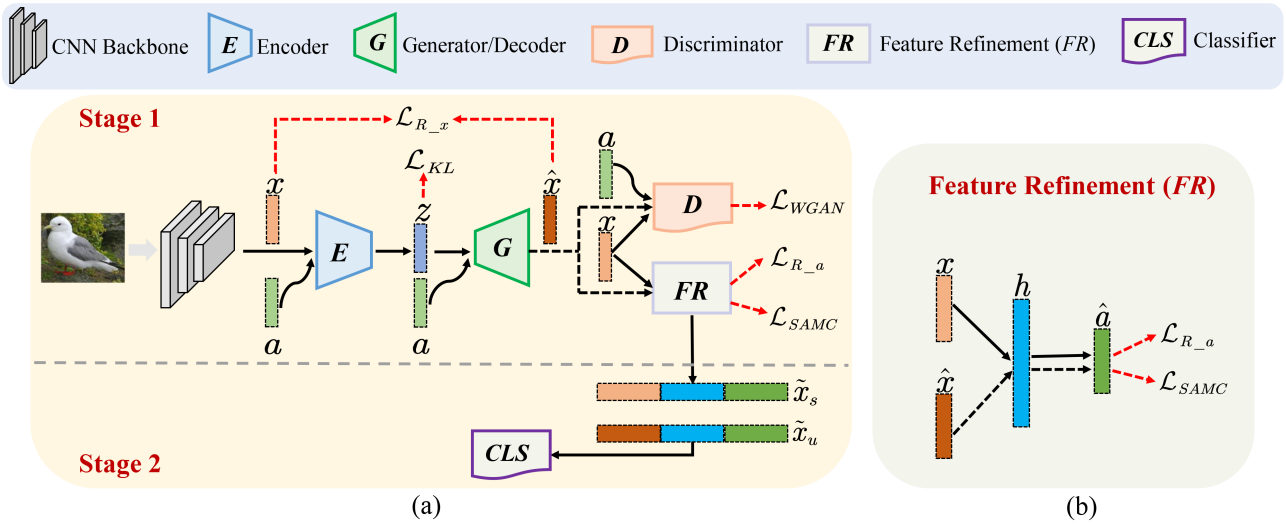


Figure 2. A schematic overview of FREE. (a) FREE consists of a feature generating VAEGAN (f-VAEGAN)[59], a feature refinement (FR) module, and a classifier. In Stage 1, f-VAEGAN aims to learn the *semantic→visual* mapping (G) for visual feature generation, and FR targets to learn discriminative representations for the real/synthesized seen visual features. They are jointly optimized. In Stage 2, we respectively refine the real seen visual features and the real/synthesized unseen visual features with the trained FR, and they are then fed into a classier for classification. (b) Proposed FR module. It learns discriminative features utilizing class label supervision (SAMC-Loss $\mathcal{L}_{SAMC}$) and semantic constraint (semantic cycle-consistency loss $\mathcal{L}_{R\_a}$).

10, 55, 20, 29]. As only seen data are involved during training, most of generative GZSL models usually leads to overfitting on seen classes [33, 8, 37], i.e., generated unseen data tend to have the same distribution as seen ones. This would heavily hamper the classification performance of unseen classes. Notably, the generative GZSL methods more easily drop in seen-unseen bias. In [33], Liu proposed a Deep Calibration Network (DCN) to enable simultaneous calibration of deep networks on the confidence of source classes and the uncertainty of target classes. Zhang *et al.* [64] employs a co-representation network to learn a more uniform visual embedding space, which effectively bypassed the bias problems and helped with classification. Maximum mean discrepancy (MMD) based methods optimized the distribution between real seen features and synthesized unseen features to tackle the bias problem explicitly [4, 49].

Although GZSL has achieved promising progress when many efforts have been contributed to focus on the problems mentioned above, another essential problem of the cross-dataset bias between ImageNet and GZSL benchmarks should also be concentrated. The cross-dataset bias results in poor-quality visual features of benchmarks, which potentially hampers the recognition performance and upper limit of GZSL classification.

3. Method

Motivation. As shown in Fig. 1, the cross-dataset bias results in poor-quality visual features of GZSL benchmark (e.g., CUB [53]) extracted by a CNN backbone pre-trained on ImageNet. It will hamper the *semantic→visual* learning, feature synthesis, and classification in the GZSL task. As a result, the upper bound of recognition performance of GZSL both in seen and unseen classes is potentially limited. Although fine-tuning may alleviate this challenge to a certain degree, some more serious problems including inefficiency and overfitting will be induced [17, 43, 27]. Generally, fine-tuning transforms GZSL into a non end-to-end model, which is unable to learn the whole pipeline. In addition, it is difficult to fine-tune on a new small dataset, and it is easy to cause the model to overfit the seen classes, which ultimately is not conducive to the generalization of GZSL.

This observation prompts us to speculate that the current poor performance of GZSL is closely related to the cross-dataset bias. In other words, we think that, by alleviating the cross-dataset bias, the visual features of GZSL benchmarks will be progressed, and thus we can also improve the GZSL classification further. To this end, we proposed a novel method, termed *Feature Refinement for Generalized Zero-Shot Learning* (**FREE**). Our strategy towards achieving this goal is to utilize class labels supervision and semantic cycle-consistency constraint to guide the proposed *Feature Refinement* (**FR**) module to learn class- and semantic-relevant feature representations in a unified network. Furthermore, FREE can effectively refine the visual features and avoids the inefficiency and overfitting risks of fine-tuning.

Overview. The pipeline of FREE is shown in Fig. 2, which

consists of a feature generating VAEGAN (f-VAEGAN) [59], a feature refinement module (FR), and a classifier. In Stage 1, f-VAEGAN aims to learn the *semantic→visual* mapping (G) for visual feature synthesis, and FR targets to learn discriminative representations for the real/synthesized seen visual features. They are jointly optimized. In stage 2, the refined real seen visual features and the refined real/synthesized unseen visual features using FR are fed into a classier, i.e., k-nearest neighbor (k-NN), for classification. To learn discriminative features, FR is optimized by using a self-adapt margin center loss and a semantic cycle-consistency loss.

Notation. We denote seen data as $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{M}$, where x_i is an visual feature vector, y_i is its class label in $\mathcal{Y}^s$, and M is the number of seen data. Let $\mathcal{Y}^u$ be the set of unseen classes, which is disjoint from the seen class set $\mathcal{Y}^s$, i.e., $\mathcal{Y}^s \cap \mathcal{Y}^u = \varnothing$. Each seen class and unseen class have their own conresponding attribute embedding $a_j \in \mathcal{A}, \forall j \in \mathcal{Y}^s \cup \mathcal{Y}^u$.

3.1. Revisiting f-VAEGAN

f-VAEGAN [59] has achieved impressive performance and thus became a popular baseline of generative GZSL methods. In this paper, we take f-VAEGAN as a baseline for learning *semantic→visual* mapping. f-VAEGAN integrates VAE [23] and GAN [14] into a unified generative model to simultaneously take advantages of them. Thus it comprises a feature generating VAE (f-VAE) and a feature generating WGAN (f-WGAN). Specifically, f-VAE consists of an encoder $E(x, a)$ (denoted as E) and a decoder G (shared with f-WGAN, as a conditional generator $G(z, a)$). The encoder $E(x, a)$ encodes an input seen visual feature x to a latent code z, while the decoder $G(z, a)$ reconstructs visual feature $\hat{x}$ from z. f-VAE is first optimized by a vae loss $\mathcal{L}_V$:

$$\begin{aligned}\mathcal{L}_V &= \mathcal{L}_{KL} + \mathcal{L}_{R\_x} \\ &= \mathrm{KL}(E(x,a)\|p(z \mid a)) - \mathbb{E}_{E(x,a)}[\log G(z,a)],\end{aligned} \tag{1}$$

where $\mathcal{L}_{KL}$ is the Kullback-Leibler divergence, $p(z \mid a)$ is a prior distribution assumed to be $\mathcal{N}(0, 1)$, and $\mathcal{L}_{R\_x}$ is visual feature reconstruction loss represented by $-\log G(z, a)$. The feature generating network (f-WGAN) comprises a generator $G(z, a)$ and a discriminator $D(x, a)$ (denoted as D). The generator $G(z, a)$ synthesizes a visual feature $\hat{x}$ from a random input noise z, whereas the discriminator $D(x, a)$ takes an input visual feature x or a synthesized visual feature $\hat{x}$ and outputs a real value indicating the degree of realness or fakeness of the input features. Both G and D are conditioned on the embedding a, optimized by the WGAN loss

$$\begin{aligned}\mathcal{L}_W = \mathbb{E}[D(x,a)] - \mathbb{E}[D(\hat{x},a)] \\ - \lambda\mathbb{E}\left[\left(\|\nabla D(x',a)\|_2 - 1\right)^2\right],\end{aligned} \tag{2}$$

where $x' = \tau x + (1-\tau)\hat{x}$ with $\tau \sim U(0, 1)$ and λ is the penalty coefficient.

3.2. Feature Refinement

Our strategy towards circumventing the cross-dataset bias is to refine the visual features of GZSL benchmarks with Feature Refinement (FR), which is constrained by self-adapt margin center loss and semantic cycle-consistency loss.

Self-Adapt Margin Center Loss. Considering the class label information is an important side-knowledge and the success of center loss in image classification [54, 21], we propose a Self-Adapt Margin Center Loss (SAMC-Loss) $\mathcal{L}_{SAMC}$ to explicitly encourage intra-class compactness and inter-class separability (especially in the subclasses) inspired by triplet loss [48]. Thus FR learns more discriminative class-relevant representations for visual features. $\mathcal{L}_{SAMC}$ is formulated as

$$\begin{aligned}\mathcal{L}_{SAMC}(\hat{a}, y, y') = \max\Big(0, \Delta + \gamma\|\hat{a} - \mathbf{c}_y\|_2^2 \\ -(1-\gamma)\|\hat{a} - \mathbf{c}_{y'}\|_2^2\Big),\end{aligned} \tag{3}$$

where $\mathbf{c}_y$ is the yth (the label of seen visual feature x) class center of semantic embedding, $\mathbf{c}_{y'}$ is the y'th (a randomly-selected class label other than y) class center of semantic embedding, Δ represents the margin to control the distance between intra-class and inter-class pairs, $\hat{a}$ is the synthesized semantic vector and $\gamma \in [0, 1]$ is used for balancing the inter-class separability and intra-class compactness.

We can use a large γ for fine-grained dataset (e.g., CUB, SUN), and a small γ for coarse-grained datasets (e.g., AWA1 [25], AWA2 [58]). The reasons for such setting are (1) when the classes are ambiguous in a fine-grained dataset, we are more easy to distinguish them by encouraging the intra-class compactness, as shown in Fig. 7(a); (2) when the classes are confused in a coarse-grained dataset, we can effectively seperate them by enlarge the inter-class separability, as shown in Fig. 7(b).

Semantic Cycle-Consistency Loss. The last layer of FR reconstructs the semantic embedding a from $\hat{x}$ or x. To further guide FR to effectively learn semantic-relevant representations, we enforce a semantic cycle-consitency loss [11, 39] on the reconstructed semantic embeddings to ensure that the synthesized semantic vector $\hat{a}$ are transformed to be the same embeddings that generated them. To this end, semantic-relevant features are learned using FR. The semantic cycle-consistency loss $\mathcal{L}_{R\_a}$ is achieved using the ℓ_1 reconstruction loss, formulated as follows:

$$\mathcal{L}_{R\_a} = \mathbb{E}\left[\|\hat{a}_{real} - a\|_1\right] + \mathbb{E}\left[\|\hat{a}_{syn} - a\|_1\right], \tag{4}$$

where $\hat{a}_{real}$ is the synthesized semantic-relevant features from x using FR, $\hat{a}_{syn}$ is the synthesized semantic-relevant features from $\hat{x}$, $\hat{a} = \hat{a}_{real} \cup \hat{a}_{syn}$, a is the semantic embeddings corresponded to visual features x and $\hat{x}$.

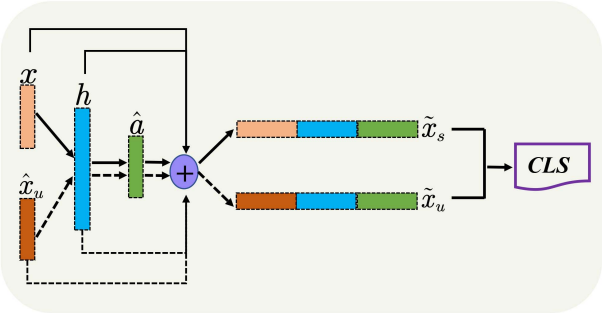


Figure 3. Visual features refinement for real seen visual features and synthesized unseen visual features with FR.

3.3. Optimization

We jointly train the encoder (E), generator (G), discriminator (D), and feature refinement (FR) to optimize the total objective, which is a weighted sum of the following losses:

$$\mathcal{L}_{total}(E,G,D,FR) = \mathcal{L}_V + \mathcal{L}_W + \lambda_{SAMC}\mathcal{L}_{SAMC} + \lambda_{R\_a}\mathcal{L}_{R\_a}, \tag{5}$$

where λ_{SAMC} and λ_{R_a} are the weights to control the importance of related loss terms. Similar to the alternative updating policy for GANs, we we alternatively train E, G before the generated visual features and the E, D and FR after the generated visual features. This is a joint learning framework that end-to-end couples *semantic→visual* mapping and visual feature refinement in a unified network. There exists an online interaction between them to benefit the two tasks for GZSL mutually.

3.4. Classification

After training at the Stage 1, we refine the real seen visual features x and synthesized unseen visual feature $\hat{x}$ into discriminative features $\tilde{x}_s$ and $\tilde{x}_u$, respectively. FR transforms the original high-dimension features into low dimensional features that inevitably discards some other discriminative information, which may hamper the GZSL classification performance. From the residual information [16], we concatenate the the visual features x and $\hat{x}$ with the corresponding latent embedding $h_s, h_u \in \mathcal{H}$ and semantic-relevant embedding $\hat{a}_s, \hat{a}_u \in \mathcal{A}$ learned from FR, as shown in Fig. 3. It is formulated as:

$$\tilde{x}_s = x \oplus h_s \oplus \hat{a}_s \tag{6}$$

$$\tilde{x}_u = \hat{x}_u \oplus h_u \oplus \hat{a}_u \tag{7}$$

where $\oplus$ is the operation of *concatentation*, $\tilde{x}_s$ and $\tilde{x}_u \in \tilde{\mathcal{X}}$. Thus, the visual feature is refined as a discriminative feature, which is class- and semantic-relevant for reducing ambiguities among features instances of different categories.

Table 1. CUB, FLO, AWA1 and AWA2 datasets, in terms of the dimensions of semantic vectors per class (Att), seen class size/unseen class size (S/U), and total number of images (Img).

	Att	**S/U**	**Img**
CUB	312	150/50	11788
FLO	1024	82/20	8189
AWA1	85	40/10	30475
AWA2	85	40/10	37322

Once we have the training data, we train a supervised classifier in the refined feature space as the final GZSL classifier. We adopt a simple classifier k-nearest neighbor (k-NN) for classification (denoted as Stage 2 in Fig. 2(a)). GZSL aims to learn the classifier $f_{gzsl}(k-NN) : \tilde{\mathcal{X}} \rightarrow \mathcal{Y}^s \cup \mathcal{Y}^u$. During testing, the test seen/unseen features are refined as new features by FR, and they are further used for testing.

4. Experiments

Datasets. We evaluate our method on four benchmark datasets, i.e., CUB (Caltech UCSD Birds 200) [53], Oxford Flowers [40], AWA1 (Animals with Attributes 1) [25], and AWA2 (Animals with Attributes 2)[58]. Among those, CUB and FLO are fine-grained datasets, whereas AWA1 and AWA2 are coarse-grained datasets. For fair comparison, we use the same seen/unseen splits and class embeddings following [58], as summarized in Table 1.

Evaluation Protocols. During testing, we use the unified evaluation protocol proposed in [58] for faciliting direct comparison. Since the test set is composed of seen classes ($\mathcal{Y}^s$) and unseen classes ($\mathcal{Y}^u$), we evaluate the Top-1 accuracies on seen classes (denoted as **S**) and unseen classes (denoted as **U**), respectively. Furthermore, Their harmonic mean (defined as $\mathbf{H} = (2 \times S \times U)/(S+U)$) is also used for evaluating the performance of GZSL.

Implementation Details. In FREE, the enconder, generator and discriminator are MLP that contains a 4096-unit hidden layer with LeakyReLU activation. The FR consists of a hidden layer h with 4096-unit activated by LeakyReLU, and the dimension of its output layer $\hat{a}$ is corresponding to the semantic vector according to different datasets (e.g., $|\hat{a}| = 312$ for CUB). We use Adam optimizer [22] with β_1 = 0.5, β_2 = 0.999. The visual features are extracted from 2048-dim top-layer pooling units of the ResNet-101 pre-trained on ImageNet following [57]. The penalty coefficient λ is set as 10. We empirically set the loss weights λ_{SAMC} and λ_{R_a} as 0.5 and 0.1 for all datasets, respectively. We take 5-NN as our classifier. More details can be found in appendix.

4.1. Comparison with State-of-the-Arts

GZSL can be classified as inductive methods that only utilize the labeled source data and transductive methods that assume the access of unlabeled target data [59, 58]. Since

Table 2. State-of-the-art comparison on four datasets. The best and the second best results are marked in **red** and **blue** respectively.

	Method	AWA1			AWA2			CUB			FLO		
		U	*S*	*H*	*U*	*S*	*H*	*U*	*S*	*H*	*U*	*S*	*H*
Non-generative	DCN(NeurIPS'18) [33]	25.5	84.2	39.1	-	-	-	28.4	60.7	38.7	-	-	-
	SP-AEN(CVPR'18) [7]	23.3	**90.9**	37.1	-	-	-	34.7	70.6	46.6	-	-	-
	AREN(CVPR'19) [60]	-	-	-	15.6	92.9	26.7	38.9	**78.7**	52.1	-	-	-
	Kai at el.(ICCV'19) [30]	**62.7**	77.0	69.1	-	-	-	47.4	47.6	47.5	-	-	-
	CRnet(ICML'19) [64]	58.1	74.7	65.4	-	-	-	45.5	56.8	50.5	-	-	-
	GAFE(IJCAI'19) [34]	25.5	76.6	38.2	26.8	78.3	40.0	22.5	52.1	31.4	-	-	-
	PQZSL(CVPR'19) [29]	-	-	-	31.7	70.9	43.8	43.2	51.4	46.9	-	-	-
	MLSE(CVPR'19) [10]	-	-	-	23.8	83.2	37.0	22.3	71.6	34.0	-	-	-
	TCN(ICCV'19) [20]	49.4	76.5	60.0	61.2	65.8	63.4	52.6	52.0	52.3	-	-	-
	RGEN(ECCV'20) [61]	-	-	-	**67.1**	76.5	**71.5**	**60.0**	73.5	**66.1**	-	-	-
	DVBE(CVPR'20) [37]	-	-	-	63.6	70.8	67.0	53.2	60.2	56.5	-	-	-
	DAZLE(CVPR'20) [19]	-	-	-	60.3	75.7	67.1	59.6	56.7	58.1	-	-	-
Generative	SE-GZSL(CVPR'18) [3]	56.3	67.8	61.5	58.3	68.1	62.8	41.5	53.3	46.7	-	-	-
	f-CLSWGAN(CVPR'18) [57]	57.9	61.4	59.6	-	-	-	43.7	57.7	49.7	59.0	73.8	65.6
	cycle-CLSWGAN(ECCV'18) [11]	56.9	64.0	60.2	-	-	-	45.7	61.0	52.3	59.2	72.5	65.1
	CADA-VAE(CVPR'19) [47]	57.3	72.8	64.1	55.8	75.0	63.9	51.6	53.5	52.4	-	-	-
	SABR(CVPR'19) [42]	-	-	-	30.3	**93.9**	46.9	55.0	58.7	56.8	-	-	-
	f-VAEGAN(CVPR'19) [59]	-	-	-	57.6	70.6	63.5	48.4	60.1	53.6	56.8	74.9	64.6
	LisGAN(CVPR'19) [28]	52.6	76.3	62.3	-	-	-	46.5	57.9	51.6	57.7	**83.8**	68.3
	GMN(CVPR'19) [46]	61.1	71.3	65.8	-	-	-	56.1	54.3	55.2	-	-	-
	E-PGN(CVPR'20) [63]	62.1	83.4	**71.2**	52.6	83.5	64.6	52.0	61.1	56.2	**71.5**	82.2	**76.5**
	OCD-CVAE(CVPR'20) [21]	-	-	-	59.5	73.4	65.7	44.8	59.9	51.3	-	-	-
	LsrGAN(ECCV'20) [51]	54.6	74.6	63.0	-	-	-	48.1	59.1	53.0	-	-	-
	IZF(ECCV'20) [49]	61.3	80.5	69.6	60.6	77.5	68.0	52.7	68.0	59.4	-	-	-
	TF-VEAGAN(ECCV'20) [39]	-	-	-	59.8	75.1	66.6	52.8	64.7	58.1	62.5	84.1	71.7
	FREE (Ours)	**67.3**	**96.4**	**79.3**	**69.3**	**97.2**	**80.9**	**62.5**	**82.6**	**71.1**	**64.5**	**99.5**	**78.3**

our FREE is an inductive method, we compare it with other state-of-the-art inductive methods for a fair comparison.

We category the compared methods into generative methods and non-generative methods. Among the non-generative GZSL methods, we select several competitive methods including DCN [33], SP-AEN [7], AREN [60], Kai *at el.* [30], CRnet [64], GAFE [34], PQZSL [29], MLSE [10], TCN [20], RGEN [61], DVBE [37], DAZLE [19]. Mearnwhile, the compared generative GZSL methods including SE-GZSL [3], f-CLSWGAN[57], cycle-CLSWGAN [11], CADA-VAE [47], SABR [42], f-VAEGAN [59], LisGAN [28], GMN [46], E-PGN [63], OCD-CVAE [21], LsrGAN [51], IZF [49], TF-VEAGAN [39]. We report the official results of these methods from referenced articles.

Table 2 shows the Top-1 accuracies on unseen classes (**U**), seen classes (**S**) and their harmonic mean (**H**) of different methods. The results show that FREE consistently attains the best performance for harmonic mean on all benchmarks, i.e., 79.3 on AWA1 (the second-best results 71.2 for E-PGN), 80.9 on AWA2 (the second-best results 71.5 for RGEN), 71.1 on CUB (the second-best results 66.1 for RGEN), and 78.3 on FLO (the second-best results 76.5 for E-PGN). It indicates that the refined features are discriminative and generic for seen classes/unseen classes on both coarse- and fine-grained datasets.

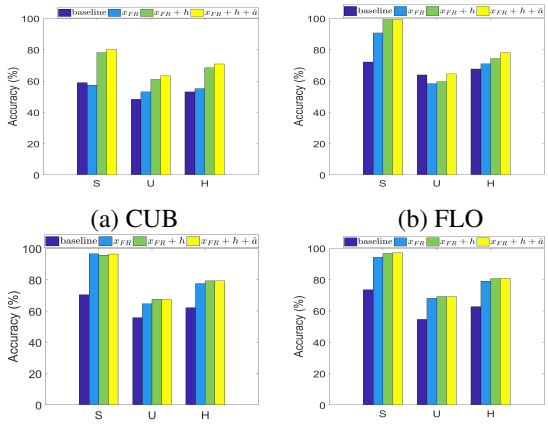
Notably, different from the compared state-of-the-art methods that only achieves good performance on seen classes or unseen classes, FREE attained promising results both in seen classes and unseen classes. On **AWA1**, FREE achieves 67.3/96.4 for U/S, while the second-best results are gained by Kai *at el.* [30] (U/S=62.7/77.0) and SP-AEN (U/S=23.3/90.9). The similar results are achieved on **AWA2** (FREE: U/S=69.3/97.2; the second-best methods RGEN: U/S=67.1/76.5 and SABR: U/S=30.3/93.9), **CUB** (FREE:U/S=62.5/82.6; the second-best methods RGEN: U/S=60.0/73.5 and AREN: U/S=38.9/78.7), and **FLO** (FREE:U/S=64.5/99.5; the second-best methods E-PGN: U/S=71.5/82.2 and LisGAN: U/S=57.7/83.8). Compared to the state-of-the-art methods that have obvious bias between seen and unseen classes, FREE consistently achieves the best performance both in seen and unseen classes on all datasets. It reveals that FREE maintains a good balance between seen and unseen classes, which is benefit from the unified model that is jointly trained for *semantic→visual* mapping and FR. To this end, they are likely to encode complementary information of the categories and encourage the two models to learn discriminative representations.

4.2. Ablation Study

To show insights about FREE, we conduct ablation studies to evaluate the effect of different model components and feature components. Since FREE bases on f-VAEGAN [59], we re-implemented this method, labeled as *baseline*. The results that we obtained from our baseline are very similar

Table 3. Ablation studies of the FREE components on CUB and AWA2 datasets. Our implementation of f-VAEGAN [59] labeled as *baseline*. The best results are marked in **boldface**.

Method	CUB			AWA2		
	U	S	H	U	S	H
f-VAEGAN [59]	48.4	60.1	53.6	57.6	70.6	63.5
baseline	48.3	58.9	53.1	54.6	73.6	62.7
baseline+FR($\mathcal{L}_{R\_a}$)	59.8	74.6	66.6	68.6	96.2	80.1
baseline+FR($\mathcal{L}_{SAMC}$)	**62.6**	81.2	70.7	64.6	95.5	76.3
baseline+FR($\mathcal{L}_{SAMC}$+$\mathcal{L}_{R\_a}$)	62.5	**82.6**	**71.1**	**69.3**	**97.2**	**80.9**



(a) CUB (b) FLO

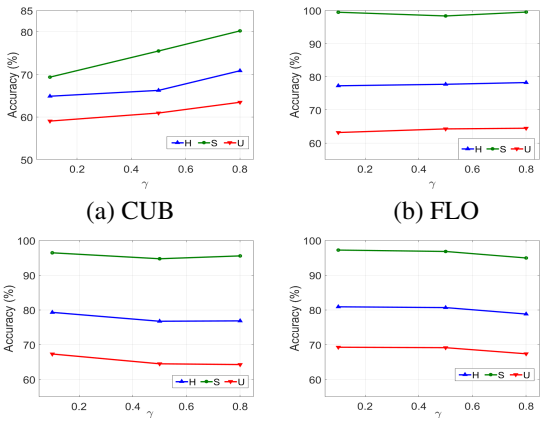
(c) AWA1 (d) AWA2

Figure 4. The effectiveness of various feature components of visual features refined by FR.

to the reported results in [59].

Analysis of Model Components. As shown in Table 3, our FR is found to largely improve over the baseline with various model components (i.e., SAMC-Loss $\mathcal{L}_{SAMC}$, semantic cycle-consistency loss $\mathcal{L}_{R\_a}$ and both them). We first evaluate the two components independently. The SAMC-Loss and semantic cycle-consistency loss individually outperform the baseline for harmonic mean on CUB (by 17.6% and 13.5%) and AWA2 (by 13.6% and 17.4%). Interestingly, since the semantic representations of some classes in the fine-grained dataset (e.g., CUB) is similar to each other [19, 51, 9], the semantic cycle-consistency performs inferior on the fine-grained dataset than it achieved on the coarse-grained dataset (e.g., AWA2). SAMC-Loss simultaneously performs well, benefitting from the class-relevant constraint. The complete version of FREE gives the highest results on all datasets, achieving a whopping accuracy gain of 14.2% and 14.7% for unseen accuracy, 23.7% and 23.6% for seen accuracy, and 18.0% and 18.2% for harmonic mean on CUB and AWA2 respectively. It indicates that the SAMC-Loss and semantic cycle-consistency loss are mutually complementary for feature refinement. These results clearly prove that FR is good for feature progress in GZSL, and thus the cross-dataset bias is well circumvented.

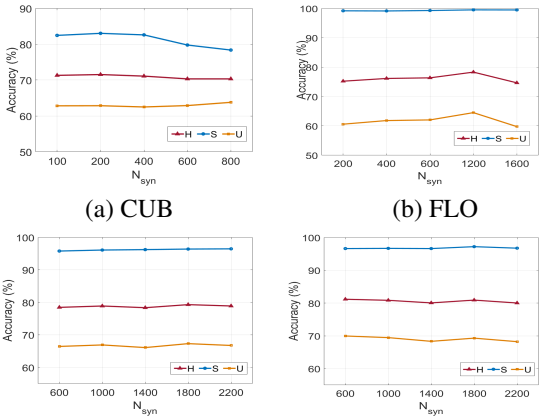
Analysis of Feature Components. We study the effectiveness of various feature components of visual features refined by FR. As shown in Fig. 4, different feature components in FR substantially achieve significant performance. When we only take the real seen features x and the synthesized unseen visual features $\hat{x}_u$ for classification, labeled as x_{FR}, FREE performs consistent improvement over the baseline. This proves that FR contributes positive effectiveness for *semantic→visual* mapping. We also concate seen/unseen visual features ($x/\hat{x}$) with the hidden features h (labeled as $x_{FR}+h$) and the learned semantic-relevant features $\hat{a}$ into $\tilde{x}_s/\tilde{x}_u$ (labeled as $x_{FR}+h+\hat{a}$) for classification, the further improvement is achieved. This benefits from that FR can learn the class- and semantic-relevant feature representations using SAMC-Loss and semantical cycle-consistency loss. Thus, the visual features are further refined.



(a) CUB (b) FLO

(c) AWA1 (d) AWA2

Figure 5. The effectiveness of balance factor γ for SAMC-Loss.



(a) CUB (b) FLO

(c) AWA1 (d) AWA2

Figure 6. The impact of the number of synthetic visual features N_{syn} for per unseen class.

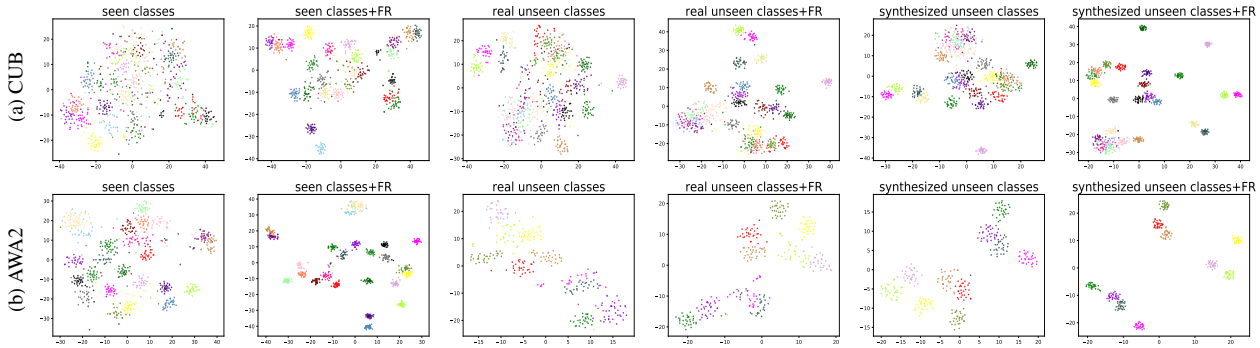


Figure 7. t-SNE visualization [35] of visual feature for real seen classes, real/synthesized unseen classes, and their refined features by FR on (a) CUB and (b) AWA2. The real unseen classes features and the synthesized unseen classes features are represented with same color. Different colors denote different classes. More results on FLO and AWA1 can be found in Appendix D.

4.3. Hyper-Parameter Analysis

Balance Factor γ. We study the effectiveness of balance factor γ in Eq. 3 to obtain a intuitive observation on the module influence. See from Fig. 5, as γ grows, **S**, **U** and **H** gain consistently improvement on the fine-grained datasets (e.g., CUB and FLO). Nevertheless, **S**, **U** and **H** consistently decreases while γ increases on coarse-grained datasets (e.g., AWA1, AWA2). The results are analyzed as follows: (1) In the fine-grained datasets, to compact the intra-class will achieve larger gains while the classes is confused. (2) In the coarse-grained datasets, to seperate the inter-class will significantly benefit for classifying the ambiguous classes. This supports the claim in Section 3.2.

Number of Synthesized Visual Features. We evaluate the impact of the number of synthetic visual features per unseen classes (denoted as N_{syn}). As shown in Fig. 6, we observe that FREE is generally insensitive to N_{syn} on all datasets. When increasing the synthetic features, the seen class accuracy drops slightly and the unseen class accuracy improves. This demonstrates that FREE can also alleviate the seen-unseen bias problem. Notably, if N_{syn} is set too large, all the evaluation metrics will drop down because there exists an upper bound of synthetic diversity. This is also a general challenge for generative GZSL methods, which could significantly be for our future work.

5. Discussion

Intuitive Analysis for FREE. As displayed in Fig. 1, we intuitively show that the cross-dataset bias results in poor-quality visual features, which potentially limits recognition performance both seen and unseen classes of GZSL. In this paper, we attempt to refine the visual features of seen/unseen classes to circumvent this challenge using the feature refinement (FR), which is encouraged to learn class- and semantic-relevant feature representations. See Fig. 7, FR significantly progresses the visual features of seen/unseen classes, which reduces the ambiguity between different categories. Interestingly, the synthesized unseen features share similar class relationships with the real unseen features, which proves that FREE learns a promising *semantic→visual* mapping. Thus, FREE achieves the large-margin performance gain over the state-of-the-art methods and baseline.

Table 4. Feature refinement vs fine-tuning. Experiment results on CUB and AWA2 datasets. The best results are marked in **boldface**.

Method	CUB			FLO			AWA2		
	U	*S*	*H*	*U*	*S*	*H*	*U*	*S*	*H*
f-VAEGAN+finetuned [59]	**63.2**	75.6	68.9	63.3	92.4	75.1	57.1	76.1	65.2
f-VAEGAN+FR	62.5	**82.6**	**71.1**	**64.5**	**99.5**	**78.3**	**69.3**	**97.2**	**80.9**

Feature Refinement vs Fine-tuning. As analyzed in Section 1, although fine-tuning may alleviate the cross-dataset bias for GZSL to a degree, it encounters the ineffectiveness and overfitting risks. Nevertheless, our proposed FR cooperatives with *semantic→visual* mapping and mutually benefits from each other. Meanwhile, the SAMC-Loss and semantic cycle-consistency loss guide FR to learn discriminative feature representations. Thus, FR significantly outperforms the fine-tuning, as shown in Table 4. FR is a promising substitute for fine-tuning in GZSL and other domain adaptation tasks.

6. Conclusion

In this paper, we have proposed a joint learning framework FREE that end-to-end couples *semantic→visual* mapping and FR to circumvent the problem of the cross-dataset bias. In the FR module, we introduce a SAMC-Loss that cooperates semantic cycle-consistency constraint to encourage FR to learn class- and semantic-relevant feature representations. Meanwhile, we incorporate various features into a new discriminative visual feature for classification. Competitive results on four popular benchmarks demonstrate the superiority and great potentials of our approach. We also believe FR could be well incorporated into other existing ZSL

methods and benefit some other domain adaptation tasks like cross-datasets person re-id [13], zero-shot sketch-based image retrieval [32], etc.

References

[1] Zeynep Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38:1425–1438, 2016. 1, 2

[2] Zeynep Akata, S. Reed, D. Walter, H. Lee, and B. Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, pages 2927–2936, 2015. 1, 2

[3] Gundeep Arora, V. Verma, Ashish Mishra, and P. Rai. Generalized zero-shot learning via synthesized examples. In *CVPR*, pages 4281–4289, 2018. 1, 2, 6

[4] M. Bucher, S. Herbin, and F. Jurie. Generating visual representations for zero-shot classification. In *ICCV Workshop*, pages 2666–2673, 2017. 3

[5] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and F. Sha. Synthesized classifiers for zero-shot learning. In *CVPR*, pages 5327–5336, 2016. 2

[6] Soravit Changpinyo, Wei-Lun Chao, and F. Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *ICCV*, pages 3496–3505, 2017. 2

[7] Long Chen, Hanwang Zhang, Jun Xiao, W. Liu, and S. Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In *CVPR*, pages 1043–1052, 2018. 1, 2, 6

[8] X. Chen, X. Lan, Fu-Chun Sun, and N. Zheng. A boundary based out-of-distribution classifier for generalized zero-shot learning. In *ECCV*, 2020. 3

[9] Zhi Chen, Sen Wang, Jingjing Li, and Zi Huang. Rethinking generative zero-shot learning: An ensemble learning perspective for recognising visual patches. In *ACM MM*, 2020. 7

[10] Z. Ding and Hongfu Liu. Marginalized latent semantic encoder for zero-shot learning. In *CVPR*, pages 6184–6192, 2019. 3, 6

[11] Rafael Felix, B. V. Kumar, I. Reid, and G. Carneiro. Multi-modal cycle-consistent generalized zero-shot learning. In *ECCV*, 2018. 2, 4, 6

[12] Chuang Gan, Ming Lin, Y. Yang, Y. Zhuang, and A. Hauptmann. Exploring semantic inter-class relationships (sir) for zero-shot action recognition. In *AAAI*, 2015. 2

[13] Yixiao Ge, Da peng Chen, and Hongsheng Li. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In *ICLR*, 2020. 9

[14] Ian J. Goodfellow, Jean Pouget-Abadie, M. Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 4

[15] Zongyan Han, Zhenyong Fu, and Jian Yang. Learning the redundancy-free features for generalized zero-shot object recognition. In *CVPR*, pages 12862–12871, 2020. 2

[16] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 2, 5

[17] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *ICML*, 2019. 2, 3

[18] H. Huang, C. Wang, Philip S. Yu, and Chang-Dong Wang. Generative dual adversarial network for generalized zero-shot learning. In *CVPR*, pages 801–810, 2019. 1, 2

[19] D. Huynh and E. Elhamifar. Fine-grained generalized zero-shot learning via dense attribute-based attention. In *CVPR*, pages 4482–4492, 2020. 1, 6, 7

[20] Huajie Jiang, R. Wang, S. Shan, and X. Chen. Transferable contrastive network for generalized zero-shot learning. In *ICCV*, pages 9764–9773, 2019. 2, 3, 6

[21] Rohit Keshari, R. Singh, and Mayank Vatsa. Generalized zero-shot learning via over-complete distribution. In *CVPR*, pages 13297–13305, 2020. 1, 4, 6

[22] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[23] Diederik P. Kingma and M. Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 4

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

[25] Christoph H. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:453–465, 2014. 1, 2, 4, 5

[26] H. Larochelle, D. Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, pages 646–651, 2008. 1

[27] Hao Li, Pratik Chaudhari, Hao Yang, Michael Lam, A. Ravichandran, Rahul Bhotika, and Stefano Soatto. Rethinking the hyperparameters for fine-tuning. In *ICLR*, 2020. 2, 3

[28] J. Li, Mengmeng Jing, K. Lu, Z. Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *CVPR*, pages 7394–7403, 2019. 2, 6

[29] J. Li, X. Lan, Y. Liu, L. Wang, and N. Zheng. Compressing unknown images with product quantizer for efficient zero-shot classification. In *CVPR*, pages 5458–5467, 2019. 3, 6

[30] K. Li, Martin Renqiang Min, and Yun Fu. Rethinking zero-shot learning: A conditional visual classification perspective. In *ICCV*, pages 3582–3591, 2019. 1, 2, 6

[31] Y. Li, D. Wang, Huanhang Hu, Yuetan Lin, and Yueting Zhuang. Zero-shot recognition using dual visual-semantic mapping paths. In *CVPR*, pages 5207–5215, 2017. 2

[32] Qing Liu, Lingxi Xie, Huiyu Wang, and A. Yuille. Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval. In *ICCV*, pages 3661–3670, 2019. 9

[33] Shichen Liu, Mingsheng Long, J. Wang, and Michael I. Jordan. Generalized zero-shot learning with deep calibration network. In *NeurIPS*, 2018. 1, 3, 6

[34] Yang Liu, D. Xie, Quanxue Gao, Jungong Han, S. Wang, and X. Gao. Graph and autoencoder based feature extraction for zero-shot learning. In *IJCAI*, 2019. 1, 2, 6

[35] L. V. D. Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. 8

[36] Tomas Mikolov, Ilya Sutskever, Kai Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, pages 3111–3119, 2013. 1

[37] Shaobo Min, Hantao Yao, Hongtao Xie, Chaoqun Wang, Z. Zha, and Yongdong Zhang. Domain-aware visual bias eliminating for generalized zero-shot learning. In *CVPR*, pages 12661–12670, 2020. 1, 3, 6

[38] Ashish Mishra, M. K. Reddy, A. Mittal, and H. Murthy. A generative model for zero shot learning using conditional variational autoencoders. In *CVPR Workshop*, pages 2269–22698, 2018. 1

[39] Sanath Narayan, A. Gupta, F. Khan, Cees G. M. Snoek, and L. Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *ECCV*, 2020. 1, 2, 4, 6

[40] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, pages 722–729, 2008. 5

[41] Mark Palatucci, D. Pomerleau, Geoffrey E. Hinton, and Tom Michael Mitchell. Zero-shot learning with semantic output codes. In *NeurIPS*, pages 1410–1418, 2009. 1

[42] Akanksha Paul, Narayanan C. Krishnan, and Prateek Munjal. Semantically aligned bias reducing zero shot learning. In *CVPR*, pages 7049–7058, 2019. 2, 6

[43] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio. Transfusion: Understanding transfer learning for medical imaging. In *NeurIPS*, 2019. 3

[44] Scott Reed, Zeynep Akata, H. Lee, and B. Schiele. Learning deep representations of fine-grained visual descriptions. In *CVPR*, pages 49–58, 2016. 1

[45] Olga Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Zhiheng Huang, A. Karpathy, A. Khosla, Michael S. Bernstein, A. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015. 2

[46] Mert Bülent Sariyildiz and Ramazan Gokberk Cinbis. Gradient matching generative networks for zero-shot learning. In *CVPR*, pages 2163–2173, 2019. 2, 6

[47] Edgar Schönfeld, S. Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero- and few-shot learning via aligned variational autoencoders. In *CVPR*, pages 8239–8247, 2019. 1, 2, 3, 6

[48] Florian Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 4

[49] Yuming Shen, J. Qin, and L. Huang. Invertible zero-shot recognition flows. In *ECCV*, 2020. 1, 2, 3, 6

[50] Christian Szegedy, W. Liu, Y. Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, D. Erhan, V. Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. 2

[51] M. R. Vyas, Hemanth Venkateswara, and S. Panchanathan. Leveraging seen and unseen semantic relationships for generative zero-shot learning. In *ECCV*, 2020. 1, 2, 6, 7

[52] Q. Wang and Ke Chen. Zero-shot visual recognition via bidirectional latent embedding. *International Journal of Computer Vision*, 124:356–383, 2017. 1, 2

[53] P. Welinder, S. Branson, T. Mita, C. Wah, Florian Schroff, Serge J. Belongie, and P. Perona. Caltech-ucsd birds 200. *Technical Report CNS-TR-2010-001, Caltech,*, 2010. 2, 3, 5

[54] Y. Wen, Kaipeng Zhang, Z. Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. 4

[55] Jiamin Wu, Tianzhu Zhang, Z. Zha, Jiebo Luo, Yongdong Zhang, and Feng Wu. Self-supervised domain-aware generative network for generalized zero-shot learning. In *CVPR*, pages 12764–12773, 2020. 3

[56] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Q. Nguyen, M. Hein, and B. Schiele. Latent embeddings for zero-shot classification. In *CVPR*, pages 69–77, 2016. 2

[57] Yongqin Xian, T. Lorenz, B. Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, pages 5542–5551, 2018. 1, 2, 5, 6

[58] Yongqin Xian, B. Schiele, and Zeynep Akata. Zero-shot learning — the good, the bad and the ugly. *CVPR*, pages 3077–3086, 2017. 1, 4, 5

[59] Yongqin Xian, Saurabh Sharma, B. Schiele, and Zeynep Akata. F-vaegan-d2: A feature generating framework for any-shot learning. In *CVPR*, pages 10267–10276, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[60] Guo-Sen Xie, L. Liu, Xiaobo Jin, F. Zhu, Zheng Zhang, J. Qin, Yazhou Yao, and L. Shao. Attentive region embedding network for zero-shot learning. In *CVPR*, pages 9376–9385, 2019. 1, 6

[61] Guo-Sen Xie, L. Liu, Xiaobo Jin, F. Zhu, Zheng Zhang, Yazhou Yao, J. Qin, and L. Shao. Region graph embedding network for zero-shot learning. In *ECCV*, 2020. 1, 6

[62] H. Yu and B. Lee. Zero-shot learning via simultaneous generating and learning. In *NeurIPS*, 2019. 1

[63] Y. Yu, Zhong Ji, J. Han, and Z. Zhang. Episode-based prototype generating network for zero-shot learning. In *CVPR*, pages 14032–14041, 2020. 1, 2, 6

[64] F. Zhang and G. Shi. Co-representation network for generalized zero-shot learning. In *ICML*, 2019. 1, 3, 6

[65] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *CVPR*, pages 6034–6042, 2016. 2

FREE: Feature Refinement for Generalized Zero-Shot Learning

Shiming Chen[1], Wenjie Wang[1], Beihao Xia[1], Qinmu Peng[1], Feng Zheng[2], Weiping Ding[3] and Xinge You[1,*]
[1]Huazhong University of Science and Technology (HUST), China
[2]Southern University of Science and Technology (SUSTech), China
[3]Nantong University (NTU), China
{shimingchen,wangwj54,xbh_hust,pengqinmu,youxg}@hust.edu.cn zfeng02@gmail.com ding.wp@ntu.edu.cn

Appendix

In this appendix, Section A summarizes the architecture details of FREE. Section B provides the key hyper-parameters setting of various datasets in our experiments. Section C demonstrates the hyper-parameters analysis of loss weights λ_{SAMC} and λ_{R_a}, and margin Δ in the SAMC-Loss. Section D presents t-SNE visualization of visual features of FLO and AWA1.

A. Network Topology

Table 5. Network topology of FREE. Att is the dimensions of semantic vectors per class, i.e., Att=312 in CUB benchmark.

Encoder (E)	**Network type**: MLP **Input**: $x \oplus a$, size=2048+Att; **hidden layer**: Fully connected, neurons=4096; LeakyReLU; **Output**: Fully connected, neurons=Att; LeakyReLU;
Generator (G)	**Network type**: MLP **Input**: $z \oplus a$, size=Att+Att; **hidden layer**: Fully connected, neurons=4096; LeakyReLU; **Output**: Fully connected, neurons=2048; Sigmoid;
Discriminator (D)	**Network type**: MLP **Input**: $x \oplus a$ or $\hat{x} \oplus a$, size=2048+Att; **hidden layer**: Fully connected, neurons=4096; LeakyReLU; **Output**: Fully connected, neurons=1;
Feature Refinement (FR)	**Network type**: MLP **Input**: x or $\hat{x}$, size=2048; **hidden layer**: Fully connected, neurons=4096; LeakyReLU; **Output**: Fully connected, neurons=Att; Sigmoid;

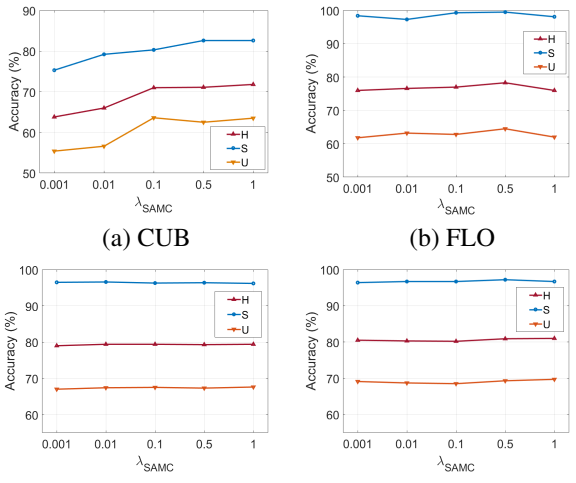
Our proposed FREE consists of the encoder (E), Generator/decoder (G), Discriminator (D), and Feature Refinement (FR). As described in the paper that E, G, D, and FR are MLP architectures, we present the architecture details of them as shown in Table 5.

B. Key Hyper-parameters Setting for Various Datasets

As shown in Table 6, we provide the key hyper-parameters setting for various datasets in our experiments.

Table 6. Key hyper-parameters setting for various datasets.

Dataset	$\lvert\hat{a}\rvert$	λ_{SAMC}	λ_{R_a}	γ	N_{syn}	margin(Δ)
CUB	312	0.5	0.1	0.8	400	200
FLO	1024	0.5	0.1	0.8	1200	200
AWA1	85	0.5	0.1	0.1	1800	50
AWA2	85	0.5	0.1	0.1	1800	50



(a) CUB (b) FLO

(c) AWA1 (d) AWA2

Figure 8. The effectiveness of loss weight λ_{SAMC} for SAMC-Loss.

C. Hyper-parameter Analysis

Loss Weights λ_{SAMC} and λ_{R_a}. Here we show how to set the FREE related loss weights: one is λ_{SAMC} that controls the importance of SAMC-Loss; the other one is λ_{R_a} that controls the semantic cycle-consistency loss. We take various values for λ_{SAMC} and λ_{R_a} to search better setting of them, i.e., $\lambda_{SAMC} = \{0.001, 0.01, 0.1, 0.5, 1\}$ and $\lambda_{R\_a} = \{0.001, 0.01, 0.1, 0.5, 1\}$. As shown in Fig. 8 and Fig. 10, FREE is overally steady for the setting of λ_{SAMC} and λ_{R_a} on coarse-grained benchmarks (i.e., AWA1, AWA2), while it is relatively sensitive to the setting of λ_{SAMC} and λ_{R_a} on fine-grained benchmarks (i.e., CUB,

*Corresponding author

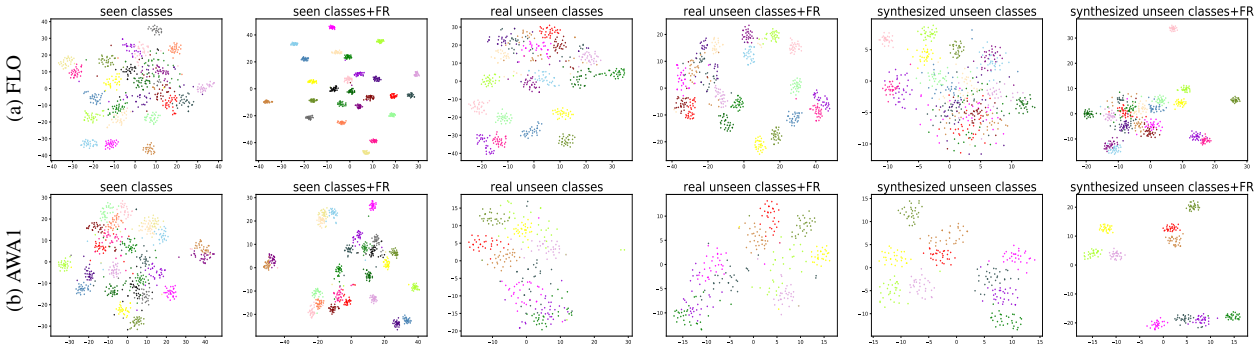
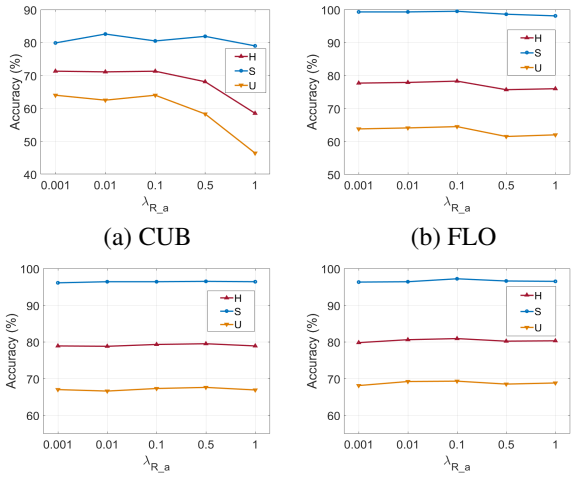


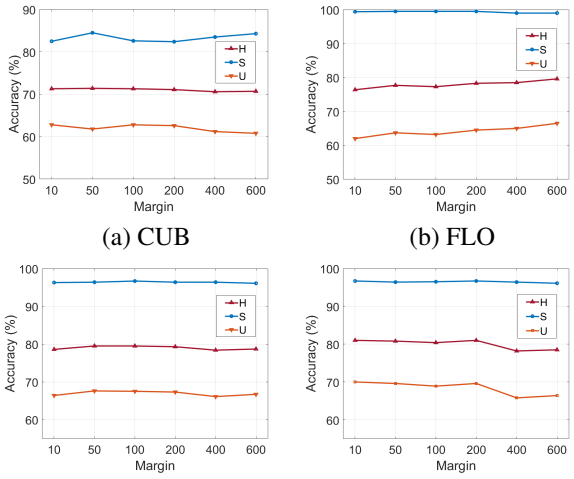
Figure 9. t-SNE visualization of visual feature for real seen classes, real/synthesized unseen classes, and their refined features by FR on (a) FLO and (b) AWA1. Different colors denote different classes.



(a) CUB (b) FLO

(c) AWA1 (d) AWA2

Figure 10. The effectiveness of loss weight λ_{R_a} for semantic cycle-consistency loss.



(a) CUB (b) FLO

(c) AWA1 (d) AWA2

Figure 11. The impact of the margin (Δ) in the SAMC-Loss.

FLO). This is because that the fine-grained benchmarks have a larger cross-dataset bias with ImageNet than coarse-grained benchmarks, and thus the GZSL classification performance of fine-grained benchmarks is more closely decided by FREE that is optimized by the SAMC-Loss and semantic cycle-consistency loss. We finally set $\lambda_{SAMC} = 0.5$ and $\lambda_{R\_a} = 0.1$ in all experiments.

Margin Δ in SMAC-Loss. We take a margin Δ to control the distance between intra-class and inter-class pairs in SAMC-Loss. As shown in Fig. 11, the larger margin is good for compacting the distance of intra-class in fine-grained datasets (e.g., CUB, FLO), while a relatively smaller margin helps inter-class separability of coarse-grained datasets (e.g., AWA1, AWA2). To this end, we set $\Delta = 200$ for CUB and FLO, and $\Delta = 50$ for AWA1 and AWA2 to conduct all experiments.

D. t-SNE Visualization for FLO and AWA1

We display the t-SNE visualization of visual features for FLO and AWA1. As shown in Fig. 9, FREE effectively refines the visual features of seen classes and real/synthesized unseen classes for FLO and AWA1, which intuitively shows that why FREE can significantly tackle the problem of cross-dataset bias. Notably, the visual features of seen classes in FLO are well progressed such that FREE achieves top-1 accuracy with 99.5%.