Generative Mixup Networks for Zero-Shot Learning

Bingrong Xu, Zhigang Zeng, *Fellow, IEEE*, Cheng Lian, *Member, IEEE*, and Zhengming Ding, *Member, IEEE*

***Abstract*— Zero-shot learning casts light on lacking unseen class data by transferring knowledge from seen classes via a joint semantic space. However, the distributions of samples from seen and unseen classes are usually imbalanced. Many zero-shot learning methods fail to obtain satisfactory results in the generalized zero-shot learning task, where seen and unseen classes are all used for the test. Also, irregular structures of some classes may result in inappropriate mapping from visual features space to semantic attribute space. A novel generative mixup networks with semantic graph alignment is proposed in this article to mitigate such problems. To be specific, our model first attempts to synthesize samples conditioned with class-level semantic information as the prototype to recover the class-based feature distribution from the given semantic description. Second, the proposed model explores a mixup mechanism to augment training samples and improve the generalization ability of the model. Third, triplet gradient matching loss is developed to guarantee the class invariance to be more continuous in the latent space, and it can help the discriminator distinguish the real and fake samples. Finally, a similarity graph is constructed from semantic attributes to capture the intrinsic correlations and guides the feature generation process. Extensive experiments conducted on several zero-shot learning benchmarks from different tasks prove that the proposed model can achieve superior performance over the state-of-the-art generalized zero-shot learning.**

***Index Terms*— Generative adversarial networks (GANs), mixup regularization, semantic graph alignment, zero-shot learning.**

I. Introduction

DEEP learning for computer vision has made incredible progress over the past decade. Deep learning-based models have been wildly used in many real applications, such as face recognition [1], remote sense [2], and object detection [3]. Most of the models heavily rely on the huge amount of annotated training samples to optimize the model. Since many benchmarks in practical applications have the long-tail distribution effect [4]–[6], the lack of some specific classes makes supervised learning impracticable. In such cases, many methods may fail to obtain ideal performance when the training data are insufficient. To overcome this challenge, different transfer learning paradigms are proposed, where the number of training samples with full labels is limited [7]–[9].

Transfer learning has been wildly used and achieved state-of-the-art performance in several practical applications [10]. Zero-shot learning can be considered a special case of it [11], [12], where the source (seen) and target (unseen) domains have different feature and label spaces, but share the same semantic space. It aims to transfer knowledge learned from seen categories to classify unseen categories. In other words, the training and testing classes are disjoint which are more challenge to deal with. Zero-shot learning tasks can be addressed by involving mid-level semantic information and usually have two-stage classification process. First, they map the visual features into the mid-layer semantic space, and then assign the semantic attributes to the corresponding labels, modeling the feature-attribute relationship based on the paired information of labeled data and attributes. For example, the bird class has semantic descriptions like "has wing," "has beak," etc. The monkey class has attributes like "has arm," "has tail," etc. The key to relating seen classes with unseen classes is the common semantic representation they shared, which usually is defined by human experts or extracted from texts. Most zero-shot learning methods are based on attributes [13]–[15]. They usually project the visual features from seen classes to the corresponding semantic attributes in the first step, then assign them to the predicted labels. In the inference stage, samples from unseen classes will be classified with the learned projection model.

Moreover, there is a more realistic scenario named generalized zero-shot learning [11]. The test samples are from both seen and unseen classes, which make the learning task more difficult. The generalized zero-shot learning methods need to tackle domain shift problems [16] and distribution difference of seen and unseen classes. Therefore, data augmentation has become a hot research topic in zero-shot learning problems recently, and some generative adversarial network (GAN)-based methods have been proposed [17]–[19]. The generated models utilize random noise and specific information of unseen classes to enhance the training sample, which aims to convert the zero-shot learning problem to a traditional

Manuscript received October 25, 2020; revised September 8, 2021; accepted January 7, 2022. This work was supported in part by the National Key R&D program of China under Grant 2018YFB1305500, the Technology Innovation Project of Hubei Province of China under Grant 2019AEA171 and the 111 Project on Computational Intelligence and Intelligent Control under Grant B18024; and in part by Bingrong Xu's Visiting in Dr. Ding's Group. *(Corresponding author: Zhigang Zeng.)*

Bingrong Xu and Zhigang Zeng are with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China, and also with the Key Laboratory of Image Processing and Intelligent Control of Education Ministry of China, Wuhan 430074, China (e-mail: bingrongxu@hust.edu.cn; zgzeng@hust.edu.cn).

Cheng Lian is with the School of Automation, Wuhan University of Technology, Wuhan 430070, China (e-mail: chenglian@whut.edu.cn).

Zhengming Ding is with the Department of Computer Science, Tulane University, New Orleans, LA 70118 USA (e-mail: zding1@tulane.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3142181.

Digital Object Identifier 10.1109/TNNLS.2022.3142181

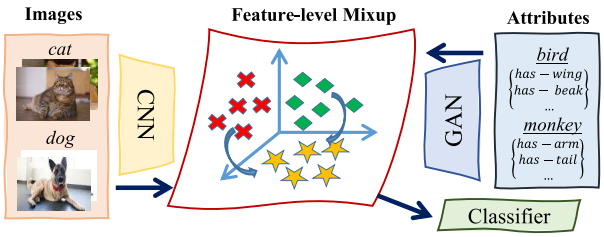

Fig. 1. Proposed generative mixup networks, where: 1) a novel conditional generator is proposed to synthesize visual features with given attributes; 2) mixup mechanism is deployed to diversify the feature space (the yellow $\star$ features are obtained by mixup the red $\times$ features and the green $\diamond$ features); and 3) all the features are input to train the classifier.

supervised classification problem. An inevitable problem with such paradigms is that they focus on generating discriminant features, ignoring manifold structure relations of the real data and generating data. There might be a domain mismatch across the two data domains. Moreover, samples from different classes are insufficient to ensure the distributions' invariance at most of the latent space. Also, the discriminator usually distinguishes the real and fake samples with the guidance of hard labels. It is more reasonable to use soft labels that combine the similarity information within different classes.

In this article, we propose a Generative Mixup Networks with semantic graph alignment for zero-shot learning (Fig. 1). Specifically, a novel generator is explored to augment samples from the given attributes as the prototype. The mixup mechanism on the feature level is utilized to improve the generalization ability of the model. It incorporates the prior knowledge to extend the training samples' distribution, for the linear interpolations of features can lead to linear interpolations of the associated labels, which can improve classifiers' ability for zero-shot learning and be regarded as a data augment method. To improve the relevance of synthesized samples to the corresponding semantic attributes and help the generator to synthesize better training images, the graph constraint and triplet gradient loss are designed to guide the GAN training process. The relationship graph can be regarded as a similarity matrix constructed based on each class's semantic attributes. It can capture the latent structure and guide the generation process of synthetic data. The triple gradient loss measures the discrepancy between the gradient vectors of three data distributions. It aims to guide the generator to minimize the classification loss and makes the mixup data closer to the real data. In summary, the contributions are highlighted as follows.

1) First of all, we propose a novel conditional generative model for zero-shot learning. A mixup mechanism on the feature level is proposed to augment the training set and explore the intrinsic structure of distributions of real and generated samples through triplet gradient loss. This method promotes generators' generalization ability and makes the discriminator judge differences relative to real and fake data with refined scores.
2) Second, the similarity graph is constructed based on the provided semantic attributes of all classes. The semantic graph alignment preserves the intrinsic structure of the semantic information to guide the training process. It constrains the relationship between real and synthetic data with the semantic structure to transfer the knowledge from seen classes to unseen classes.
3) Finally, the proposed model is evaluated on several well-known zero-shot learning benchmarks for different visual tasks, such as classification, retrieval, and annotation. The experimental results demonstrate that the proposed model is superior to the existing methods. Also, an ablation study is provided for the analysis of the model.

The following sections of this article are organized as: Section II reviews the related work on zero-shot learning. Section III introduces the proposed model in detail. Section IV presents the experimental evaluation and parameter analyses. Finally, this article is concluded in Section V.

II. Related Work

A. Zero-Shot Learning

Zero-shot learning generally aims to project the visual features and semantic information in linear or deep learning ways [14], [20], [21]. In [22], the zero-order learning problem is first studied, and the direct attribute prediction (DAP) method is proposed, which uses attribute variables as an intermediate layer to decouple labels and features. A projection function is learned in [15], which aims to embed the visual features of the samples with the semantic space. Some approaches attempt to map labels and features into the same space for training and testing [23]. In [24], a regularized sparse coding framework is constructed to project class labels and given features into the semantic space, which effectively overcomes the problem of shifting the projection domain. A projection framework with manifold regularization is proposed in [25], which is based on matrix tri-factorization and can effectively capture the geometric manifold structure of feature space and semantic space. Similarly, some methods try to embed feature space and semantic space into the intermediate space of classification [26], [27]. A novel classifier synthesis mechanism is presented in [28]. The semantic space is aligned with the visual feature space, and the classifiers for seen and unseen classes are constructed based on the common set of phantom classes. A latent embedding model is studied in [29], which aims to learn a compatibility function between image and class embeddings. The model is trained with rank-based constraints, which verifies the effectiveness of the method. In addition, some articles have constructed a data-driven metric to calculate the similarity of features and attributes of embeddings [30], [31].

As for generalized zero-shot learning, it is a more realistic protocol compared with zero-shot learning [11], [32]. The training set is the same, but the test set combines the samples from both seen and unseen classes. The classification ability of the generalized zero-shot learning model is measured by the classification accuracies of seen and unseen categories, respectively. And another evaluation score is introduced as harmonic mean, which is the measurement of classification

results. Several works have been done specifically for such setting [33], [34].

B. Generative Zero-Shot Learning

Recently, deep learning-based methods have good performances dealing with zero-shot learning [35], [36]. GAN is also widely utilized as a data augmentation mechanism in such a problem [18], [19], [37]. It typically consists of a generator and a discriminator that is trained in an adversarial manner [38]. The generator tries to defraud the discriminator with a synthetic distribution of random noise close to the real data distribution, while the discriminator tries to distinguish between the real distribution and the synthetic distribution. GAN has been proven efficient in generating data in computer vision and natural language processing tasks [39], [40].

In zero-shot learning, semantic attributes are the primary source of auxiliary information, combined with random noise as input for GAN. Several generative models are evaluated for the ability to generate training examples in [41]. The noisy semantic descriptions are taken as input to generate unseen visual features in [42]. Hence the classifier can recognize the novel classes with no samples being seen. GAN is constrained with a multi-modal cycle consistency loss term in [43], which is a regularization of the generation of synthetic features. It can force the GAN training process to generate more constrained visual representations. Separate re-constructor, discriminator, and classifier are used to target at features to overcome the domain-shift problem in [44]. Wasserstein GAN (WGAN) [45] is explored for the zero-shot learning problem in [18]. In addition to the unsupervised adversarial loss, a discriminative supervised loss function is proposed. WGAN is combined with a gradient matching loss in [19], the value of the synthesized examples is maximized by measuring the gradient signal's quality which can improve the classification results significantly. WGAN is also utilized in [37], and a boundary loss is introduced to minimizing feature confusion.

Most existing methods [20], [46], [47] utilize the feature and semantic information from the seen classes to learn a projection function to predict the samples from unseen classes. It means more training samples will lead to better generalization ability for unseen test samples. Thus, GAN-based methods [18], [19], [48], [49] are proposed to synthesize more features to feed the classifier. Although they performed well compared with state-of-the-art methods, some issues still exist. The classification accuracies of the GAN-based methods heavily depend on the quality of synthesized samples. If the generated unseen features are far from the real ones who share the same semantic attributes, the classification accuracy would dramatically drop. Moreover, the decision boundary of normal GAN only contained 0 and 1, which might cause sensitivity to adversarial examples.

III. Proposed Method

A. Preliminaries

Denote the training set of seen classes as $\mathcal{S}_{\text{set}} = \{x, y, c(y)\}$, where $x \in \mathcal{X}^{\mathcal{S}}$ is the feature vector of a training sample, y is the corresponding label in the label set $\mathcal{Y}^S = \{y_1, y_2, \ldots, y_S\}$ of S seen classes. $c(y) \in \mathcal{C}$ is the semantic embedding of a certain class y. Moreover, we denote the training set from unseen classes as $\mathcal{U}_{\text{set}} = \{c(y_u)\}$, where $c(y_u) \in \mathcal{C}$ is the semantic information of the unseen classes $y_u \in \mathcal{Y}^U$. The total number of classes is the sum of seen and unseen classes, $l = S + U$. In the training process, the sets $\mathcal{S}_{\text{set}}$ and $\mathcal{U}_{\text{set}}$ are provided. $\mathcal{S}_{\text{set}}$ is used to train the generative model, $\mathcal{S}_{\text{set}}$ and $\mathcal{U}_{\text{set}}$ are all utilized to train the separate classifier. To be more specific, the visual features of the unseen classes are not available, but the corresponding semantic information is accessible to use. The task of zero-shot learning is to learn the classifier based on the training features for unseen classes $\mathcal{X}^{\mathcal{U}}$ to $\mathcal{Y}^U$, and the classifier for generalized zero-shot learning task is to classify both seen and unseen classes' samples to $\mathcal{Y}^S \cup \mathcal{Y}^U$.

B. Generative Mixup Networks

The proposed Generative Mixup Networks aims to make the distributions of the real and generated data closer by involving a feature fusion mechanism, and it is trained with smoother decision boundaries. The main insight of the proposed model is to use attribute information as the prototype and generate more training samples through GAN. Conditional GAN is the basic spirit of our model. The mixup mechanism on the feature level is adopted as a data augmentation method, which can lead decision boundaries to transition linearly and provide a smoother estimate of different classes. The semantic graph alignment loss and the triple gradient matching loss are come up with to guide the training process of the CGAN. Semantic graph alignment constraints utilize the similarity matrix constructed from the semantic attributes of the class. Its purpose is to capture the latent structures and guide the generation of synthetic data. The triplet gradient loss measures the discrepancy between the gradient vectors of three data distributions and works as a proxy for the classification loss, which attempts to guide the generator to minimize the classification loss and make the mixup data closer to the real data. The pipeline of our model is illustrated in Fig. 2, which contains three modules as below.

1) Attribute-Visual Generative Mixup Networks: GAN [38] consists of a generative network G and a discriminative network D. Generator G attempts to generate images close to the real images to fool the discriminator, while the discriminator D tries to distinguish the real and the generated images. Conventional GAN can be extended to a conditional model when the generator and discriminator are conditioned on some auxiliary information [50]. We perform the conditioning by involving semantic attributes of a specific class with prior noise as input in joint representation, and the visual features of the class are outputs

$$\mathcal{L}_{\text{CGAN}} = \mathbb{E}[D(X^r, c(Y^r))] - \mathbb{E}[D(X^g, c(Y^g))] \tag{1}$$

where $X^r \in \mathcal{X}^{\mathcal{S}}$ is the mini-batch of real features from the seen training set, $X^g = G(Z, c(Y^g))$ is the batch of generated features, Z is the random Gaussian noise. Compared with the traditional GAN, the log term is removed.

Mixup is a new data fusion mechanism [51]. It combines the pairs of examples and the corresponding labels. It can create

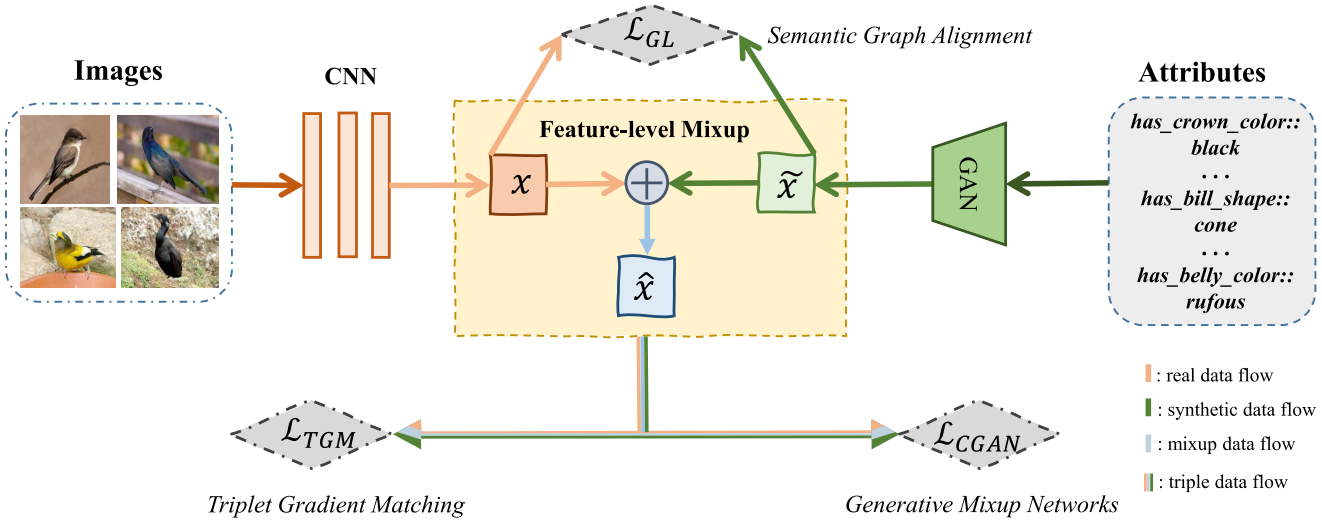

Fig. 2. Pipeline in the training phase. The real features x are extracted from the raw images, and the synthetic features $\tilde{x}$ are generated from the corresponding class-level attributes. The mixup features $\hat{x}$ are obtained by mixing the real and synthetic features. Then all the features are used for conditional GAN optimization.

smoother decision boundaries that have been proved to have a positive effect on generalization performance. The vicinal mixup distribution encourages the model to behave linearly in-between training examples. To explore the latent structure of the real data source and generated data source, mixup data are linearly interpolated as follows:

$$\begin{aligned} X^m &= \lambda X^r + (1-\lambda)X^g \\ Y^m &= \lambda Y^r + (1-\lambda)Y^g \end{aligned} \tag{2}$$

where $\lambda \sim \text{Beta}(\alpha, \alpha)$ for $\alpha \in (0, \infty)$. X^r and X^g are all in batches, Y^r and Y^g are the corresponding label matrix.

Then we combine the mixup data with the conditional GAN optimization process. Inspired by the f-CLSWGAN in [18], gradient penalty is applied on discriminators to constraint the Lipschitz norm of mixup data and provide generators with more stable gradients. The generative model is formulated as

$$\begin{aligned} \mathcal{L}_{\text{CGAN}} = \mathbb{E}[D(X^r, c(Y^r))] - E[D(X^g, c(Y^g))] \\ - \mu\mathbb{E}[(\nabla_{X^m}\|D(X^m, c(Y^m))\|_2 - 1)^2] \end{aligned} \tag{3}$$

where X^m is the mixup data batch, and μ is the penalty parameter. The first two terms approximate the Wasserstein distance, and the gradient penalty has the unit norm of the mixup data. The generator takes the combination of semantic attributes per class and random Gaussian noise as input and produces the samples associated with certain classes as output. The mixup data would lead the discriminator to behave linearly between real and fake datasets, which may have a higher capacity to judge the generated images containing oscillations.

Remarks: Since the real and the generated data share different distributions, the proposed model aims to map real and generated data into a joint latent distribution with the mixup feature embedding in the training process. The latent distributions of real and synthetic samples from the same classes could be domain-invariant, forming the inter-class clusters. The gradient term of X^m is similar to the WGAN. There is a Lipschitz constraint on the optimal discriminator. If f^* is the optimal solution of $\max_{\|f\|_L \le 1} \mathbb{E}_{x\sim P_r} f(x) - \mathbb{E}_{y\sim P_g} f(y)$, where P_r and P_g are the data distributions of real and generated samples in a compact metric space, respectively. Note $\mathcal{C}$ as the optimal coupling between P_r and P_g, the minimize problem can be defined as: $\inf_{\mathcal{C}\in\Omega(P_r,P_g)} \mathbb{E}_{(x,y)\sim\mathcal{C}}\|x-y\|$, where $\Omega(P_r, P_g)$ is the joint distribution with marginals P_r and P_g. If f^* is differentiable, $\mathcal{C}(x = y) = 0$, and $x^m = \lambda x^r + (1-\lambda)x^g$, it holds that $P_{(x,y)\sim\mathcal{C}}[\nabla f^*(x^m) = 1]$. f^* has gradient norm 1 almost everywhere in $\mathcal{C}$. The proof of it can be referred to in [45]. Under such constraints, the gradient concerning its input is better behaved than its GAN counterpart, optimizing the generator.

2) Semantic Graph Alignment: To further improve the quality of the synthesized features, we hope to transfer the data's geometric structure without violating the intrinsic distributions. Thus, the semantic space is used as the bridge to project the real data distribution to synthetic data distribution. If the real samples and synthetic samples share the same semantic descriptions, they are supposed to be close in the intrinsic geometry of the data distribution. The geometric structure can be modeled by the nearest neighbor graph in the semantic space.

Hence we introduce an auxiliary semantic-embedding graph to guide the data generation as follows:

$$\mathcal{L}_{\text{GL}} = \|X^r - G_a X^g\|_F^2 \tag{4}$$

where $\|\cdot\|_F$ is the Frobenius norm. X^r and X^g are the mini-batch of real data and generated data, respectively. G_a is the attribute-based similarity matrix, which presents the intrinsic relationship of each class semantic presentation. It is the binary symmetric weight matrix of the k-nearest-neighbor graph constructed from semantic attributes $c(Y)$, where Y is the corresponding label matrix of the data batch. The graph G_a only contains 1 s or 0 s. The proposed graph constraint loss preserves the geometric structure of the common semantic

space and aims to lead generated data distribution close to the real distribution.

3) Triplet Gradient Matching: To ensure that the generator can produce examples from which accurate classifiers can be trained, F-CLSWGAN [18] is proposed to minimize the classification loss of the generated data using a negative log-likelihood. However, the classification loss guides the optimization of the classifier rather than the GAN model. Gradient matching network (GMN) [19] observed that the partial derivatives of the classification loss over synthetic dataset are highly correlated with that of the real training dataset. Thus, they proposed a loss function to minimize the gradients' approximation error obtained over the synthetic data, which is the gradient matching loss.

We combine the gradient matching loss with the mixup mechanism and propose a triplet gradient matching loss. The gradient loss of three data sources are defined as

$$\begin{aligned} g_r(\theta) &= \mathbb{E}\big[\nabla_\theta \mathcal{L}_C(X^r, c(Y^r); \theta)\big] \\ g_m(\theta) &= \mathbb{E}\big[\nabla_\theta \mathcal{L}_C(X^m, c(Y^m); \theta)\big] \\ g_g(\theta) &= \mathbb{E}\big[\nabla_\theta \mathcal{L}_C(X^g, c(Y^g); \theta)\big] \end{aligned} \tag{5}$$

where θ is the classifier parameter, $\mathcal{L}_C$ is the loss function to train the compatibility function $f(X, c(Y); \theta)$. All the data are fed to network in batches.

In the classification process, both visual features and semantic attributes are utilized to obtain better classification results. The bilinear model, which implements the compatibility function, is defined as

$$f(X, c(Y); \theta) = X^\top W c(Y) + B \tag{6}$$

where compatibility matrix W and the bias matrix B respond to the θ. $\mathcal{L}_C$ is adopted with cross-entropy loss.

Then we measure the discrepancy across real samples, synthetic samples, and mixup samples in the triplet way. The final loss function based on cosine distance is formulated as

$$\mathcal{L}_{\text{TGM}} = \mathbb{E}_\theta\left[1 - \frac{g_r(\theta)^\top g_m(\theta)}{\|g_r(\theta)\|_2 \|g_m(\theta)\|_2} + \frac{g_g(\theta)^\top g_m(\theta)}{\|g_g(\theta)\|_2 \|g_m(\theta)\|_2}\right]. \tag{7}$$

The triplet gradient matching loss guarantees that the classifier will be trained based on the gradient loss. It can be regarded as a proxy for classification loss. Also, it constrains the distance between three data distributions, making the mixup data closer to the real data, then leading to the optimization of the generative model.

C. Overall Objective and Optimization

To sum up, the overall objective function then becomes

$$\mathcal{L} = \mathcal{L}_{\text{CGAN}} + \gamma\, \mathcal{L}_{\text{GL}} + \eta \mathcal{L}_{\text{TGM}} \tag{8}$$

where γ and η are the regularization parameters to balance the contributions of different loss terms.

In the training process, the generator G, the discriminator D, and the classifier θ are optimized alternatively according to the total loss function (8). The detailed training process is presented in Algorithm 1. After the training process of generating the model, the samples are synthesized with the well-trained CGAN model by providing certain semantic attributes. The classifier is then retrained with real and generated features, and used to perform zero-shot learning and generalized zero-shot learning. The test samples will be classified with the highest compatibility score.

Algorithm 1 Training Process of Generative Mixup Networks

Input:
Features and the corresponding semantic attributes of seen classes; the number of iterations times T; the hyperparameters.

Initialization:
The learned parameters G for generator, D for discriminator, and θ for classifier.

while $t < T$ **do**

1. Feed the model with a minibatch of real features X^r, the corresponding label matrix Y^r, and the semantic attributes $c(Y^r)$. Gaussian noises $Z \sim \mathcal{N}(0, 1)$
2. Compute the generated features X^g, and the mixup features X^m, via (2).
3. Optimize G, D, θ alternatively as follows:
   $G := \text{argmin}_G\ \mathcal{L}$;
   $D := \text{argmin}_D\ \mathcal{L}_{CGAN}$;
   $\theta := \text{argmin}_\theta\ \mathcal{L}_{TGM}$.
4. $t := t + 1$.

end while

Output: G, D, θ.

IV. Experiments

In this section, we present the experimental results of our proposed model. First, we introduce three wildly used benchmarks and the evaluation protocol. Then, we report the comparing results of our proposed model with the state-of-art methods for zero-shot learning and generalized zero-shot learning classification tasks. Also, we test our model on retrieval and annotation tasks. Finally, an ablation study of our model under different conditions is provided.

A. Experimental Setup

1) Datasets: We evaluate our model on three standard zero-shot learning datasets, i.e., Animal with Attribute 1 and 2 (AWA1 [22] and AWA2 [11]), CUB-200-2011 Birds (CUB) [52], and SUN Attribute (SUN) [53]. AWA1 dataset is a coarse-grained dataset with 30 475 images from 50 classes. The class set of it is relatively small, which makes zero-shot learning classification on unseen classes more difficult. AWA2 dataset can act as a drop-in replacement to the AWA1 dataset, it consists of 37 322 images of 50 animals classes with 85 numeric attribute values for each class. SUN and CUB are fine-grained image datasets. SUN benchmark consists of 717 scene categories with 14 340 samples. CUB benchmark includes 11 788 images from 200 different species of birds. For these two datasets, each class contains a relatively small set of images for training making it challenging for zero-shot

TABLE I
STATISTICS OF THE ZERO-SHOT LEARNING BENCHMARKS

Dataset	AWA1	AWA2	SUN	CUB
Seen Classes	40	40	707	150
Unseen Classes	10	10	10	50
Samples	30,475	37,322	14,340	11,788
Attribute Dim	85	85	102	312
Attribute Type	continuous	continuous	continuous	continuous

learning tasks. The specific information of each dataset is shown in Table I.

2) Evaluation Metric: When the conditional GAN is well-trained, we synthesize data with the model for both seen and unseen classes to train the separate classifier. Zero-shot learning aims to classify unseen classes. Thus, in the zero-shot learning tasks, we compute normalized mean top-1 accuracies for unseen classes as evaluation scores. In the generalized zero-shot learning settings, all classes are for testing. Here, we compute the average accuracy of each class on seen classes (denote as s) and unseen classes (denote as u), the harmonic mean h is calculated as $h = 2 \times (s \times u)/(s + u)$.

3) Implementation: In our experiments, the data splits, images, and class embedding of zero-shot learning and generalized zero-shot learning followed the protocols proposed in [54]. The image features of 2048 dimension are extracted from 101-layered ResNet pre-trained on ImageNet-1K [55] and not fine-tuned. The attributes of each class are continuous and distributed between 0 and 1. None of the visual features from unseen classes are used during the training process. In the experiments, the generator G and discriminator D are constructed as MLPs with two hidden layers with 2048 units. The activation function is set as ReLU, and the classification loss is CrossEntropyLoss. The training samples' semantic attributes are fed to training conditional GAN in batches that are randomly chosen from different classes. Thus, the semantic information from seen classes is fully utilized to construct a better affinity graph matrix to guide the optimization process. The batch size is set as 256. The noise z is with the same dimension as class attributes drawn from a unit Gaussian distribution, fed into the generator with certain class attributes to synthesize corresponding samples. In the experiments, the mixup parameter α is set as 2, and μ of conditional GAN is set as 10. The graph constraint parameter γ is set as 0.01. The value of k for KNN is set as 10. The triplet gradient parameter η is set as 100.

B. Zero-Shot Classification

In the classification tasks, we test our proposed model on four benchmarks: AWA1, AWA2, SUN, and CUB under zero-shot learning and generalized zero-shot learning settings. Compared with some state-of-the-art methods: CMT [35], DeViSE [56], LatEm [29], SynC [28], SE [57], CADA-VAE [36], f-CLSWGAN [18], AFC-GAN [37], and GMN [19].

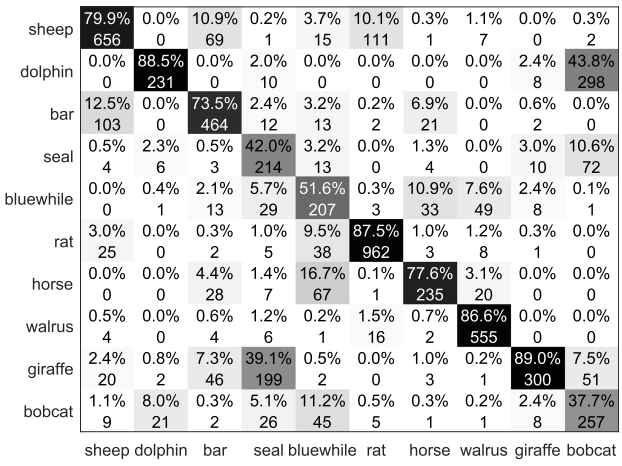

Fig. 3. Confusion matrix of test results on ten unseen classes from AWA1 dataset of the proposed model. Diagonal numbers indicate the correct number and prediction accuracy. Column means the ground truth, and row denotes the predictions.

The classification results are shown in Table II. From the results, it is obvious that deep learning-based methods outperform the traditional methods in all three datasets, especially for generalized zero-shot learning tasks. And the generative model-based methods significantly improve over the baselines where only real samples are used for training. We can observe that the mixup mechanism helps the generative baselines promote 1–3% in the standard zero-shot learning setting and improves zero-shot learning results. In AWA1, the u score of AFC-GAN is slightly higher than the proposed method, but the s score is 10% lower. Meanwhile, SynC has good performance with seen classes in generalized zero-shot learning, but the score of u is only 8.9%. There is a tradeoff between the classification results of u and s, their harmonic means h can demonstrate the two terms in a compromise way. In AWA1, AWA2, and SUN dataset, though the s score and u score of the proposed model in generalized zero-shot learning are not the best, the h score is the highest. In the CUB, the proposed method achieves the best performance in all evaluation terms.

Moreover, the visualization of the classification results obtained by the proposed model as the confusion matrix is demonstrated in Fig. 3. Rows and columns in the matrix correspond to the ground truth label and the predicted labels, respectively. Diagonal elements in the matrix indicate the correct prediction accuracy of each category. Some classes share very similar attributes and features, which may lead to misclassification. Still, the proposed model achieved satisfying results in some classes, such as dolphin, rat, and walrus. The classification results of these three classes are higher than 80%.

C. Zero-Shot Retrieval

In this subsection, we conduct retrieval experiments on AWA1, CUB, and SUN datasets. The zero-shot classification problem is to classify a given testing sample into the most relevant candidate class. In contrast, the zero-shot retrieval task is an inverse-process. It retrieves images related to the specified attribute descriptions of unseen classes. For the retrieval task,

TABLE II

COMPARISONS OF CLASSIFICATION RATES (%) OF THE PROPOSED MODEL AND THE STATE-OF-ART METHODS ON THREE BENCHMARKS UNDER ZERO-SHOT LEARNING AND GENERALIZED ZERO-SHOT LEARNING SETTINGS (%)

Dataset	AWA1				AWA2				SUN				CUB			
Method	ZSL	u	s	h	ZSL	u	s	h	ZSL	u	s	h	ZSL	u	s	h
CMT [35]	39.5	6.9	67.6	12.5	37.9	8.7	89.0	15.9	39.9	8.1	21.8	11.8	34.6	7.2	49.8	12.6
DeViSE [56]	54.2	13.5	68.7	22.4	59.7	17.1	74.7	27.8	56.5	27.5	16.9	20.9	52.0	53.1	23.8	32.8
LatEm [29]	55.2	7.3	71.7	13.3	55.8	11.5	77.3	20.0	55.3	14.7	28.9	19.5	49.4	15.3	57.3	24.0
SynC [28]	67.0	8.9	**87.3**	16.2	46.6	9.7	**89.7**	17.5	59.4	32.9	40.8	39.9	39.9	45.7	43.8	44.7
SE [57]	69.5	56.3	67.8	61.5	69.2	58.3	68.1	62.8	63.4	40.9	30.5	34.9	59.6	41.5	53.3	46.7
CADA-VAE [36]	69.5	57.3	72.8	64.1	-	55.8	75.0	63.9	57.7	35.7	**47.2**	40.7	55.7	53.5	51.7	52.6
f-CLSWGAN [18]	67.2	55.0	61.5	58.0	66.1	52.1	68.9	59.4	59.5	41.6	34.7	37.8	62.3	51.0	58.1	54.5
AFC-GAN [37]	69.1	**58.2**	66.8	62.2	-	-	-	-	63.3	**49.1**	36.1	41.6	62.9	53.5	59.7	56.4
GMN [19]	70.4	52.1	76.6	62.1	68.7	58.9	65.7	62.1	62.5	43.0	32.9	37.3	64.7	56.0	59.3	57.6
Ours	**71.3**	55.3	78.2	**64.8**	**69.5**	**60.3**	69.7	**64.7**	**67.7**	46.3	38.4	**42.0**	**66.3**	**58.8**	**60.7**	**59.7**

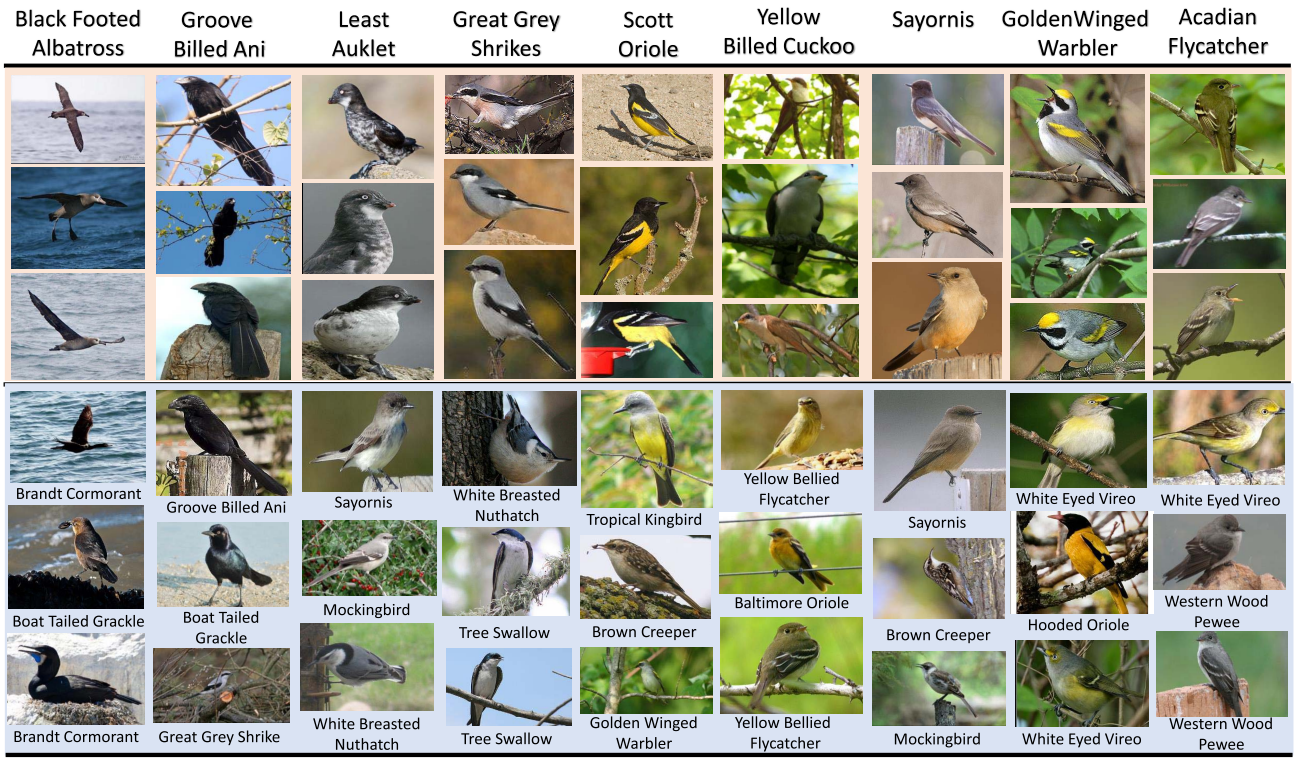

Fig. 4. Retrieval results of the proposed method on CUB dataset, where the labels of test classes are shown on the top, the top-3 correct classified samples are listed in the middle row (yellow area), the top-3 misclassified samples are listed in the bottom row (blue area) and the corresponding labels are annotated beneath the pictures.

only semantic information of unseen classes is provided. The test images are sorted with the similarity values by calculating the loss with the given attributes. In Table III, the mean average precisions (mAP) of three benchmarks are reported. It is obvious that the GAN-based methods significantly outperform the non-GAN based methods. Although the retrieval result of the proposed model on SUN dataset is lower than that of GANzrl, the proposed model beats it on AWA1 and CUB datasets.

Fig. 4 presents nine categories of the unseen test classes from the CUB benchmark, and we report the top-3 correct classified images (the top yellow area in the Fig. 4) and the top-3 misclassified images (the bottom blue area in the Fig. 4) of each class. From the results, we can notice that some misclassified samples share some very similar appearance with the correct classified samples, which makes it difficult to distinguish. From the results, it can be concluded that the proposed method can capture visual information discriminatively based on semantic attributes of each test category.

D. Zero-Shot Annotation

In this task, we explicitly infer the attributes corresponding to a specified novel class. Given a new sample from unseen classes, we want to annotate it by calculating the closest semantic attributes. The goal of zero-shot annotation is to

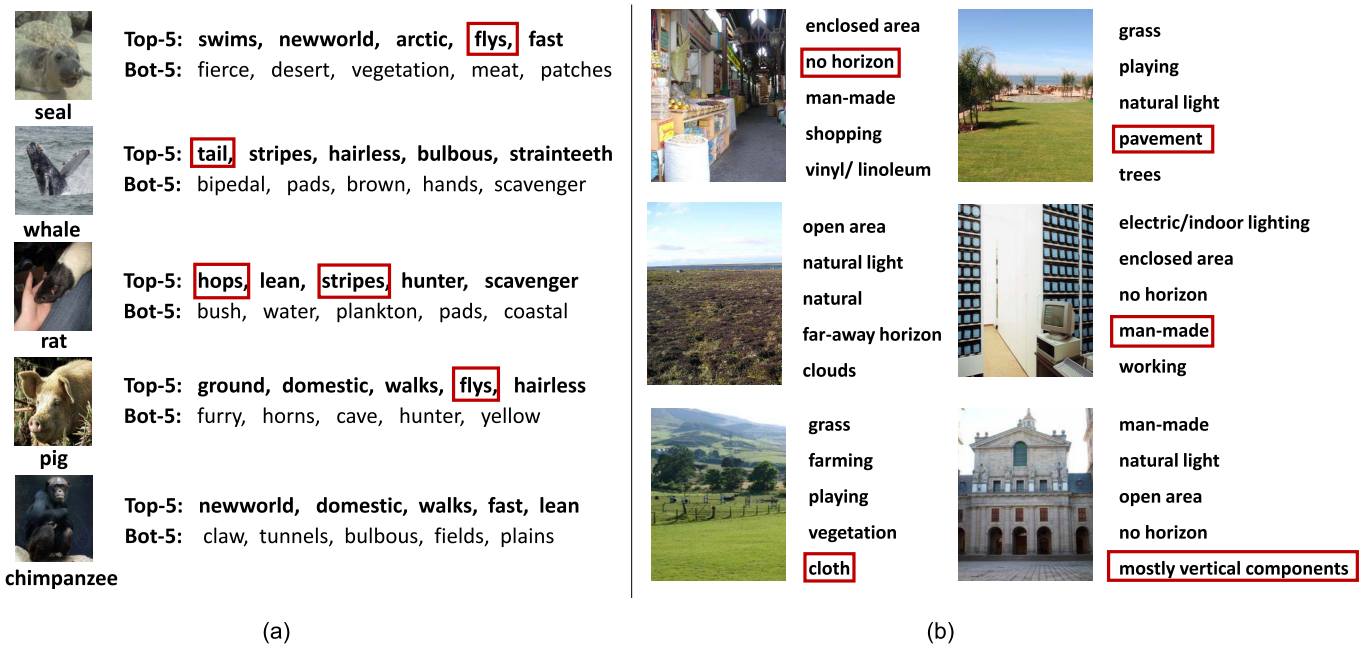

Fig. 5. Annotation results of proposed method on (a) AWA1 benchmark: Top and bottom five attributes predicted for five classes are listed. The false positives are highlighted in red bounding boxes. (b) SUN benchmark: Top five predicted attributes for six samples are listed. The annotated attributes in the red bounding boxes represent the wrong attributes.

TABLE III
RETRIEVAL PERFORMANCE COMPARISON (%) ON DIFFERENT BENCHMARKS (%)

Method	AWA1	CUB	SUN
SSE [26]	46.3	4.7	58.9
JSLE [27]	67.7	29.2	80.0
SynC [28]	65.4	23.9	76.5
MFMR [25]	70.8	30.6	77.4
GANzrl [17]	75.6	60.8	**90.1**
GMN [19]	74.9	62.6	80.3
Ours	**76.1**	**63.7**	82.7

learn a projection function $c(y) = f(x)$ for describing the test samples. However, this may suffer from the domain shift problem. The proposed model may alleviate this problem to some degree, for the synthetic samples generated according to the semantic attributes of unseen classes and the real data from seen classes are all used for training the projection model, making the model more discriminative.

The annotation experiment is conducted on the AwA1 dataset. Fig. 5(a) illustrates the description results of 5 unseen classes from AWA1. The top-5 and bottom-5 predicted attributes associated with a specific class are shown in the table. In ideal cases, all top 5 should be true positives, and all bottom 5 are true negatives. Obviously, the correct attributes are meaningful, and the wrong attributes have no direct relationship with certain classes. Fig. 5(b) shows the top five most related attributes corresponding to some unseen class samples on SUN benchmarks. The non-related attributes are highlighted with a red bounding box.

TABLE IV
ABLATION STUDY OF THE PROPOSED MODEL UNDER ZERO-SHOT LEARNING SETTINGS (%)

Methods	AWA1	SUN	CUB
GMN ($Baseline$)	70.4	62.5	64.7
Ours ($Mixup + Triplet$)	70.9	64.7	65.9
Ours ($Mixup + Triplet + Graph$)	71.3	68.2	66.3

E. Ablation Study

In this section, we first study the contributions of different terms and parameters' influences in the designed terms. Then, the effect of the feature generation will be presented. And the distributions of real data and synthetic data are visualized by the t-SEN.

1) Contribution of Different Terms: First of all, we evaluate the contribution of the mixup mechanism and the semantic graph alignment strategy to our model by removing one of them and keeping all other architectures under ZSL and GZSL settings, compared with baseline GMN [19]. The comparison results are shown in Tables IV and V, respectively. The proposed model without semantic graph alignment (but with mixup mechanism and triplet gradient loss) is noted as "Ours (Mixup+Triplet)" in the tables. The whole model (with mixup mechanism, semantic graph alignment, and triplet gradient loss) are noted as "Ours (Mixup + Triplte + Graph)." From the results, we observe that both the mixup mechanism and semantic graph alignment strategy benefit the classification tasks.

2) Contribution of Bilinear Scheme: The bilinear model (6) combines visual features and semantic attributes to classify

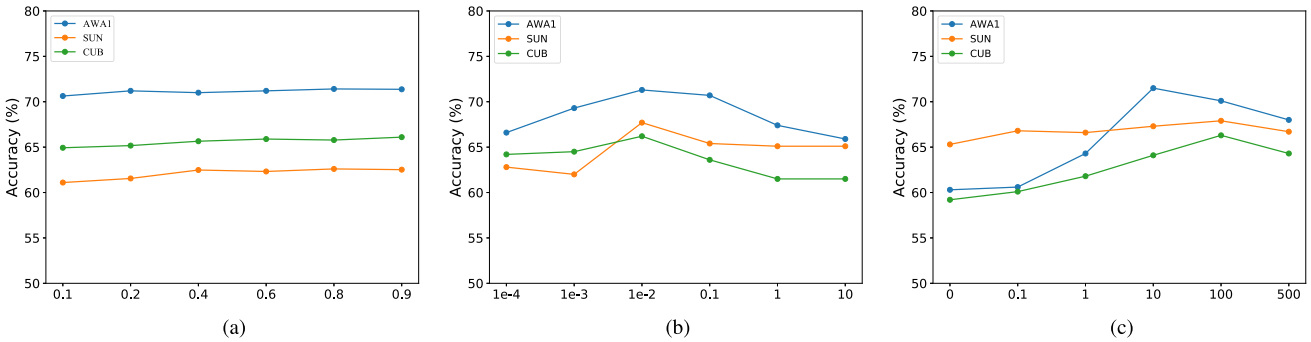

Fig. 6. Analysis of the impacts of mixupratio λ, semantic graph alignment constraint parameter γ, and triplet gradient matching constraint parameter η in different benchmarks, respectively. (a) Performance with various values of λ. (b) Performance with various values of γ. (c) Performance with various values of η.

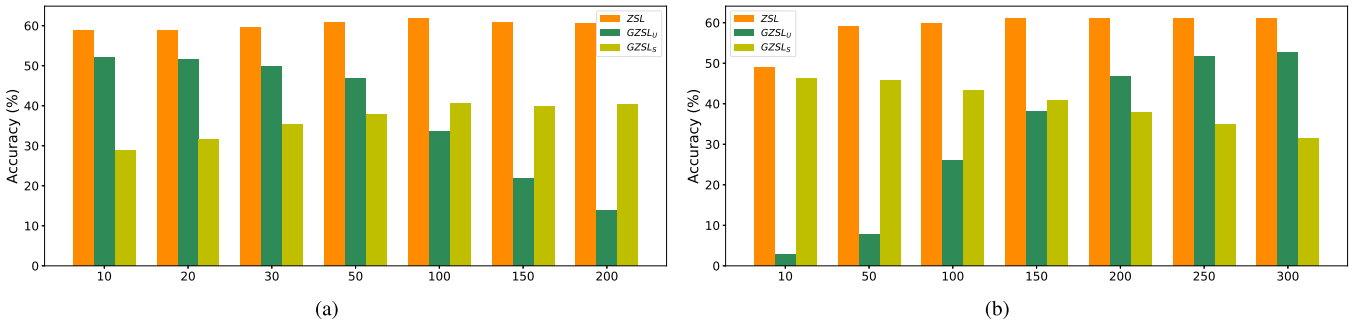

Fig. 7. Analysis of the impact of the number of synthesized features under ZSL and GZSL settings on SUN dataset. (a) Results under various number of synthetic data per seen class. (b) Results under various number of synthetic data per unseen class.

TABLE V
ABLATION STUDY OF THE PROPOSED MODEL UNDER GENERALIZED ZERO-SHOT LEARNING SETTINGS (%)

Dataset	Methods	u	s	h
AWA1	GMN ($Baseline$)	52.1	76.6	62.1
	Ours ($Mixup + Triplet$)	54.2	77.8	63.8
	Ours ($Mixup + Triplet + Graph$)	55.3	78.2	64.8
SUN	GMN ($Baseline$)	43.0	32.9	37.3
	Ours ($Mixup + Triplet$)	45.5	34.0	38.9
	Ours ($Mixup + Triplet + Graph$)	46.3	38.4	42.0
CUB	GMN ($Baseline$)	56.0	59.3	57.6
	Ours ($Mixup + Triplet$)	57.5	60.3	58.9
	Ours ($Mixup + Triplet + Graph$)	58.8	60.7	59.7

images. And we evaluate the contribution of the scheme by comparing it with a single model that only contains label information for classification. The experiments are performed on the AWA1, SUN, and CUB datasets, and the results are presented in Table VI, where the bilinear model is denoted as 'Bilinear-classifier' and the single model is denoted as "Single-classifier." The classification model with the bilinear scheme is better than the traditional classification model, which proves the effectiveness of the bilinear scheme.

3) Parameter Sensitivity: The mixup ratio λ is following the beta distribution of α, the value is distributed between 0 and 1. To show the mixup ratio's influence directly, we fix and tune the mixup ratio from the set $\{0.1, 0.2, 0.4, 0.6, 0.8, 0.9\}$.

TABLE VI
ABLATION STUDY OF THE PROPOSED MODEL UNDER ZERO-SHOT LEARNING SETTINGS (%)

Methods	AWA1	SUN	CUB
Bilinear-classifier	71.3	68.2	66.3
Single-classifier	67.0	64.7	64.1

Fig. 6(a) shows the performances on three datasets. The increase of the mixup ratio value will improve the classification results. However, the impact is trivial. reported in Table IV, it shows that the mixup mechanism can improve the model generating ability. The plots of experimental results with different graph constraint values γ on three datasets are shown in Fig. 6(b), which demonstrates that the proposed model is sensitive to the graph constraint with its parameter set. In AWA1 and CUB dataset, the accuracy rate first rises and then falls. The classification curve of SUN is fluctuating. When γ is smaller than a certain value, the classification accuracy changes dramatically. The influences of triplet gradient matching are shown in Fig. 6(c). For the loss is based on the cosine similarity, the value of parameter η is set between 0 and 500. In the AWA1 and CUB, the accuracy curve rises at first and then drops a little. In SUN, the curve is stabilizing. It is obvious that the triple gradient matching works well in the model optimization process, for in all three datasets, the accuracies with small gradient matching values are lower than the other results.

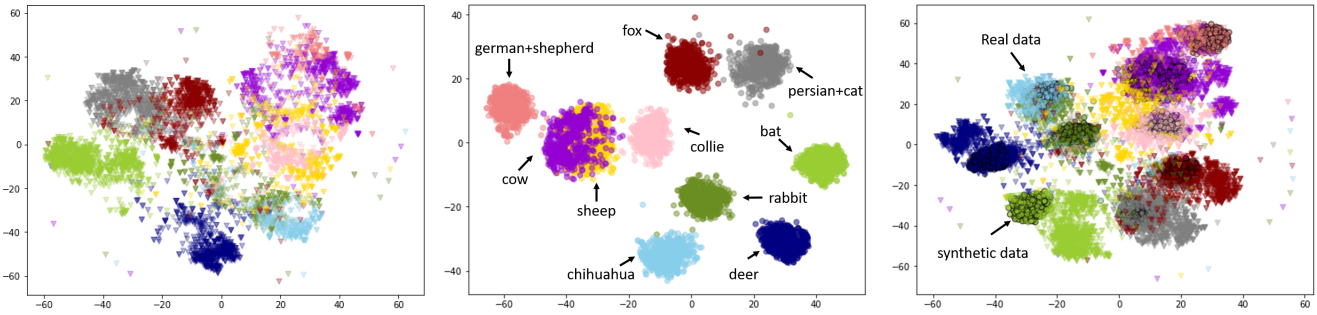

Fig. 8. t-SNE of the features from ten unseen classes on AWA1 dataset, where different colors denote different categories. (a) Real embedding with shape triangle △ and (b) synthetic embedding with shape circle ○. (c) Embedding of real (△) and synthesized visual features (○ with black bounding circles) together.

TABLE VII
IMPACT OF THE MINI-BATCH SIZE UNDER ZERO-SHOT LEARNING SETTINGS (%)

Dataset	32	64	128	256	512
AWA1	70.0	69.5	70.8	71.3	68.9
SUN	65.6	67.2	66.5	67.7	67.0
CUB	65.4	65.8	65.9	66.3	65.8

4) Influence of Batch Size: Batch size is an important hyper-parameter for training the proposed model with the semantic graph alignment term and the stochastic gradient descent (SGD) optimization process. The bigger the batch size is, the more information the Laplacian graph constraints. But the batch size also influences the efficiency of SGD. To demonstrate the impact of the batch size, we set the batch size as $\{32, 64, 128, 256, 512\}$ and record the ZSL classification results on AWA1, SUN, and CUB datasets, respectively. As shown in Table VII, the model gets the best classification results when batch size is set as 256.

5) Analysis of the Impact of the Number of Synthesized Features: To observe the influence of the number of synthesized features on the seen dataset, we fix the synthetic number of per unseen class as 200. As for the experiments on unseen classes, the generated number of per seen class is set as 50. From Fig. 7(a), we can see that with the number of synthesized features of seen classes increases, classification results of unseen classes drop rapidly, and the results of seen classes increase. The results of zero-shot learning are stable, for it is to classify the unseen classes. Thus, the change of sample number from seen classes has little influence on it. In Fig. 7(b), the curves of generalized zero-shot learning are just opposite to (a), and the zero-shot learning accuracy slightly improves. We can conclude that there is a tradeoff between the accuracy of seen and unseen classes. Thus the h measurement is meaningful to evaluate the generalized zero-shot learning models.

6) t-SNE Visualization: Finally, we visualize the 10 unseen categories in AwA1 dataset by exploring t-SNE. Here, we choose 5685 samples in the test set as the real data distribution for visualization as shown in Fig. 8(a). Five hundred samples per unseen class are generated by the proposed model, the distribution of 5000 synthesized features are shown in Fig. 8(b). Though t-SNE of different data are not aligned, the result of mixing the real features (inverted triangle symbol) and the synthetic data (spots with black bounding circles) together is shown in Fig. 8(c). We notice that the original test data has some overlapped parts, for some classes may share a similar semantic description. The visualization of the alignment data indicates that the synthesized data has a similar distribution as the real one, and most classes are discriminative. The visualization results demonstrate that the proposed model is valid in generating data with similar real data distribution.

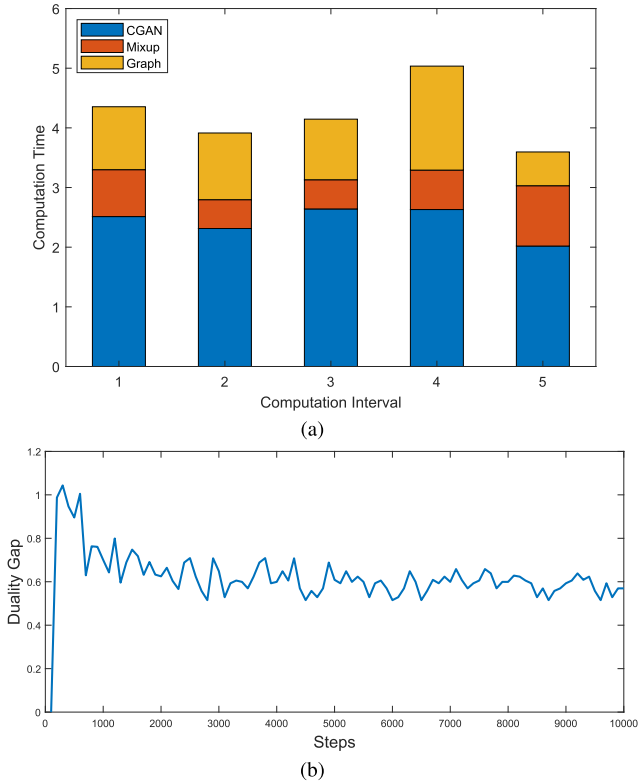

Fig. 9. (a) Computation time of each part and (b) quality gap value for monitoring and evaluating the proposed model.

F. Computational Complexity Analysis

GANs have achieved remarkable results in modeling data distributions, but their evaluation is still an open question.

Here, we follow the method proposed in [58] that leverages the notion of duality gap from game theory to present a measurement that can monitor the progress of a GANs model throughout training. To analysis the computational complexity of the proposed model, we conduct the experiment on the SUN dataset. The CGAN, the mixup mechanism, and the graph alignment terms are the main contributing factors that affect the computation time of the proposed model while training. Fig. 9(a) presents the computational time (over five trials). Each column denotes the cost time of 10 batches. The blue parts denote the computation time of the baseline CGAN, the red and the yellow parts denote the time of the mixup mechanism and semantic graph alignment strategy, respectively. Moreover, the duality gap allows for assessment of the similarity between the generated data and true data distributions, and detects stable mode collapse. The quality gaps of the proposed model are recorded per 50 epochs, and the curve is shown in Fig. 9(b). The curve fluctuates within a certain range and tends to be stable. The value of the duality gap is not zero, but the model reaches a dynamic equilibrium status.

V. Conclusion

In this work, we propose a Generative Mixup Networks combined with semantic graph alignment and triplet gradient matching loss to transfer the distribution from real data to synthetic data. The model captures modalities of visual features with semantic class embeddings as the prototype to recover the feature distribution from the semantic attributes. The features generated by the conditional GAN can alleviate the long tail distribution problem of the samples. The experimental results demonstrate that the proposed generative model can synthesize discriminative features which shared similar distribution with real features, and can be used for classification tasks, image retrieval, and semantic annotation tasks.

References

[1] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2015, vol. 1, no. 3, p. 6.

[2] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative CNNs," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 5, pp. 2811–2821, May 2018.

[3] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Appl. Soft Comput.*, vol. 70, pp. 41–65, Sep. 2018.

[4] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: A large data set for nonparametric object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1958–1970, Nov. 2008.

[5] G. Cheng, R. Li, C. Lang, and J. Han, "Task-wise attention guided part complementary learning for few-shot image classification," *Sci. China Inf. Sci.*, vol. 64, no. 2, pp. 1–14, 2021.

[6] G. Cheng *et al.*, "Prototype-CNN for few-shot object detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–10, 2022.

[7] Z. Lin, G. Ding, J. Han, and J. Wang, "Cross-view retrieval via probability-based semantics-preserving hashing," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4342–4355, Dec. 2016.

[8] X. Lu, Y. Yuan, and X. Zheng, "Joint dictionary learning for multispectral change detection," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 884–897, Apr. 2016.

[9] J. Wang, W. Feng, Y. Chen, H. Yu, M. Huang, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 402–410.

[10] J. Li, E. Chen, Z. Ding, L. Zhu, K. Lu, and Z. Huang, "Cycle-consistent conditional adversarial transfer networks," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 747–755.

[11] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2251–2265, Sep. 2018.

[12] C. Zhang, X. Lyu, and Z. Tang, "TGG: Transferable graph generation for zero-shot and few-shot learning," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1641–1649.

[13] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, Mar. 2013.

[14] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[15] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, "Evaluation of output embeddings for fine-grained image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2927–2936.

[16] Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 584–599.

[17] T. Bin *et al.*, "Adversarial zero-shot learning with semantic augmentation," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 2476–2483.

[18] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5542–5551.

[19] M. B. Sariyildiz and R. G. Cinbis, "Gradient matching generative networks for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2168–2178.

[20] S. Min, H. Yao, H. Xie, Z.-J. Zha, and Y. Zhang, "Domain-specific embedding network for zero-shot recognition," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 2070–2078.

[21] T. Long, X. Xu, Y. Li, F. Shen, J. Song, and H. T. Shen, "Pseudo transfer with marginalized corrupted attribute for zero-shot learning," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1802–1810.

[22] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 951–958.

[23] Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, and Y. Matsumoto, "Ridge regression, hubness, and zero-shot learning," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Springer, 2015, pp. 135–151.

[24] E. Kodirov, T. Xiang, Z. Fu, and S. Gong, "Unsupervised domain adaptation for zero-shot learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2452–2460.

[25] X. Xu, F. Shen, Y. Yang, D. Zhang, H. T. Shen, and J. Song, "Matrix tri-factorization with manifold regularizations for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3798–3807.

[26] Z. Zhang and V. Saligrama, "Zero-shot learning via semantic similarity embedding," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4166–4174.

[27] Z. Zhang and V. Saligrama, "Zero-shot learning via joint latent similarity embedding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 6034–6042.

[28] S. Changpinyo, W.-L. Chao, B. Gong, and F. Sha, "Synthesized classifiers for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5327–5336.

[29] Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele, "Latent embeddings for zero-shot classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 69–77.

[30] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka, "Metric learning for large scale image classification: Generalizing to new classes at near-zero cost," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2012, pp. 488–501.

[31] M. Bucher, S. Herbin, and F. Jurie, "Improving semantic embedding consistency by metric learning for zero-shot classiffication," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 730–746.

[32] F. Pourpanah *et al.*, "A review of generalized zero-shot learning methods," 2020, *arXiv:2011.08641*.

[33] L. Feng and C. Zhao, "Transfer increment for generalized zero-shot learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2506–2520, Jun. 2021.

[34] G.-S. Xie *et al.*, "Generalized zero-shot learning with multiple graph adaptive generative networks," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 25, 2021, doi: 10.1109/TNNLS.2020.3046924.

[35] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 935–943.

[36] E. Schonfeld, S. Ebrahimi, S. Sinha, T. Darrell, and Z. Akata, "Generalized zero- and few-shot learning via aligned variational autoencoders," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8247–8255.

[37] J. Li, M. Jing, K. Lu, L. Zhu, Y. Yang, and Z. Huang, "Alleviating feature confusion for generative zero-shot learning," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1587–1595.

[38] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[39] E. L. Denton *et al.*, "Deep generative image models using a Laplacian pyramid of adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1486–1494.

[40] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2152–2161.

[41] F. Jurie, M. Bucher, and S. Herbin, "Generating visual representations for zero-shot classification," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2666–2673.

[42] Y. Zhu, M. Elhoseiny, B. Liu, X. Peng, and A. Elgammal, "A generative adversarial approach for zero-shot learning from noisy texts," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1004–1013.

[43] R. Felix, V. B. Kumar, I. Reid, and G. Carneiro, "Multi-modal cycle-consistent generalized zero-shot learning," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 21–37.

[44] H. Zhang, Y. Long, L. Liu, and L. Shao, "Adversarial unseen visual feature synthesis for zero-shot learning," *Neurocomputing*, vol. 329, pp. 12–20, Feb. 2019.

[45] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5767–5777.

[46] Z. Ding, M. Shao, and Y. Fu, "Low-rank embedded ensemble semantic dictionary for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2050–2058.

[47] Z. Ding and H. Liu, "Marginalized latent semantic encoder for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6191–6199.

[48] Z. Ding, M. Shao, and Y. Fu, "Generative zero-shot learning via low-rank embedded semantic dictionary," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 12, pp. 2861–2874, Dec. 2019.

[49] J. Li, M. Jing, K. Lu, Z. Ding, L. Zhu, and Z. Huang, "Leveraging the invariant side of generative zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7402–7411.

[50] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[51] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," 2017, *arXiv:1710.09412*.

[52] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-UCSD birds-200-2011 dataset," California Inst. Technol., Tech. Rep. CNS-TR-2011-001, 2011.

[53] G. Patterson and J. Hays, "SUN attribute database: Discovering, annotating, and recognizing scene attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2751–2758.

[54] Y. Xian, B. Schiele, and Z. Akata, "Zero-shot learning—The good, the bad and the ugly," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4582–4591.

[55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[56] A. Frome *et al.*, "Devise: A deep visual-semantic embedding model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2121–2129.

[57] V. K. Verma, G. Arora, A. Mishra, and P. Rai, "Generalized zero-shot learning via synthesized examples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4281–4289.

[58] P. Grnarova *et al.*, "A domain agnostic measure for monitoring and evaluating GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 12092–12102.

Bingrong Xu received the B.S. degree from the School of Automation, Wuhan University of Technology, Wuhan, China, in 2015, and the Ph.D. degree in control science and engineering from the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, in 2021.

Her current research interests include zero-shot learning, transfer learning, sparse representation, and low-rank representation.

Zhigang Zeng (Fellow, IEEE) received the Ph.D. degree in systems analysis and integration from the Huazhong University of Science and Technology, Wuhan, China, in 2003.

He is currently a Professor with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, and also with the Key Laboratory of Image Processing and Intelligent Control of the Education Ministry of China, Wuhan. He has authored or coauthored over 200 international journal papers. His current research interests include the theory of functional differential equations and differential equations with discontinuous right-hand sides, and their applications to dynamics of neural networks, memristive systems, and control systems.

Dr. Zeng has been an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS from 2010 to 2011, the IEEE TRANSACTIONS ON CYBERNETICS since 2014, the IEEE TRANSACTIONS ON FUZZY SYSTEMS since 2016, and a member of the Editorial Board of *Neural Networks* since 2012, *Cognitive Computation* since 2010, *Applied Soft Computing* in since 2013.

Cheng Lian (Member, IEEE) received the B.S. degree in electrical engineering and automation and the M.S. degree in control science and engineering from the School of Automation, Wuhan University of Technology, Wuhan, China, in 2008 and 2011, respectively, and the Ph.D. degree in control science and engineering from the School of Automation, Huazhong University of Science and Technology, Wuhan, in 2014.

He is currently an Associate Professor with the School of Automation, Wuhan University of Technology. His current research interests include machine learning, data mining, and pattern recognition.

Zhengming Ding (Member, IEEE) received the B.Eng. degree in information security and the M.Eng. degree in computer software and theory from the University of Electronic Science and Technology of China (UESTC), China, in 2010 and 2013, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA, in 2018.

He was a Faculty Member affiliated with the Department of Computer, Information and Technology, Indiana University-Purdue University Indianapolis. He has been a Faculty Member affiliated with the Department of Computer Science, Tulane University, New Orleans, LA, USA, since 2021. His research interests include transfer learning, multiview learning, and deep learning.

Dr. Ding is a member of ACM and AAAI. He received the National Institute of Justice Fellowship from 2016 to 2018. He was the recipient of the Best Paper Award (SPIE 2016) and Best Paper Candidate (ACM MM 2017). He is currently an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT), *Journal of Electronic Imaging (JEI)*, and *IET Image Processing*.