GENERALIZED ZERO-SHOT LEARNING USING CONDITIONAL WASSERSTEIN AUTOENCODER

Junhan Kim and Byonghyo Shim

Department of Electrical and Computer Engineering, Seoul National University, Seoul, Korea
Email: {junhankim, bshim}@islab.snu.ac.kr

ABSTRACT

Generalized zero-shot learning (GZSL) is a technique to train a deep learning model to identify unseen classes. Conventionally, conditional generative models have been employed to generate training data for unseen classes from the attribute. In this paper, we propose a new conditional generative model that improves the GZSL performance greatly. In a nutshell, the proposed model, called conditional Wasserstein autoencoder (CWAE), minimizes the Wasserstein distance between the real and generated image feature distributions using an encoder-decoder architecture. From the extensive experiments on various benchmark datasets, we show that the proposed CWAE outperforms conventional generative models in terms of the GZSL classification performance.

***Index Terms*—** Generalized zero-shot learning, generative model, generative adversarial network, variational autoencoder

1. INTRODUCTION

Image classification is a long-standing yet important task with a wide range of applications including autonomous driving, industrial automation, and medical diagnosis [1, 2]. In solving the task, supervised learning (SL) techniques have been popularly used for its excellent performance [3]. Well-known drawback of SL is that a large number of training data are required for each and every class to be identified. Unfortunately, in many practical cases, it is difficult to collect training data for certain classes such as endangered or newly observed species. When there are *unseen* classes where training data is unavailable, SL-based models are biased towards the *seen* classes, impeding the identification of the unseen classes.

Recently, to overcome this drawback, a technique to use the *attribute*, manually annotated visually distinctive characteristics of classes (e.g., color, size, and shape; see Fig. 1), in the unseen class identification has been proposed [4,5]. In this technique, dubbed as generalized zero-shot learning (GZSL), the relationship between the image feature and the attribute is learned from seen classes and then used for the identification of unseen classes. In [6], for example, a network estimating the attribute from the image feature has been used. In [7], an approach to measure the compatibility between the image feature and the attribute has been proposed. In [8,9], an approach to generate the image feature from the attribute has been proposed. Among these methods, the generation-based approach has received special attention for its superiority [8–14]. In this approach, generative adversarial network (GAN)-based models (e.g., conditional GAN (CGAN) [16] and conditional Wasserstein GAN (CWGAN) [9]) have been employed as a main network to synthesize the image feature from the attribute. While the GAN-based models have shown great potential for many data generation tasks [9, 16], it is well-known that the training of GAN-based models is very difficult and unstable [18, 19].

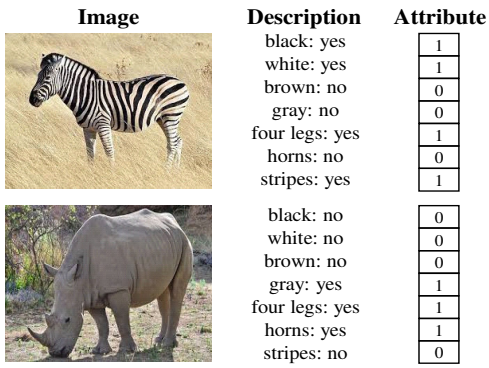

Fig. 1. Examples of the attribute. Types (e.g., 'black' and 'stripes') of attribute are the same for all classes, but the values are different.

In this paper, we propose a new conditional generative model that mitigates the training difficulty in the GAN-based models and also improves the GZSL performance greatly. To generate real-like image features from a given attribute, the proposed model, henceforth referred to as conditional Wasserstein autoencoder (CWAE), minimizes the Wasser-

This work was supported in part by the Samsung Research Funding & Incubation Center for Future Technology of Samsung Electronics under Grant SRFC-IT1901-17 and in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) under Grant 2020R1A2C2102198.

stein distance between the real and generated image feature distributions. To do so, CWAE uses an encoder-decoder architecture where the encoder compresses a high-dimensional image feature into a low-dimensional standard normal latent vector and the decoder reconstructs the original image feature using the latent vector and the attribute. The encoder and the decoder are jointly trained to minimize the reconstruction error (difference between the original and reconstructed image features), and an additional adversarial network is used in the training to enforce the encoder output distribution to be standard normal. Our theoretical analysis demonstrates that by taking such training method, the Wasserstein distance between the real and generated image feature distributions can be reduced. When the training is finished, CWAE only uses the decoder for the purpose of synthesizing image features.

We mention that the proposed CWAE is similar to the conventional GAN-based models in the sense that an adversarial network is used in the training to model some target distribution (standard normal distribution of the encoder output in our case). The main difference between CWAE and GAN-based models lies in the target distribution of the adversarial training. Specifically, while the GAN-based models aim to learn a high-dimensional image feature distribution by exploiting an adversarial network, CWAE aims to model a low-dimensional standard normal distribution. By moving the target distribution from a high-dimensional complex distribution to a low-dimensional simple distribution, CWAE can mitigate the training issues in the GAN-based models. From extensive experiments on benchmark datasets (AwA1 [4], AwA2 [20], CUB [21], and SUN [22]), we show that CWAE outperforms conventional GAN-based models by a large margin. For example, for the AwA1 dataset, CWAE achieves 5% improvement in the GZSL classification accuracy over conventional models.

2. CWAE

To generate real-like image features from a given attribute, the proposed CWAE uses two main networks in the training process, a decoder (generator) and an encoder. In this section, we describe CWAE with emphasis on these two networks.

2.1. Generator

For given image attribute $\mathbf{a}$, the generator G generates an image feature satisfying $\mathbf{a}$ from a latent vector $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Let $\widetilde{\mathbf{x}} = G(\mathbf{z}, \mathbf{a})$ be the generated image feature, then the probability density function (PDF) $p_g(\widetilde{\mathbf{x}}|\mathbf{a})$ of $\widetilde{\mathbf{x}}$ can be expressed as

$$p_g(\widetilde{\mathbf{x}}|\mathbf{a}) = \int_{\mathbf{z}} p(\mathbf{z})\delta(\widetilde{\mathbf{x}} - G(\mathbf{z}, \mathbf{a}))d\mathbf{z}, \tag{1}$$

where $p(\mathbf{z})$ is the PDF of $\mathbf{z}$ and δ is the Dirac delta function. To generate real-like image features, we train the generator in a way to minimize the distance between the real image feature distribution P_r and the generated image feature distribution P_g. In measuring the distance between the two distributions, we use Wasserstein distance[1] defined as

$$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} \mathbb{E}_{(\mathbf{x}, \widetilde{\mathbf{x}}) \sim \gamma(\mathbf{x}, \widetilde{\mathbf{x}})}[\|\mathbf{x} - \widetilde{\mathbf{x}}\|_2], \tag{2}$$

where $\Pi(P_r, P_g)$ is the set of joint PDFs of real and generated image features. Then, the loss function $\mathcal{L}_G$ of the generator can be expressed as

$$\mathcal{L}_G = W(P_r, P_g). \tag{3}$$

We note that it is intractable to compute $W(P_r, P_g)$ using (2) since an exhaustive search over all joint PDFs $\gamma \in \Pi(P_r, P_g)$ is needed. In the following theorem, we present a simplified form of $W(P_r, P_g)$.

Theorem 1. *Let $\mathcal{Q}$ be the set of conditional PDFs $p(\mathbf{z}|\mathbf{x})$ of a latent vector $\mathbf{z} \sim p(\mathbf{z})$ given $\mathbf{x} \sim P_r$, then $W(P_r, P_g)$ can be expressed as*

$$W(P_r, P_g) = \inf_{p(\mathbf{z}|\mathbf{x}) \in \mathcal{Q}} \mathbb{E}_{\mathbf{x} \sim P_r}\left[\mathbb{E}_{\mathbf{z} \sim p(\mathbf{z}|\mathbf{x})}[\|\mathbf{x} - G(\mathbf{z}, \mathbf{a})\|_2]\right]. \tag{4}$$

Proof. One can show that for each joint PDF $\gamma \in \Pi(P_r, P_g)$, there exists $p(\mathbf{z}|\mathbf{x}) \in \mathcal{Q}$ such that (see Appendix A)

$$\gamma(\mathbf{x}, \widetilde{\mathbf{x}}) = \int_{\mathbf{z}} p_r(\mathbf{x}|\mathbf{a})p(\mathbf{z}|\mathbf{x})\delta(\widetilde{\mathbf{x}} - G(\mathbf{z}, \mathbf{a}))d\mathbf{z}. \tag{5}$$

Then, we have

$$\begin{aligned}
&\mathbb{E}_{(\mathbf{x}, \widetilde{\mathbf{x}}) \sim \gamma(\mathbf{x}, \widetilde{\mathbf{x}})}[\|\mathbf{x} - \widetilde{\mathbf{x}}\|_2] \\
&= \int_{\mathbf{x}} \int_{\widetilde{\mathbf{x}}} \|\mathbf{x} - \widetilde{\mathbf{x}}\|_2 \int_{\mathbf{z}} p_r(\mathbf{x}|\mathbf{a})p(\mathbf{z}|\mathbf{x})\delta(\widetilde{\mathbf{x}} - G(\mathbf{z}, \mathbf{a}))d\mathbf{z}d\widetilde{\mathbf{x}}d\mathbf{x} \qquad (6) \\
&\overset{(a)}{=} \int_{\mathbf{x}} p_r(\mathbf{x}|\mathbf{a}) \left(\int_{\mathbf{z}} p(\mathbf{z}|\mathbf{x}) \|\mathbf{x} - G(\mathbf{z}, \mathbf{a})\|_2 d\mathbf{z} \right) d\mathbf{x} \qquad (7) \\
&= \mathbb{E}_{\mathbf{x} \sim P_r}\left[\mathbb{E}_{\mathbf{z} \sim p(\mathbf{z}|\mathbf{x})}[\|\mathbf{x} - G(\mathbf{z}, \mathbf{a})\|_2]\right], \qquad (8)
\end{aligned}$$

where (a) is from the sifting property of the delta function. By combining (2) and (8), we obtain the desired result in (4). □

In contrast to the original form in (2) that requires an exhaustive search over joint PDFs of two high-dimensional random variables $\mathbf{x}$ and $\widetilde{\mathbf{x}}$, the modified form in (4) requires to search PDFs of one low-dimensional latent vector $\mathbf{z}$. By combining (3) and (4), we obtain

$$\mathcal{L}_G = \inf_{p(\mathbf{z}|\mathbf{x}) \in \mathcal{Q}} \mathbb{E}_{\mathbf{x} \sim P_r}\left[\mathbb{E}_{\mathbf{z} \sim p(\mathbf{z}|\mathbf{x})}[\|\mathbf{x} - G(\mathbf{z}, \mathbf{a})\|_2]\right]. \tag{9}$$

[1] In [17], it has been theoretically analyzed that Wasserstein distance is much more sensible for the network training than other well-known measures such as Kullback-Leibler (KL) and Jensen-Shannon (JS) divergences.

2.2. Encoder

From (9), one can observe that for each real image feature $\mathbf{x}$ and the attribute $\mathbf{a}$, the latent vector $\mathbf{z} \sim p(\mathbf{z}|\mathbf{x})$ minimizing the reconstruction error $\|\mathbf{x} - G(\mathbf{z}, \mathbf{a})\|_2$ is needed to compute the generator loss function $\mathcal{L}_G$. To find out such $\mathbf{z}$, we use an encoder E trained to minimize

$$\mathcal{L}_E = \mathbb{E}_{\mathbf{x}\sim P_r}\left[\|\mathbf{x} - G\left(\mathbf{z}_{\text{enc}}, \mathbf{a}\right)\|_2\right] + \lambda \mathcal{L}_{Reg}, \tag{10}$$

where $\mathbf{z}_{\text{enc}} = E(\mathbf{x}, \mathbf{a})$, λ is a regularization coefficient, and $\mathcal{L}_{Reg}$ is the loss term to enforce $\mathbf{z}_{\text{enc}} \sim p(\mathbf{z})$.[2]

In computing $\mathcal{L}_{Reg}$, we use an additional network D estimating the probability for which $\mathbf{z}_{\text{enc}} \sim p(\mathbf{z})$. We note that if $\mathbf{z}_{\text{enc}} \sim p(\mathbf{z})$, then $D(\mathbf{z}_{\text{enc}})$ would be close to one. Thus, by minimizing

$$\mathcal{L}_{Reg} = \left(D(\mathbf{z}_{\text{enc}}) - 1\right)^2 = \left(D(E(\mathbf{x}, \mathbf{a})) - 1\right)^2, \tag{11}$$

we can maximize the probability for which $\mathbf{z}_{\text{enc}} \sim p(\mathbf{z})$.[3] By combining (10) and (11), we obtain

$$\mathcal{L}_E = \mathbb{E}_{\mathbf{x}\sim P_r}\Big[\|\mathbf{x} - G\left(\mathbf{z}_{\text{enc}}, \mathbf{a}\right)\|_2 + \lambda \left(D(\mathbf{z}_{\text{enc}}) - 1\right)^2\Big].$$

In training D, we employ an adversarial training strategy. Specifically, we draw a sample $\mathbf{z} \sim p(\mathbf{z})$ and then train D to output one for $\mathbf{z}$ and zero for $\mathbf{z}_{\text{enc}}$. The loss function $\mathcal{L}_D$ can be expressed as

$$\mathcal{L}_D = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[(D(\mathbf{z}) - 1)^2] + \mathbb{E}_{\mathbf{x}\sim P_r}[(D(\mathbf{z}_{\text{enc}}))^2]. \tag{12}$$

Comparison with GAN-based models The proposed CWAE is similar to conventional GAN-based models in the sense that an adversarial network is used in the generative model training. The key difference between CWAE and the GAN-based models is that while GAN-based networks aim to model a high-dimensional image feature distribution using an adversarial network [9], CWAE tries to model a low-dimensional standard normal distribution. By moving the target distribution from a high-dimensional complex distribution to a low-dimensional simple distribution, CWAE can mitigate the training difficulty in the conventional GAN-based models.

3. EXPERIMENT

3.1. Experimental Setup

Datasets We evaluate the performance of CWAE using four benchmark datasets: AwA1, AwA2, CUB, and SUN. The AwA1 and AwA2 datasets contain 50 classes of animal images annotated with 85 attributes [4, 20]. The CUB dataset contains 200 species of bird images annotated with 312 attributes [21]. The SUN dataset contains 717 classes of scene images annotated with 102 attributes [22]. In dividing the total classes into seen and unseen classes, we adopt the conventional dataset split presented in [20].

[2] In fact, $\mathbf{z}_{\text{enc}}$ should satisfy $\mathbf{z}_{\text{enc}} \sim p(\mathbf{z}|\mathbf{x})$. Since $\mathbb{E}_{\mathbf{x}\sim P_r}[p(\mathbf{z}|\mathbf{x})] = \int_{\mathbf{x}} p(\mathbf{z}|\mathbf{x})p(\mathbf{x})d\mathbf{x} = \int_{\mathbf{x}} p(\mathbf{z}, \mathbf{x})d\mathbf{x} = p(\mathbf{z})$, we enforce $p(\mathbf{z}|\mathbf{x})$ to be the same as $p(\mathbf{z})$ for each $\mathbf{x}$.

[3] Instead of binary cross-entropy loss, we use the least squares-based loss in (11), which has led to better performance in conventional generative models (e.g., least squares GAN (LSGAN) [23]).

	AwA1	AwA2	CUB	SUN
CGAN	66.4	72.8	**55.1**	51.8
CLSGAN	67.2	**72.9**	54.5	52.8
CWGAN	**68.6**	71.9	**56.0**	59.1
CVAE	67.4	72.3	54.8	**59.5**
CWAE	**71.1**	**74.7**	**56.0**	**61.4**

Table 1. ZSL performance of conditional generative models.

Implementation details In extracting image features, we use ResNet-101 pre-trained on the ImageNet [3]. As in [9], we fix ResNet in the training process. All the networks (encoder, generator, and discriminator) are implemented using the multilayer perceptron (MLP) with one hidden layer. As in [9], we set the number of hidden units to 4096 and use LeakyReLU as a nonlinear activation function. For comparison, we also consider CGAN [16], conditional least squares GAN (CLSGAN) [23], CWGAN [17], and conditional variational autoencoder (CVAE) [15] in our experiments.

3.2. Results

First, we consider the ZSL case where image samples of only unseen classes are given in the test phase. To evaluate the performance of generative models, we take the following steps whenever generative models are trained for one epoch:

1) Synthesize image features of unseen classes using the generative models.
2) Train a softmax classifier using the synthetic image features.
3) Measure the top-1 accuracy of the classifier using real image features of unseen classes.

We train generative models for 100 epochs and use the maximum top-1 accuracy as a metric to evaluate the performance of models. In Table 1, we summarize the ZSL performance of generative models for each dataset. One can see that the proposed CWAE outperforms conventional generative models for all the datasets. Specifically, for the AwA1, AwA2, and SUN datasets, CWAE achieves about 2% improvement in the classification accuracy over conventional models. In particular, for the AwA1 dataset, the classification accuracy obtained by CWAE is 2.5% higher than those obtained by conventional models.

Next, we consider the more realistic scenario where image samples of both seen and unseen classes are given in the test phase. In this case, we follow the standard evaluation process presented in [20]. Specifically, we synthesize image features of both seen and unseen classes using a generative

	AwA1			AwA2			CUB			SUN		
	acc_s	acc_u	acc_h	acc_s	acc_u	acc_h	acc_s	acc_u	acc_h	acc_s	acc_u	acc_h
CGAN	64.3	53.0	58.1	65.8	57.0	61.1	45.3	30.4	36.3	22.0	21.9	21.9
CLSGAN	62.2	56.7	**59.3**	66.5	59.9	**63.0**	43.0	30.8	35.9	21.8	19.6	20.6
CWGAN	67.2	52.6	59.0	70.7	53.9	61.2	50.1	41.9	**45.6**	27.6	25.7	26.6
CVAE	64.0	55.2	**59.3**	72.9	54.5	62.4	46.0	42.8	44.3	31.5	23.9	**27.2**
CWAE	70.2	58.9	**64.0**	71.0	60.0	**65.1**	48.8	45.3	**47.0**	33.4	27.4	**30.1**

Table 2. GZSL performance of conditional generative models.

	AwA1			AwA2			CUB			SUN		
	acc_s	acc_u	acc_h	acc_s	acc_u	acc_h	acc_s	acc_u	acc_h	acc_s	acc_u	acc_h
SE-GZSL (CVAE+Reg.) [8]	67.8	56.3	61.5	68.1	58.3	62.8	53.3	41.5	46.7	30.5	40.9	34.9
f-CLSWGAN (CWGAN+Cls.) [9]	61.4	57.9	59.6	-	-	-	57.7	43.7	49.7	36.6	42.6	39.4
f-VAEGAN-D2 (CVAE+Dis.) [10]	-	-	-	70.6	57.6	**63.5**	60.1	48.4	**53.6**	38.0	45.1	**41.3**
LisGAN (CWGAN+Cls.) [11]	76.3	52.6	62.3	-	-	-	57.9	46.5	**51.6**	37.8	42.9	40.2
DASCN (CWGAN+Reg.) [12]	68.0	59.3	**63.4**	-	-	-	59.0	45.9	**51.6**	38.5	42.4	40.3
Zero-VAE-GAN (CVAE+Dis.) [13]	66.8	58.2	62.3	70.9	57.1	62.5	47.9	43.6	45.5	30.2	45.2	36.3
CWAE without regularizers	70.2	58.9	**64.0**	71.0	60.0	**65.1**	55.2	47.5	51.0	39.1	48.3	**43.2**

Table 3. GZSL performance of generative models combined with additional sub-networks. 'Reg.', 'Cls.', and 'Dis.' denote the attribute estimator, image feature classifier, and discriminator, respectively. '-' means that the accuracy is not reported in references. To compare CWAE with conventional methods under the same condition, we use real image features of seen classes in the classifier training in this experiment.

model[4] and then train a classifier using the synthetic image features. After the classifier training, we measure the average top-1 classification accuracies acc_s and acc_u on seen and unseen classes, respectively, and then compute their harmonic mean acc_h. In Table 2, we summarize the harmonic mean accuracy of generative models on different datasets. One can observe that for all the datasets, the proposed CWAE outperforms other conditional generative models by a large margin. In particular, for the AwA1 dataset, CWAE achieves about 5% improvement in the harmonic mean accuracy over conventional generative models. Also, for the AwA2, CUB, and SUN datasets, CWAE achieves about 2%, 1.5%, and 3% improvement in the harmonic mean accuracy, respectively.

So far, we have considered the GZSL performance of pure generative models. Recently, various auxiliary networks have been combined with generative models to improve the quality of generated image features [9–13]. For example, in [8, 12], a network estimating the attribute of image features has been used in the generative model training to make sure that synthetic image features satisfy the attribute of unseen classes. In [9, 11, 13], an additional image feature classifier has been used to generate sufficiently distinct image features for different classes. In [10, 13], an additional discriminator has been used to improve the compatibility between generated image features and attributes. In Table 3, we summarize the performance of generative models combined with these networks. From the results, one can observe that the performance of CWAE is competitive even though CWAE does not use any additional network. In particular, for the AwA1, AwA2, and SUN datasets, the harmonic mean accuracies acc_h obtained by CWAE are 0.6%, 1.6% and 1.9% higher than those obtained by conventional techniques.

[4]Although real image features are available for seen classes, we use synthetic image features in the classifier training to compare the performance of different generative models.

4. CONCLUSION

In this paper, we proposed a new conditional generative model called CWAE. To generate real-like image features from the attribute, CWAE minimizes the Wasserstein distance between the real and generated image feature distributions using an encoder-decoder architecture. From the experimental results on various benchmark datasets, we demonstrated that CWAE outperforms conventional generative models in terms of the GZSL classification performance.

A. PROOF OF (5)

Let $p(\mathbf{x}, \widetilde{\mathbf{x}}, \mathbf{z})$ be a joint PDF of $\mathbf{x} \sim P_r$, $\widetilde{\mathbf{x}} \sim P_g$, and $\mathbf{z} \sim P_z$ such that $\gamma(\mathbf{x}, \widetilde{\mathbf{x}}) = \int_{\mathbf{z}} p(\mathbf{x}, \widetilde{\mathbf{x}}, \mathbf{z}) d\mathbf{z}$. From the definition of the conditional PDF, we have

$$\gamma(\mathbf{x}, \widetilde{\mathbf{x}}) = \int_{\mathbf{z}} p_r(\mathbf{x}|\mathbf{a}) p(\mathbf{z}|\mathbf{x}) p(\widetilde{\mathbf{x}}|\mathbf{z}, \mathbf{x}) d\mathbf{z}. \tag{13}$$

Also, since the generated image feature $\widetilde{\mathbf{x}}$ is determined as $G(\mathbf{z}, \mathbf{a})$ when the latent vector $\mathbf{z}$ is given, we have

$$p(\widetilde{\mathbf{x}}|\mathbf{z}, \mathbf{x}) = \delta(\widetilde{\mathbf{x}} - G(\mathbf{z}, \mathbf{a})). \tag{14}$$

By combining (13) and (14), we obtain the result in (5).

B. REFERENCES

[1] H. Fujiyoshi, T. Hirakawa, and T. Yamashita, "Deep learning-based image recognition for autonomous driving," *IATSS Research*, vol. 43, no. 4, pp. 244–252, 2019. 1

[2] R. Ren, T. Hung, and K. C. Tan, "A generic deep-learning-based approach for automated surface inspection," *IEEE Trans. Cybernetics*, vol. 48, no. 3, pp. 929–940, 2017. 1

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778. 1, 3

[4] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. CVPR*, 2009, pp. 951–958. 1, 2, 3

[5] W. L. Chao, S. Changpinyo, B. Gong, and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild," in *Proc. ECCV*, 2016, pp. 52–68. 1

[6] C. H. Lampert, H. Nickisch, and S. Harmeling, Attribute-based classification for zero-shot visual object categorization, *IEEE Trans. Pattern Analysis Mach. Intell.*, vol. 36, no. 3, pp. 453–465, 2013. 1

[7] Z. Akata, F. Perronni, Z. Harchaoui, and C. Schmid, "Label-embedding for image classification," *IEEE Trans. Pattern Analysis Mach. Intell.*, vol. 38, no. 7, pp. 1425–1438, 2016. 1

[8] V. Kumar Verma, G. Arora, A. Mishra, and P. Rai, "Generalized zero-shot learning via synthesized examples," in *Proc. CVPR*, 2018, pp. 4281–4289. 1, 4

[9] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *Proc. CVPR*, 2018, pp. 5542–5551. 1, 3, 4

[10] Y. Xian, S. Sharma, B. Schiele, and Z. Akata, "f-VAEGAN-D2: A feature generating framework for any-shot learning," in *Proc. CVPR*, 2019, pp. 10275–10284. 1, 4

[11] J. Li, M. Jing, K. Lu, Z. Ding, L. Zhu, and Z. Huang, "Leveraging the invariant side of generative zero-shot learning," in *Proc. CVPR*, 2019, pp. 7402–7411. 1, 4

[12] J. Ni, S. Zhang, and H. Xie, "Dual adversarial semantics-consistent network for generalized zero-shot learning," *arXiv:1907.05570*, 2019. 1, 4

[13] R. Gao, X. Hou, J. Qin, J. Chen, L. Liu, F. Zhu, Z. Zhang, and L. Shao, "Zero-VAE-GAN: Generating unseen features for generalized and transductive zero-shot learning," *IEEE Trans. Image Process.*, vol. 29, pp. 3665–3680, 2020. 1, 4

[14] J. Kim, K. Shim, and B. Shim, "Semantic feature extraction for generalized zero-shot learning," *arXiv:2112.14478*, 2021. 1

[15] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in Neural Information Processing Systems*, vol. 28, pp. 3483–3491, 2015. 3

[16] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv:1411.1784*, 2014. 1, 3

[17] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *arXiv:1701.07875*, 2017. 2, 3

[18] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," *arXiv:1606.03498*, 2016. 1

[19] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," *arXiv:1701.04862*, 2017. 1

[20] Y. Xian, B. Schiele, and Z. Akata, "Zero-shot learning-the good, the bad and the ugly," in *Proc. CVPR*, 2017, pp. 4582–4591. 2, 3

[21] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD birds 200," 2010. 2, 3

[22] G. Patterson and J. Hays, "Sun attribute database: Discovering, annotating, and recognizing scene attributes," in *Proc. CVPR*, 2012, pp. 2751–2758. 2, 3

[23] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proc. ICCV*, 2017, pp. 2794–2802.

3