Exploring Intra-class Variation Factors with Learnable Cluster Prompts for Semi-supervised Image Synthesis

Yunfei Zhang[1*], Xiaoyang Huo[1*], Tianyi Chen[1], Si Wu[1,2,3†], and Hau San Wong[4]
[1]School of Computer Science and Engineering, South China University of Technology
[2]Peng Cheng Laboratory
[3]PAZHOU LAB
[4]Department of Computer Science, City University of Hong Kong
{cszhangyunfei, csxyhuo, csttychen}@mail.scut.edu.cn, cswusi@scut.edu.cn,
cshswong@cityu.edu.hk

Abstract

Semi-supervised class-conditional image synthesis is typically performed by inferring and injecting class labels into a conditional Generative Adversarial Network (GAN). The supervision in the form of class identity may be inadequate to model classes with diverse visual appearances. In this paper, we propose a Learnable Cluster Prompt-based GAN (LCP-GAN) to capture class-wise characteristics and intra-class variation factors with a broader source of supervision. To exploit partially labeled data, we perform soft partitioning on each class, and explore the possibility of associating intra-class clusters with learnable visual concepts in the feature space of a pre-trained language-vision model, e.g., CLIP. For class-conditional image generation, we design a cluster-conditional generator by injecting a combination of intra-class cluster label embeddings, and further incorporate a real-fake classification head on top of CLIP to distinguish real instances from the synthesized ones, conditioned on the learnable cluster prompts. This significantly strengthens the generator with more semantic language supervision. LCP-GAN not only possesses superior generation capability but also matches the performance of the fully supervised version of the base models: BigGAN and StyleGAN2-ADA, on multiple standard benchmarks.

1. Introduction

Generative Adversarial Networks (GANs) have achieved considerable success in modeling complex data distributions and generating high-fidelity images from random vectors [2, 18, 25]. To control class semantics in the generation process, object category is typically represented in the form of a discrete label, which is injected into both generator and discriminator through learnable embedding layers. However, sufficient labeled training data may be difficult to collect in real-world applications. Significant efforts have been devoted to semi-supervised generative learning that aims to reduce the dependence of class-conditional GANs on labeled training data [6, 15, 29, 33].

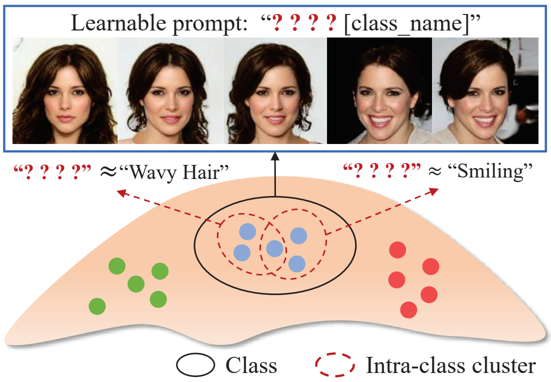

Figure 1. Different from generic class-conditional GANs conditioned on discrete class labels, LCP-GAN learns intra-class cluster-specific prompts to guide the generation process and capture underlying variation factors.

In the semi-supervised setting, the amount of unlabeled training samples can be significantly greater than that of labeled ones. As one of the early attempts, CatGAN [44] trained a discriminator to infer the class labels of real images with high confidence, but not for the synthesized ones. Both TripleGAN [29] and Δ-GAN [15] incorporated an auxiliary classifier to focus on class label prediction in the adversarial training process. The unlabeled images with pseudo labels were used to train the class-

[*]Joint first authors.
[†]Corresponding author.

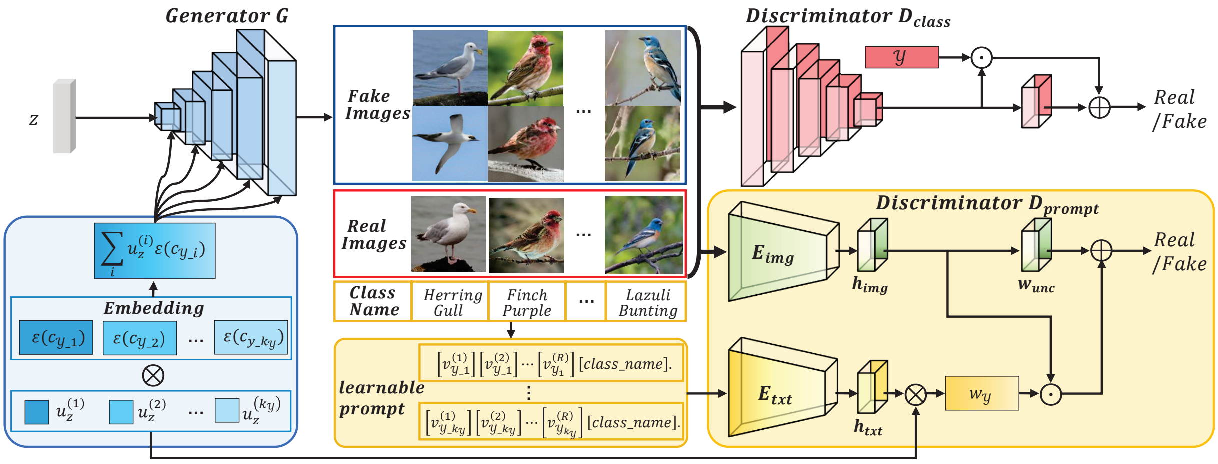

Figure 2. An overview of the proposed LCP-GAN. The generator G synthesizes class-specific images, conditioned on the combination of intra-cluster label embeddings: $\sum_i u_z^{(i)} \mathcal{E}(c_{y\_i})$. In addition to class-conditional adversarial training via the discriminator D_{class}, we incorporate an additional discriminator D_{prompt} to distinguish real images from the synthesized ones, conditioned on the combination of the learnable cluster-specific prompts $\{t_{y\_1}, \dots, t_{y\_k_y}\}$. By competing with D_{class} and D_{prompt}, G learns to associate the cluster label embeddings with the underlying visual concepts described by the prompts.

conditional discriminators. There have also been some attempts at improving GANs via unsupervised data partitioning [14, 16, 21, 31, 39]. SphericGAN [6] imposed hard partitioning on the training data, and aligned the real and synthesized data clusters in a hyper-spherical latent space. The existing semi-supervised GANs are conditioned on class identity, while the semantics encapsulated in the class names is overlooked. This impedes the generative model from using prior knowledge in natural language as humans do. Furthermore, only one single label embedding is learnt per class, which is insufficient to account for large intra-class variance. To address these issues, we explore intra-class variation factors by performing soft and finer partitions on each class and learning cluster-specific prompts to represent underlying visual concepts as shown in Figure 1.

More specifically, we propose a Learnable Cluster Prompt-based GAN to facilitate semi-supervised class-conditional image generation, and our model is referred to as LCP-GAN. To better match class-specific data distribution, the generator learns to synthesize high-fidelity images, conditioned on a combination of intra-class cluster label embeddings. Capturing intra-class variation factors is a non-trivial task, since the semantics reflected by the clusters may not be well-defined. Considering that natural language can express a wide range of visual concepts, we make an attempt to learn from CLIP [42], which provides an effective way to understand the content of images. Inspired by CoOp [56], we can model the cluster-specific context words with learnable vectors, and further learn a mapping to adapt the CLIP representation to our generation task. As a result, the generator is guided to capture cluster semantics in the adversarial training process. The framework of LCP-GAN is illustrated in Figure 2. We adopt the state-of-the-art architectures: BigGAN [2] and StyleGAN2-ADA [23], and achieve significant improvements over them, suggesting that semi-supervised image generation can benefit from modeling intra-class variation with the language-vision pre-training.

The main contributions of this work are summarized as follows: (a) We associate the intra-class cluster label embeddings with the cluster semantics, and the expressiveness of their combination is higher than that of a single class label embedding for capturing multiple underlying modes with diverse visual appearances. (b) To address the issue that the visual concepts reflected by the clusters may not be well defined, we leverage the language-vision pre-training and represent the clusters with learnable prompts. (c) To guide the generator to capture intra-class variation factors, the cluster prompts serve as conditional information and are jointly learnt in the adversarial training process.

2. Related Work

2.1. Generic GAN-based Image Synthesis

GANs have served as the leading models and brought rapid progress in image synthesis recently [11, 25, 51, 52]. GAN training is based on the minimax theorem and suffers from mode collapse due to optimization difficulties. There are a variety of approaches that focus on the stability of the adversarial training process, including Wasserstein distance-based distribution alignment [1], Lipschitz continuity [36, 47], minibatch discrimination [43], architectural constraints [24, 41, 50], and so on.

To synthesize class-specific images, a straightforward strategy [35] is to feed the class label into a generator and a discriminator together with a latent code and the produced image, respectively. To ensure correct class semantics of the synthesis results, Springenberg [44] proposed a Categorical GAN (CatGAN), in which the discriminator was trained to classify real images while aggregating the synthesized ones into an additional class. In addition, Odena et al. [40] and Gong et al. [17] enhanced conditional GANs by incorporating Auxiliary Classification heads in the discriminator (ACGANs). Kang et al. [22] further enhanced ACGAN by imposing inter-class separability regularization on the discriminator, and the resulting model is referred to as ReACGAN (Rebooted Auxiliary Classifier-based GAN). Another strategy is to learn the class-specific normalization parameters to regularize the generator features, while the discriminator distinguishes real images from the synthesized ones, conditioned on the learnable class label embeddings [2, 26, 37].

Recently, data augmentation techniques were used to cover unseen variations and proven to be effective in improving the generation performance of GANs [23, 55]. Chen et al. [5] proposed a self-supervised GAN, in which the discriminator was required to predict the rotation angles for the randomly rotated images. To prevent the synthesized images from matching the augmented data distribution instead of the original one, Zhao et al. [53] imposed multiple types of differentiable augmentation operations on both real and synthesized images, and the model could converge to a better solution. In [54], the augmentation operations were performed on latent vectors, and consistency regularization was imposed on the images synthesized from the resulting vectors. Instead of pre-specified augmentation operations, Karras et al. [23] adopted a reinforcement learning paradigm to optimize a GAN with adaptively augmented data.

2.2. Semi-supervised GANs

Semi-supervised generative learning aims to learn class-conditional data distribution on partially labeled data, and the synthesis results are expected to be realistic and reflect precise class semantics. To address the lack of labeled data, Li et al. [29] proposed Triple-GAN that incorporates a classifier to pseudo-label the unlabeled images as accurately as possible, such that the images can be identified as real by the class-conditional discriminator. To precisely capture class semantics, Wu et al. [48] improved Triple-GAN by imposing feature-semantics matching regularization on the generator. In Δ-GAN [15], Gan et al. adopted an additional discriminator to distinguish unlabeled images from the synthesized ones. To improve the discriminator's capability, the random regional replacement strategy [49] was leveraged to construct hard examples in R^3-CGAN [32] and MED-GAN [33], and the discriminator was encouraged to apply attention on the semantically meaningful regions. Another group of semi-supervised GANs [10, 13, 30, 43] focus more on image classification tasks, and the synthesized instances are used to extend the training data [8, 12].

Self-CondGAN [31] and SphericGAN [6] are mostly related to this work. We summarize the fundamental differences as follows: (1) Both Self-CondGAN and SphericGAN performed clustering on all the training data, and each cluster may partially cover multiple classes. In contrast, we perform soft partitioning on each class, which can explore intra-class variation factors without changing class semantics. (2) We exploit the language-vision pre-training to represent the underlying visual concepts reflected by the clusters. This has not been considered by the two methods.

3. Proposed Method

3.1. Overview

Given a random variable z drawn from a pre-defined Gaussian distribution and a class index y, the previous works typically train a class-conditional generator that synthesizes the instances $x_z \sim p_z(x_z|y)$ to match the y-th class data distribution $p_{data}(x|y)$. In this work, we aim to improve the class-conditional generation performance, and our key idea is to model a set of intra-class clusters denoted by $\mathbf{c}_y = \{c_{y\_1}, ..., c_{y\_k_y}\}$ in the CLIP feature space, such that the synthesized data $x_z \sim p_z(x_z|\mathbf{c}_y)$ can better encapsulate class semantics, where k_y denotes the number of clusters in class y. In addition to leveraging the semantic information encapsulated in the class name, we believe that the semantically meaningful variation factors can be better captured by learning from language-vision pre-training, and our design thus lies in how cluster prompts $\mathbf{t}_y = \{t_{y\_1}, \dots, t_{y\_k_y}\}$ are learnt and improve the generation process.

LCP-GAN mainly consists of four components: A predictor $C : \mathbb{R}^{h\times w\times 3} \to Y$ infers the class labels of unlabeled RGB images with resolution of $h \times w$, where Y denotes the label space. A conditional generator $G : \mathbb{R}^m \times \mathbb{R}^{k_y} \times \mathbb{R}^{k_y} \to \mathbb{R}^{h\times w\times 3}$ synthesizes an image $x_z = G(z, \mathbf{c}_y, u_z)$ from a m-dimensional random vector $z \in \mathbb{R}^m$ together with cluster labels and a coefficient vector $u_z \in \mathbb{R}^{k_y}$. A class-conditional discriminator $D_{class} : \mathbb{R}^{h\times w\times 3} \times Y \to \{0, 1\}$ distinguishes real images from the synthesized ones, conditioned on the class label. An additional discriminator $D_{prompt} : \mathbb{R}^{h\times w\times 3} \times \mathbb{R}^{k_y} \times \mathbb{R}^{k_y} \to \{0, 1\}$ performs adversarial training, conditioned on cluster prompt $\mathbf{t}_y$. Due to the lack of well-defined visual concepts on the clusters, we build D_{prompt} by incorporating a real-fake classification head on top of CLIP. In addition to competing with D_{class} to capture class semantics, G is also trained to deceive D_{prompt} by matching the semantics of the synthesized images with the cluster prompts. We judiciously design an optimization scheme to jointly train the components.

3.2. Cluster-conditional Generation

We model the intra-class clusters and learn the corresponding label embeddings to control the generation process. Toward this end, we perform intra-class data partitioning via soft k-means clustering in the feature space of a ResNet [19] pre-trained on ImageNet [9]. For simplicity, let x denote a labeled/unlabeled image, and the class label y is defined as follows:

$$y = \begin{cases} \texttt{Ground-truth}, & \text{if } x \text{ is labeled}, \\ \texttt{one-hot}(C(x)), & \text{otherwise}, \end{cases} \tag{1}$$

where $\texttt{one-hot}(\cdot)$ is the one-hot encoding function for pseudo-labeling unlabeled images. For class y, the images are divided into k_y clusters denoted by $\mathbf{c}_y = \{c_{y\_1}, ..., c_{y\_k_y}\}$, and the corresponding prototypes $\boldsymbol{\rho}_y = \{\rho_{y\_1}, ..., \rho_{y\_k_y}\}$ are computed by the weighted mean vectors of the embedded images as follows:

$$\rho_{y\_i} \leftarrow (1-\mu)\rho_{y\_i} + \mu u_x^{(i)} f(x), \tag{2}$$

where $f(\cdot)$ denotes the pre-trained network features, and the weighting factor μ controls the rate of moving average. In the above equation, the degree $u_x^{(i)}$ to which x belongs to intra-class cluster i is computed as follows:

$$u_x^{(i)} = \frac{\exp(\cos(\rho_{y\_i}, f(x))/\delta)}{\sum_j \exp(\cos(\rho_{y\_j}, f(x))/\delta)}, \tag{3}$$

where δ is a temperature parameter. For image generation, we simulate the condition by randomly sampling a coefficient vector $u_z = [u_z^{(1)}, \dots, u_z^{(k_y)}]$ to combine the cluster label embeddings, based on which the generator synthesizes an image conditioned as follows:

$$x_z = G\Big(z, \sum_i u_z^{(i)} \mathcal{E}(c_{y\_i})\Big), \tag{4}$$

where $\mathcal{E}(\cdot)$ denotes the learnable embedding layer.

3.3. Learning Cluster-specific Prompts

It is non-trivial to determine the context words accompanying each intra-class cluster due to the lack of prior knowledge on the underlying visual concepts. To address this issue, we jointly learn the cluster-specific context vectors in the adversarial training process, such that the generator can be guided with the supervision from CLIP, which consists of two encoders, one for producing prompt representation from context words and the other for mapping images into the same representation space. In LCP-GAN, the context words are in the form of continuous vectors that have the same dimension as the word embeddings. Specifically, we adopt the prompt form as follows:

$$t_{y\_i} = [v_{y\_i}^{(1)}][v_{y\_i}^{(2)}] \dots [v_{y\_i}^{(R)}][\texttt{class\_name}], \tag{5}$$

where the context vectors $\{v_{y\_i}^{(r)}\}_{r=1}^{R}$ are learnable, and the word embedding vector of the i-th class name is used in the token position $[\texttt{class\_name}]$.

The training objective is to pull the representations of cluster-specific images closer to the corresponding prompt. Let $E_{img}(x)$ denote the representation extracted by the CLIP image encoder E_{img}, and $E_{txt}(t_{y\_i})$ be the one generated by the CLIP text encoder E_{txt}. Maximizing the cosine similarity between the representations does not guarantee that the prompts represent the differences among intra-class clusters, and the generator may fail to capture the variation factors in this case. We address this issue by performing the prompt-conditional adversarial training, and the corresponding discriminator D_{prompt} is built by incorporating a real-fake classification head on top of the CLIP encoders. The head adopts a two-branch architecture, in which two learnable light-weight mappings $\{h_{txt}, h_{img}\}$ are used to transform the CLIP representations. The real instances should be distinguished from the synthesized ones in the task-specific embedding space. The conditional identification weight is defined as follows:

$$w_y = \sum_i u_x^{(i)} h_{txt}\big(E_{txt}(t_{y\_i})\big), \tag{6}$$

and the prediction probability is computed as follows:

$$\begin{aligned} D_{prompt}(x, \mathbf{t}_y, u_x) = {} & w_y \cdot h_{img}\big(E_{img}(x)\big) \\ & + w_{unc} \cdot h_{img}\big(E_{img}(x)\big), \end{aligned} \tag{7}$$

where w_{unc} represents the unconditional identification weight. The adversarial training loss is defined as:

$$\begin{aligned} L_{prompt}^{real} &= \mathbb{E}_x[\log D_{prompt}(x, \mathbf{t}_y, u_x)], \\ L_{prompt}^{fake} &= \mathbb{E}_z[\log(1 - D_{prompt}(x_z, \mathbf{t}_y, u_z))]. \end{aligned} \tag{8}$$

By competing with the generator, the prompts $\mathbf{t}_y$ are encouraged to characterize the intra-class clusters, while at the same time D_{prompt} learns to identify real and synthesized instances, conditioned on $\mathbf{t}_y$. The gradient signals are back-propagated all the way through the generator, such that the knowledge encoded in the CLIP feature space is used to guide the generator. Learnable cluster prompts allow intra-class variation factors to be explored by interpolating the cluster label embeddings, which lead to a smooth embedding space and in turn improve the synthesis diversity.

3.4. Model Optimization

Considering that CLIP may be insensitive to the fine details of images, we also incorporate an additional discriminator D_{class} to distinguish real samples from the synthesized ones, conditioned on class label, and the corresponding loss is defined as follows:

$$\begin{aligned} L_{class}^{real} &= \mathbb{E}_x[\log D_{class}(x, y)], \\ L_{class}^{fake} &= \mathbb{E}_z[\log(1 - D_{class}(x_z, y))]. \end{aligned} \tag{9}$$

The predictor C is also jointly optimized in the adversarial training process. For both the labeled data and synthesized data, the ground truth labels are available, and C is required to infer the labels as accurately as possible. For the unlabeled data, C is encouraged to produce high-confidence predictions. The training loss of C is defined as follows:

$$\begin{aligned}L_{label}^{real} &= \mathbb{E}_x[\phi(C(x), y)] + \mathbb{E}_{x_{unl}}[-C(x_{unl}) \log C(x_{unl})], \\ L_{label}^{fake} &= \mathbb{E}_z[\phi(C(x_z), y)],\end{aligned} \tag{10}$$

where x_{unl} represents an unlabeled image, and $\phi(\cdot, \cdot)$ denotes the similarity measure between distributions, such as cross entropy. Based on the above, the overall optimization formulation of LCP-GAN can be expressed as follows:

$$\begin{aligned}\min_{C} \quad & L_{label}^{real} + L_{label}^{fake}, \\ \min_{G} \quad & L_{class}^{fake} + \lambda L_{prompt}^{fake} + L_{label}^{fake}, \\ \max_{D_{class}} \quad & L_{class}^{real} + L_{class}^{fake}, \\ \max_{D_{prompt}, \{\mathbf{t}_y\}} \quad & L_{prompt}^{real} + \lambda L_{prompt}^{fake},\end{aligned} \tag{11}$$

where λ is a weighting factor for balancing the two types of adversarial training terms. All the components of LCP-GAN are jointly optimized from scratch.

4. Experiments

Extensive experiments are performed to assess LCP-GAN on a variety of class-conditional image synthesis tasks. We first provide information about the benchmarks and experimental settings, which is followed by the investigation on the advantages of LCP-GAN over the base models. Furthermore, we provide insights via visualization and analysis of the adopted strategies, and further compare LCP-GAN with multiple leading semi-supervised GANs.

4.1. Experimental Settings

Datasets. CUB-200 [45] contains 6k/6k images from 200 bird categories for training/testing. Dogs-120 [27] is a dog image dataset, which consists of 12k training images and 9k test images from 120 dog categories. CelebA [34] is another popular dataset that contains about 202k face images from 10k celebrities, and we build CelebA-500 by selecting the largest 500 classes (10k training images and 5k test images), since the remaining classes contain too few images for GANs to properly learn class-conditional data distribution. As a more challenging benchmark, ImageNet [9] contains 1281k/50k training/validation images from 1k object categories.

Semi-supervised setting. To conduct a fair comparison with the competing semi-supervised GANs, we follow the setting that there are 2.8k/6k/5k/130k randomly sampled images that are labeled (14/50/10/130 images per class) for CUB-200/Dogs-120/CelebA-500/ImageNet-1k.

Table 1. Comparison of LCP-GAN and the base models.

	CUB-200			CelebA-500		
Method	FID↓	Intra-FID↓	RA↑	FID↓	Intra-FID↓	RA↑
Semi-BigGAN	24.4	112.83	64.52	25.02	144.61	55.67
LCP-GAN (B)	**13.61**	**88.03**	**93.24**	**15.38**	**136.72**	**81.60**
Semi-StyleGAN	18.17	90.09	56.77	16.23	144.54	35.71
LCP-GAN (S)	**10.78**	**71.39**	**81.72**	**14.29**	**131.78**	**64.00**

Base Models. We consider two state-of-the-art GAN architectures: BigGAN [2] and StyleGAN2-ADA [23], due to its widespread adoption and superior generation performance. We build two baseline models, called *Semi*-BigGAN and *Semi*-StyleGAN, through joint optimization with an ResNet-50-based image classifier [7], where the training scheme is the same as Triple-GAN [29]. Further, we make a number of necessary modifications to build our models: LCP-GAN (B) and LCP-GAN (S) accordingly.

Hyperparameter. The images of each class are projected into the feature space of a ResNet-50 [7] pre-trained on ImageNet, followed by k-means clustering. We empirically find that the intra-class variation can be well modeled by up to 3 clusters. The clusters with prototype cosine similarity $>$0.8 are merged. The number of intra-class clusters can be different for each class. This strategy provides the flexibility for the cluster setting. For each cluster, we set the number of learnable context vectors to 4, and randomly initialize them via a Gaussian distribution with mean 0 and standard deviation 0.02. LCP-GAN is updated using the Adam optimizer [28] with a learning rate of 0.0002. There are 500 training epochs, and the batch size is set to 16.

Evaluation metrics. We quantitatively evaluate the diversity and the degree of realism of the synthesized images in terms of Fréchet Inception Distance (**FID**) [20] and Inception Score (**IS**) [43]. For our class-conditional image generation task, it is important to measure the intra-class diversity and class-semantic accuracy. We compute the FID score separately for each class, and report the average score over all classes, which is referred to as **Intra-FID** [37]. We also report the Recognition Accuracy (**RA** %) that is computed from an independent classifier.

4.2. Improvement over Base Models

Diversity and degree of realism. We begin by quantitatively comparing LCP-GAN with the baseline models: *Semi*-BigGAN and *Semi*-StyleGAN. As shown in Table 1, LCP-GAN maintains a significant advantage over the baseline models in terms of FID and Intra-FID on each target dataset. In particular, the improvement over *Semi*-BigGAN reaches about 25 Intra-FID points on CUB-200. In Figure 3, we plot the relative improvements obtained by our approach for the 200 CUB classes over *Semi*-BigGAN, and find that the proposed approach brings positive effects on most of the classes where the reduction in FID reaches over 5 points on

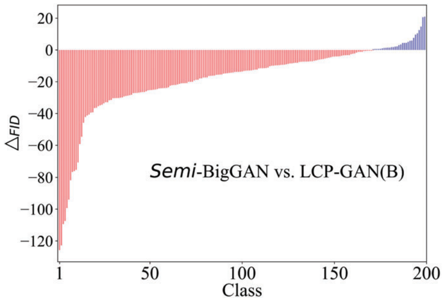

Figure 3. Per-class relative performance gain of the proposed model against the base model in terms of FID.

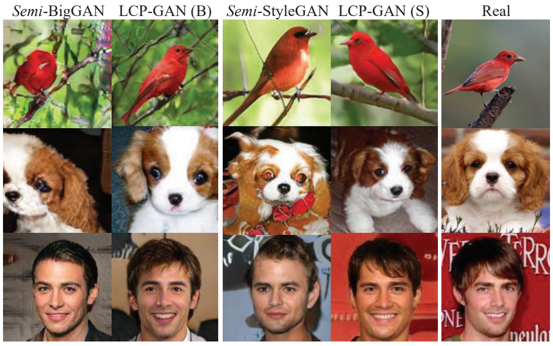

Figure 4. Synthesis results of LCP-GAN and the base models.

146 classes. Although *Semi*-StyleGAN achieves a low FID score of 16.23 on CelebA-500, LCP-GAN (S) can still lead to a performance gain in this case. We consider the correct modeling of intra-class data structure as an important step in class-conditional generative learning.

Class semantics. For semi-supervised image generation, it is challenging to learn a generator to capture precise class semantics. For both BigGAN and StyleGAN2-ADA architectures, LCP-GAN is able to achieve noticeably higher RA scores than those of the corresponding base models on each dataset. On CelebA-500, the improvement reaches above 25 percentage points. The capability to achieve the improvement suggests that integrating the supervision from a pre-trained language-vision model is a significant step towards conditional synthesis with precise class semantics. In Figure 4, we visualize a number of representative synthesized images to demonstrate the consistent improvement in class semantics across the datasets.

4.3. Intra-class Variation Factors

Although there is no prior knowledge about the intra-class clusters, we believe that the learnable cluster-specific prompts benefit from CLIP and are important for modeling intra-class data distribution. As a result, LCP-GAN is able to associate the cluster label embeddings with the underlying visual concepts reflected by the clusters. We perform linear interpolation to construct an interpolation path between the paired cluster label embeddings, and a set of class-specific images can be synthesized from the same latent vector, conditioned on the interpolated embeddings. As shown in Figure 5, the synthesis results have realistic appearances and hold consistent class semantics. We consider that LCP-GAN is capable of modeling the intra-class variation, since the synthesized images correspond to a smooth transformation along the interpolation path.

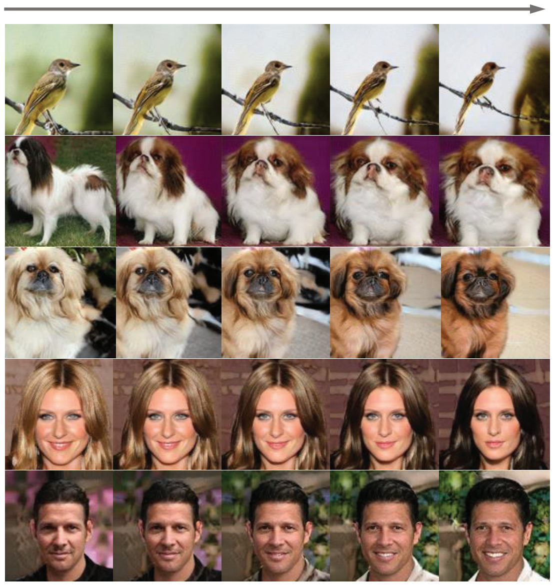

Figure 5. Interpolation results of LCP-GAN by linearly combining the paired cluster label embeddings.

Table 2. Results of LCP-GAN and the ablative models.

	CUB-200			Dogs-120		
Method	FID↓	Intra-FID↓	RA↑	FID↓	Intra-FID↓	RA↑
Semi-BigGAN	24.4	112.83	64.52	31.08	97.52	79.68
Semi-BigGAN+L_{CLIP}	17.37	104.82	88.58	18.02	77.86	89.41
Semi-BigGAN+*Clusters*	18.32	108.38	82.83	18.39	82.83	84.38
LCP-GAN (B) w/o LP	15.03	92.60	91.22	17.89	74.75	91.42
LCP-GAN (B)	**13.61**	**88.03**	**93.24**	**15.03**	**72.06**	**92.75**

4.4. Analysis of the Main Components

To better demonstrate the effectiveness of our model design, we perform a series of experiments in Table 2.

Is the supervision from CLIP helpful? We first build a variant '*Semi*-BigGAN+L_{CLIP}', in which the class prompt template 'A photo of [`class_name`]' is used to depict the content of the images from the same class. An additional training goal of the generator is to maximize the cosine similarity between the synthesized image and corresponding prompt $\mathbf{t}_y$ as follows:

$$L_{CLIP} = \mathbb{E}_z \left[\log \frac{\exp(\cos(E_{txt}(\mathbf{t}_y), E_{img}(x_z))/\delta)}{\sum_y \exp(\cos(E_{txt}(\mathbf{t}_y), E_{img}(x_z))/\delta)} \right].$$

Table 3. Comparison of LCP-GAN and competing semi-supervised GANs on various benchmark datasets.

	CUB-200			CelebA-500				Dogs-120		
Method	IS↑	FID↓	Intra-FID↓	IS↑	FID↓	Intra-FID↓	IDS↑	IS↑	FID↓	Intra-FID↓
Real data	*5.96±0.05*	-	-	*3.77±0.08*	-	-	-	*46.54±1.01*	-	-
Triple-GAN [29]	3.91±0.05	140.94	-	-	-	-	-	-	-	-
ETGAN [48]	3.95±0.06	133.57	-	-	-	-	-	-	-	-
Δ-GAN [15]	4.22±0.03	96.42	-	-	-	-	-	-	-	-
R^3-CGAN [32]	4.46±0.08	88.62	-	-	-	-	-	-	-	-
SReGAN [4]	4.84±0.04	19.13	-	-	-	-	-	-	-	-
SSC-GAN [3]	4.68±0.04	20.03	91.75	2.70±0.02	49.41	159.69	0.42	33.94±0.22	27.64	94.70
SphericGAN [6]	5.03±0.05	18.87	91.60	2.93±0.03	45.69	171.90	0.42	33.18±0.30	27.19	95.45
MED-GAN [33]	5.54±0.10	16.90	91.41	-	-	-	-	-	-	-
ReACGAN [22]	**5.81**±0.07	32.29	145.45	2.51±0.02	29.02	154.36	0.35	28.30±0.73	20.48	91.59
LCP-GAN (B)	5.04±0.04	13.61	88.03	3.05±0.03	15.38	136.72	**0.44**	**39.92**±0.62	15.03	72.06
LCP-GAN (S)	4.96±0.07	**10.78**	**71.39**	**3.27**±0.03	**14.29**	**131.78**	0.43	29.57±0.05	**8.71**	**56.40**

Compared with *Semi*-BigGAN, the variant exploits the semantic information of label texts and improves the performance with about 8 Intra-FID/24 RA points on CUB-200 (20/10 points on Dogs-120), suggesting that the supervision from CLIP is helpful to learn more precise class semantics.

Are the intra-class clusters important? We extend *Semi*-BigGAN to model intra-class clusters. Specifically, we modify the generator and discriminator to be conditioned on the combination of learnable cluster label embeddings, and the resulting model is referred to as '*Semi*-BigGAN+*Clusters*'. An improvement (about 4/15 Intra-FID points on CUB-200/Dogs-120) over *Semi*-BigGAN can be observed. The result confirms that synthesizing images with multiple label embeddings can better match class-specific data distribution. Note that there is still a substantial performance gap between the variant and LCP-GAN (B).

Are the learnt prompts meaningful? Due to the lack of prior knowledge on intra-class clusters, we initialize cluster-specific context vectors and freeze them during training, and the variant is referred to as 'LCP-GAN (B) w/o LP' (Learnable Prompts, LP). We find that the variant fails to attain a strong performance as that of LCP-GAN (B). Specifically, fixing the context words leads to a negative influence on the intra-class diversity, and the performance drop reaches about 5 Intra-FID points on CUB-200. The result confirms the effectiveness of customizing the prompt representation to each cluster. We also search for the words that are closest to the learnt prompts based on the cosine similarity in the CLIP feature space. The resulting Top-5 words are listed in Figure 6, and one can find that most of them are somewhat relevant to the synthesized images.

4.5. Comparison to State-of-the-Arts

To demonstrate the advantages of LCP-GAN with respect to state-of-the-art semi-supervised GANs, we perform extensive comparison on diverse image generation tasks. The results of the competing methods are detailed in Table 3. The results suggest that the proposed approach outperforms all the competing methods in terms of both FID and Intra-FID across all datasets. In particular, LCP-GAN (B) achieves FID 13.61 on CUB-200, which appears to be a new state-of-the-art result, and outperforms the second best method (MED-GAN, FID 16.90) by about 3 points. On Dogs-120, the advantage of LCP-GAN (B) is also noticeable, and the Intra-FID score is 72.06, which is considerably lower than that of ReACGAN by about 20 points. On CelebA-500, we measure the IDentity similarity (IDS) between class-specific real and synthesized face images via CosFace [46]. Compared to SphericGAN and ReACGAN, we achieve a higher IDS score. We believe that this is primarily due to a more accurate modeling of the underlying intra-class data structure.

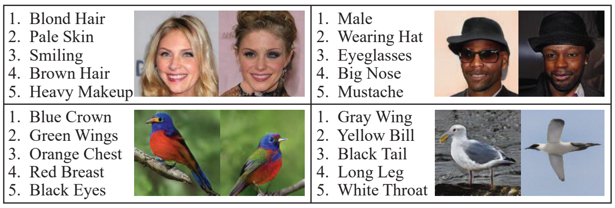

Figure 6. The class-specific nearest words and synthesized images.

4.6. Further Analysis

Amount of labeled data. We are also interested to know whether LCP-GAN has stable performance when the amount of labeled data decreases. To compare with the existing methods, the experiments are conducted on CUB-200 and FaceScrub-100 [38], and the number of labeled images per class is limited in the ranges: $\{3, 6, 9, 12, 14, 28\}$ and $\{13, 26, 39, 52, 65, 130\}$, respectively. Figure 7 shows that our design significantly enhances the model robustness to the amount of labels. In contrast, the results of the base model and competing methods are unsatisfactory, especially for the case of $<$9 labeled images per class. It is surprising to find that LCP-GAN achieves a FID score of 15.15 when using only 26 labels per class, while the FID score of the fully supervised BigGAN is 15.51 on FaceScrub-100. We

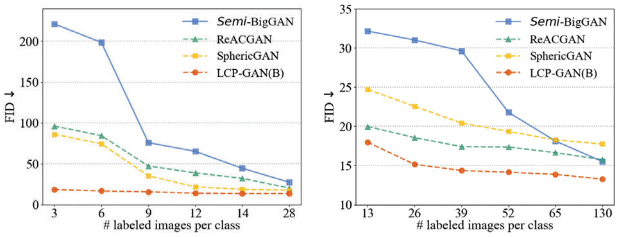

Figure 7. Comparison between LCP-GAN and the competing methods on CUB-200 (*left*) and FaceScrub-100 (*right*).

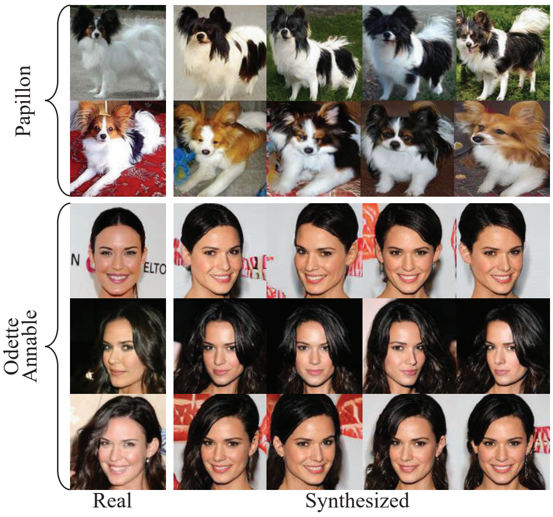

Figure 8. The cluster-specific images synthesized by LCP-GAN.

conjecture that CLIP provides wider supervision for capturing visual concepts, compared to the strong base model.

Semantically meaningful clusters. LCP-GAN is conditioned on the combination of intra-class cluster label embeddings. By assigning a one-hot coefficient vector, our model is able to synthesize cluster-specific images. In Figure 8, we visualize the synthesized images for a number of semantically meaningful clusters, and observe that the clusters are associated with pose, appearance, viewpoint, and so on. Note that the intra-class clustering does not guarantee that the resulting clusters are semantically meaningful, and they may not necessarily correspond to the subordinate categories. In general, LCP-GAN is independent of the underlying clustering algorithm used.

4.7. Results on ImageNet

In this example, we focus on a more challenging dataset: ImageNet. In Figure 9, we plot the FID scores of LCP-GAN and *Semi*-BigGAN in the training process. One can find that LCP-GAN converges to a lower FID than the baseline, and matches its best result up to 2 times faster. More results are summarized in Table 4. LCP-GAN achieves significant performance improvement against *Semi*-BigGAN, and the improvement of about 44 Intra-FID points on ImageNet is noteworthy. DiffAug [53] and LCP-GAN aim to facilitate generative learning in different settings. We attempt to build a strong base model by adopting DiffAug. LCP-GAN can even outperform *Semi*-BigGAN+DiffAug [53]. We believe that the results serve as strong evidence of the LCP-GAN's capability of complex class-conditional image synthesis.

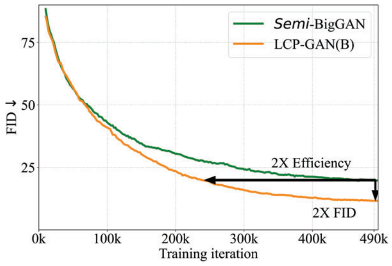

Figure 9. Convergence properties of LCP-GAN and the base model on ImageNet.

Table 4. Comparison of LCP-GAN and *Semi*-BigGANs on ImageNet.

Method	IS↑	FID↓	Intra-FID↓	RA↑
Semi-BigGAN	44.27	21.45	185.94	53.59
Semi-BigGAN+DiffAug	73.32	12.95	160.27	58.02
LCP-GAN (B)	**94.91**	**10.94**	**142.15**	**82.41**

5. Conclusion

We have investigated the possibility of modeling intra-class data structure to facilitate semi-supervised generative learning. We extend a semi-supervised GAN framework to learn from intra-class clusters, and enable class-specific image synthesis to be conditioned on the combination of cluster label embeddings. This design enhances our framework to model large intra-class variance. Due to the lack of prior knowledge about the clusters, we leverage the language-vision pre-training and jointly learn cluster-specific prompts through prompt-conditional adversarial training. The proposed model is able to discover a wide range of semantically meaningful intra-class variation factors and achieve superior performance on multiple semi-supervised image synthesis tasks. We hope the insights presented can be useful for facilitating semi-supervised generative learning.

Acknowledgments

This work was supported in part by the China Scholarship Council, in part by the National Natural Science Foundation of China (Project No. 62072189), in part by the Research Grants Council of the Hong Kong Special Administration Region (Project No. CityU 11206622), and in part by the Natural Science Foundation of Guangdong Province (Project No. 2022A1515011160).

References

[1] Martin Arjovsky, Soumith Chintala, and Leon Bottou. Wasserstein generative adversarial networks. In *Proc. International Conference on Machine Learning*, 2017. 2

[2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natual image synthesis. In *Proc. International Conference on Learning Representation*, 2019. 1, 2, 3, 5

[3] Tianyi Chen, Yi Liu, Yunfei Zhang, Si Wu, Yong Xu, Feng Liangbing, and Hau San Wong. Semi-supervised single-stage controllable GANs for conditional fine-grained image generation. In *Proc. International Conference on Computer Vision*, 2021. 7

[4] Tianyi Chen, Si Wu, Xuhui Yang, Yong Xu, and Hau-San Wong. Semantic regularized class-conditional GANs for semi-supervised fine-grained image synthesis. *IEEE Transactions on Multimedia (Early Access)*, 2021. 7

[5] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised GANs via auxiliary rotation loss. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 3

[6] Tianyi Chen, Yunfei Zhang, Xiaoyang Huo, Si Wu, Yong Xu, and Hau San Wong. SphericGAN: semi-supervised hyper-spherical generative adversarial networks for fine-grained image synthesis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 3, 7

[7] Yue Chen, Yalong Bai, Wei Zhang, and Tao Mei. Destruction and construction learning for fine-grained image recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 5157–5166, 2019. 5

[8] Z. Dai, Z. Yang, F. Yang, W. Cohen, and R. Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Proc. Neural Information Processing Systems*, 2017. 3

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: a large-scale hierarchical image database. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 4, 5

[10] Zhijie Deng, Hao Zhang, Xiaodan Liang, Luona Yang, Shizhen Xu, Jun Zhu, and Eric P. Xing. Structured generative adversarial networks. In *Proc. Neural Information Processing Systems*, 2017. 3

[11] Emily Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep generative image models using a Laplacian pyramid of adversarial networks. In *Proc. Neural Information Processing Systems*, 2015. 2

[12] Jinhao Dong and Tong Lin. MarginGAN: adversarial training in semi-supervised learning. In *Proc. Neural Information Processing Systems*, 2019. 3

[13] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. In *Proc. International Conference on Learning Representation*, 2017. 3

[14] Hamid Eghbal-zadeh, Werner Zelinger, and Gerhard Widmer. Mixture density generative adversarial networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[15] Zhe Gan, Liqun Chen, Weiyao Wang, Yunchen Pu, Yizhe Zhang, Hao Liu, Chunyuan Li, and Lawrence Carin. Triangle generative adversarial networks. In *Proc. Neural Information Processing Systems*, 2017. 1, 3, 7

[16] Arnab Ghosh, Viveka Kulharia, Vinay Namboodiri, Philip H.S. Torr, and Puneet K. Dokania. Multi-agent diverse generative adversarial networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[17] Mingming Gong, Yanwu Xu, Chunyuan Li, Kun Zhang, and Kayhan Batmanghelich. Twin auxiliary classifiers GAN. In *Proc. Neural Information Processing Systems*, 2019. 3

[18] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proc. Neural Information Processing Systems*, 2014. 1

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 4

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, and Bernhard Nessler. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proc. Neural Information Processing Systems*, 2017. 5

[21] Quan Hoang, Tu Dinh Nguyen, Trung Le, and Dinh Phung. MGAN: training generative adversarial nets with multiple generators. In *Proc. International Conference on Learning Representation*, 2018. 2

[22] Minguk Kang, Woohyeon Shim, Minsu Cho, and Jaesik Park. Rebooting ACGAN: auxiliary classifier GANs with stable training. In *Proc. Neural Information Processing Systems*, 2021. 3, 7

[23] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Proc. Neural Information Processing Systems*, 2020. 2, 3, 5

[24] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[25] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2

[26] Ilya Kavalerov, Wojciech Czaja, and Rama Chellappa. c-GANs with multi-hinge loss. *arXiv:1912.04216*, 2019. 3

[27] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization*, 2011. 5

[28] Diederik P. Kingma and Jimmy Lei Ba. Adam: a method for stochastic optimization. In *Proc. International Conference on Learning Representation*, 2015. 5

[29] Chongxuan Li, Kun Xu, Jun Zhu, and Bo Zhang. Triple generative adversarial nets. In *Proc. Neural Information Processing Systems*, 2017. 1, 3, 5, 7

[30] Zeyu Li, Cheng Deng, Erkun Yang, and Dacheng Tao. Staged sketch-to-image synthesis via semi-supervised gen-

erative adversarial networks. *IEEE Transactions on Multimedia*, 23:2694–2705, 2021. 3

[31] Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, and Antonio Torralba. Diverse image generation via self-conditioned GANs. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 3

[32] Yi Liu, Guangchang Deng, Xiangping Zeng, Si Wu, Zhiwen Yu, and Hau-San Wong. Regularizing discriminative capability of CGANs for semi-supervised generative learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 3, 7

[33] Yi Liu, Xiaoyang Huo, Tianyi Chen, Xiangping Zeng, Si Wu, Zhiwen Yu, and Hau-San Wong. Mask-embedded discriminator with region-based semantic regularization for semi-supervised class-conditional image synthesis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 1, 3, 7

[34] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proc. International Conference on Computer Vision*, 2015. 5

[35] M. Mirza and S. Osindero. Conditional generative adversarial nets. In *arXiv:1411.1784*, 2014. 3

[36] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2018. 2

[37] Takeru Miyato and Masanori Koyama. cGANs with projection discriminator. In *Proc. International Conference on Learning Representation*, 2018. 3, 5

[38] Hong-Wei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *Proc. IEEE International Conference on Image Processing*, 2014. 7

[39] Mehdi Noroozi. Self-labeled conditional GANs. *arXiv:2012.02162*, 2020. 2

[40] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier GANs. In *Proc. International Conference on Machine Learning*, 2017. 3

[41] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2016. 2

[42] Alec Redford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proc. International Conference on Machine Learning*, 2021. 2

[43] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Proc. Neural Information Processing Systems*, 2016. 2, 3, 5

[44] Jost Tobias Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2016. 1, 3

[45] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. In *Technical Report CNS-TR-2011-001, California Institute of Technology*, 2011. 5

[46] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. CosFace: large margin Cosine loss for deep face recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 7

[47] Xiang Wei, Boqing Gong, Zixia Liu, Wei Lu, and Liqiang Wang. Improving the improved training of Wasserstein GANs: a consistency term and its dual effect. In *Proc. International Conference on Learning Representation*, 2018. 2

[48] Si Wu, Guangchang Deng, Jichang Li, Rui Li, Zhiwen Yu, and Hau-San Wong. Enhancing TripleGAN for semi-supervised conditional instance synthesis and classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 3, 7

[49] S. Yun, D. Han, S. Oh, S. Chun, J. Choe, and Y. Yoo. CutMix: regularization strategy to train strong classifiers with localizable features. In *Proc. International Conference on Computer Vision*, 2019. 3

[50] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *Proc. International Conference on Machine Learning*, 2019. 2

[51] Han Zhang, Zizhao Zhang, Augustus Odena, and Honglak Lee. Consistency regularization for generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2019. 2

[52] Miaoyun Zhao, Yulai Cong, and Lawrence Carin. On leveraging pretrained GANs for generation with limited data. In *Proc. International Conference on Machine Learning*, 2020. 2

[53] Shenyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient GAN training. In *Proc. Neural Information Processing Systems*, 2020. 3, 8

[54] Zhengli Zhao, Sammer Singh, Honglak Lee, Zizhao Zhang, Augustus Odena, and Han Zhang. Improved consistency regularization for GANs. *arXiv:2002.04724*, 2020. 3

[55] Zhengli Zhao, Zizhao Zhang, Ting Chen, Sameer Singh, and Han Zhang. Image augmentations for GAN training. *arXiv:2006.02595*, 2020. 3

[56] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130:2337–2348, 2022. 2