DIFFUSED INSTANCE CONDITIONED GAN

Anonymous authors
Paper under double-blind review

ABSTRACT

Recently, numerous data partitioning methods for generative adversarial networks has been developed for better distribution coverage on complex distribution. Most of these approaches aims to build fine-grained overlapping clusters in data manifold and condition both generator and discriminator with compressed representation about cluster. Although giving larger size of condition can be more informative, existing algorithms only utilize low dimension vector as condition due to dependency on clustering algorithm and unsupervised / self-supervised learning methods. In this work, we take a step towards using richer representation for cluster by utilizing diffusion based Gaussian mixture. Our analysis shows that we can derive continuous representation of cluster with Gaussian mixture when noise scale is given. Moreover, unlike other counterparts, we do not need excessive computation for acquiring clustered representation. Experiments on multiple datasets show that our model produces better results compared to recent GAN models.

1 INTRODUCTION

Recently, generative adversarial networks (GAN)(Goodfellow et al., 2020) shows monumental success in unconditional image generation(Sauer et al., 2021; Karras et al., 2021). While they are showing qualitative results, training GAN accompanies difficulties in optimization to avoid mode collapse. Samples from collapsed generators would not obtain a full distribution coverage nor produce realistic images. Previous literatures have endeavored to solve this problem in various aspects (Gulrajani et al., 2017; Mescheder et al., 2018; Zhao et al., 2020). Among them, Class conditional GANs(Mirza & Osindero, 2014; Brock et al., 2018) shows such difficulties can be mitigated by learning partitioned data distribution. By exploiting given attributes per instance as a partition information, they could guide the generator to only model subspace of target distribution. Despite their superior results than unconditional GANs, they are constrained to existence of label in datasets, which may not exists or expensive to attain.

To alleviate requirement for labeled datasets, recent literatures proposed unsupervised / self-supervised learning based data partitioning methods(Eghbal-zadeh et al., 2019; Liu et al., 2020; Casanova et al., 2021; Armandpour et al., 2021). Most of them utilize either k-means clustering in feature space or contrastive loss to form fine-grained overlapping cluster. Such well-designed cluster enables generating high-quality samples without mode collapse. Although previous methods gain plausible data partition with k-means and contrastive loss, they can't extend themselves to utilize high dimensional condition. For those with clustering, high dimensional space will cause distant instance to belonging to same cluster. The others with contrastive loss suffers when dimension increase due to excessive costs at computing similarity on the fly during training.

Recently, diffusion-based generative models (DM)(Ho et al., 2020; Sohl-Dickstein et al., 2015) are showing state-of-the-art performance at unconditional image generation. They are a parameterized Markov chain trained with a given fixed posterior called forward process which gradually adds Gaussian noise to data according to time. Most interesting part of DM is **JH: TBD**

Inspired by aforementioned property of diffusion models, we introduce a new data partitioning approach for GAN, called diffused instance-conditioned GAN (DIC-GAN). DIC-GAN model a mixture of local densities with diffusion based Gaussian mixture. Like previous works using partition guidance, DIC-GAN trained to cover distribution of local neighborhood of data point. Such local neighborhood distribution will be determined by sample probability within a diffusion based Gaus-

(a) DIC-GAN samples (Art-painting) (b) DIC-GAN samples (Flowers)

sian mixture. Both generator and discriminator will be provided a noised representation extracted from the data point and trained by using that data point as a target real.

JH: Have to change Unlike previous works, DIC-GAN can model a mixture of local data densities with high dimension condition by utilizing diffusion based Gaussian mixture. Moreover, using this diffusion based Gaussian mixture gives us opportunity to make our latent space friendly to diffusion model. we use this advantage and train latent diffusion model which modeling target distribution as features from encoder. Such methods enable unconditional image generation of our model. We validate our approach on unconditional image generation task. Additionally, we show our result on few-shot dataset which have discrete distribution and hard to train. Overall, we make the following contributions:

- We propose Diffused instance-conditioned GAN (DIC-GAN), which utilize noised instance feature as a condition for partitioned dataset.
- We validated our approach on unlabeled image generation tasks, showing consistent improvements over baseline.

2 DIFFUSED INSTANCE-CONDITIONED GAN

Our motivation for DIC-GAN is to utilize diffusion based Gaussian mixture for representing clusters of datasets. Given an unlabeled dataset $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^N$ and noise scale σ, each cluster is defined by a noised representation $\mathbf{h}$, which sampled by noising a data point $\mathbf{x}_i$ in feature space. The data distribution $p(\mathbf{x})$ can be modeled by a mixture of conditional distributions $p(\mathbf{x}|\mathbf{h}, \sigma)$ where noised representation $\mathbf{h}$ is sampled with probability of $p(\mathbf{h}|\sigma)$.

Given a feature extractor f parameterized by ψ and noise scale σ, we sample the noised representation by $\mathbf{h} = \sqrt{1-\sigma^2} f(\mathbf{x}_i; \psi) + \sigma\epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$ is standard normal noise. The sampling probability of $p(\mathbf{h}|\sigma)$ can be expressed as follows.

$$p(\mathbf{h}|\sigma) = \sum_{i=1}^{N} p(\mathbf{h}|\mathbf{x}_i, \sigma) p(\mathbf{x}_i) \tag{1}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \mathcal{N}\big(\mathbf{h}; \sqrt{1-\sigma^2} f(\mathbf{x}_i; \psi), \sigma^2 I\big). \tag{2}$$

From Equation 2, we can derive $p(\mathbf{x}_i|\mathbf{h}, \sigma)$ by applying bayes rule as follows.

$$p(\mathbf{x}_i|\mathbf{h}, \sigma) = \frac{p(\mathbf{x}_i, \mathbf{h}|\sigma)}{p(\mathbf{h}|\sigma)} \tag{3}$$

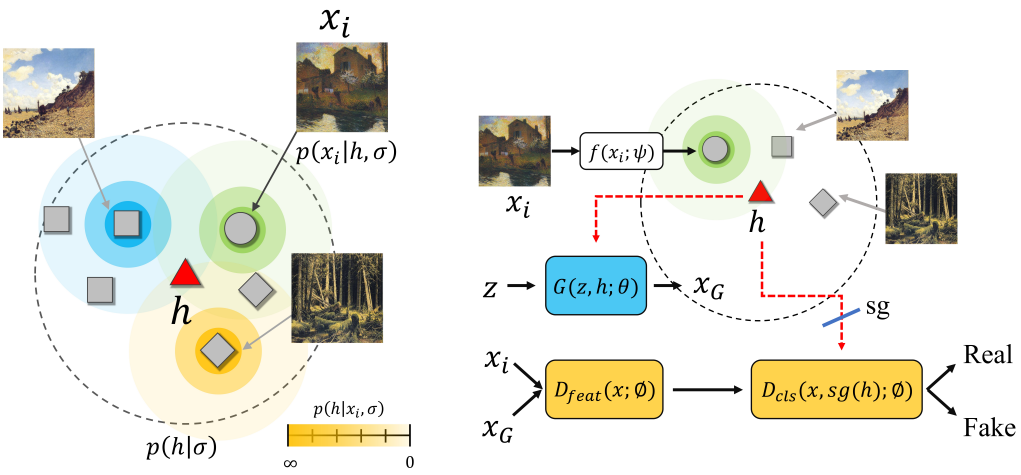
$$= \frac{\mathcal{N}\left(\mathbf{h}; \sqrt{1-\sigma^2} f_\psi(\mathbf{x}_i), \sigma^2 I\right)}{\sum_{j=1}^{N} \mathcal{N}\left(\mathbf{h}; \sqrt{1-\sigma^2} f_\psi(\mathbf{x}_j), \sigma^2 I\right)}. \tag{4}$$

Equation 4 shows which data point will be belonged to the cluster represented with $\mathbf{h}$ and its sampling probability within the cluster. Figure 2a describes a sample $\mathbf{x}_i$ and noised representation $\mathbf{h}$.

Our GAN consists of a conditional generator $G(\mathbf{z}, \mathbf{h}; \theta)$, conditional discriminator $D(\mathbf{x}, \mathbf{h}; \phi)$, and feature extractor $f(\mathbf{x}; \psi)$ where θ, ϕ, ψ are parameters for generator, discriminator, and feature extractor. We denote the internal discriminator feature layers as D_{feat} and classification layers as D_{cls} so $D = D_{cls} \circ D_{feat}$. The generator $G(\mathbf{z}, \mathbf{h}; \theta)$ trained to generate sample from a partitioned distribution $p(\mathbf{x}|\mathbf{h}, \sigma)$ given a unit Gaussian prior $\mathbf{z} \sim \mathcal{N}(0, I)$ and noised representation $\mathbf{h}$. To get better representation describing data distribution, we jointly train our feature extractor $f(\cdot; \psi)$ with our generator. The generator, discriminator, and feature extractor are trained to optimize following adversarial objective.

$$\min_{G,f} \max_{D} \mathbb{E}_{\mathbf{x}_i \sim p(\mathbf{x}), \mathbf{h} \sim p(\mathbf{h}|\mathbf{x}_i, \sigma)} [\log D(\mathbf{x}_i, \mathrm{sg}(\mathbf{h}))] + \mathbb{E}_{\mathbf{x}_i \sim p(\mathbf{x}), \mathbf{z} \sim p(\mathbf{z}), \mathbf{h} \sim p(\mathbf{h}|\mathbf{x}_i, \sigma)} [\log(1 - D(G(\mathbf{z}, \mathbf{h}), \mathrm{sg}(\mathbf{h})))]. \tag{5}$$

Note that we train our feature extractor only through generator's gradient by applying stop-gradient operation (sg) when noised representation $\mathbf{h}$ given to discriminator. Also, although Equation 5 is expressed only with $\mathbf{h}$ sampled from $p(\mathbf{h}|\mathbf{x}_i, \sigma)$, such $\mathbf{h}$ can be sampled from different $\mathbf{x}_i$ due to Equation 4. Figure 2b illustrate how generator and discriminator are trained with our objective.



(a) Instance $\mathbf{x}_i$ and noised representation $\mathbf{h}$

(b) Workflow of DIC-GAN

Figure 2: Overview of DIC-GAN. (a) Our data distribution is partitioned based on diffusion based Gaussian mixture. Given noised representation $\mathbf{h}$, generator trained to generate sample following $p(\mathbf{x}_i|\mathbf{h}, \sigma)$. Other two samples which have probability to sample $\mathbf{h}$ are depicted in the figure. (b) Conditioned on noised representation $\mathbf{h}$ and sampled noise z, generator sample $\mathbf{x}_G$. Fake sample $\mathbf{x}_G$ and real sample $\mathbf{x}_i$ are used to train the discriminator about distribution $p(\mathbf{x}_i|\mathbf{h}, \sigma)$. Following our baseline projected-GAN(Sauer et al., 2021), we use the pretrained network as D_{feat}. Since our D_{feat} is fixed, we use $\mathbf{h}$ as condition for discriminator by concatenating them to input of D_{cls}. To train our feature extractor $f(\cdot, \psi)$ with generator's gradient only, we apply stop-gradient operation when they are given to discriminator's condition.

Introducing noised condition $\mathbf{h}$ to our generator cause two problems. The first problem is artifacts introduced by additive noise. We utilized noise scale aware condition modulation to solve this

problem. The second problem is unfeasibility of unconditional generation due to condition. During training, our generator can sample with condition from training set. However, by conditioning the generator on $\mathbf{h}$, DIC-GAN are no longer available for unconditional generation. To generate samples without condition from training set, we need an additional module to sample $\mathbf{h}$ from latent distribution. Here, we introduce a latent diffusion model to the latent distribution of $f(\mathbf{x}_i; \psi), \mathbf{x}_i \sim p(\mathbf{x})$. In the following section, we would like to describe about these problems more detail and present methods to solve these problems.

2.1 Noise scale aware condition modulation.

Since we are using additive noise to sample $\mathbf{h}$, naive usage of condition in the generator can cause serious artifact to the generated sample. As shown in Figure **??**, additive noise manifest themselves as stochastic artifact in generated samples. To mitigate this problem, we designed noise scale aware condition modulation and apply them where the generator receives the condition. Our generator provided with noise scale σ and modulate given condition $\mathbf{h}$ proportional to given noise scale as follows.

$$\hat{\mathbf{h}} = (\mathbf{h} - \sigma G_{mod}(\mathbf{h}, z))/(\sqrt{1-\sigma^2}) \tag{6}$$

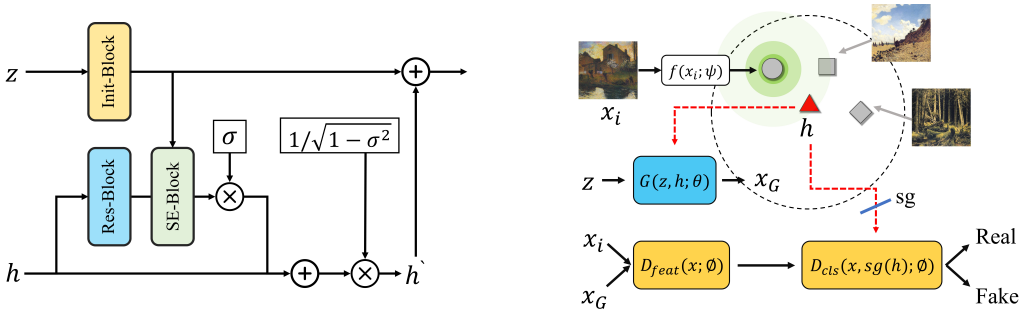
This modulation was inspired by denoising process in diffusion model. As the condition $\mathbf{h}$ is noised with additive Gaussian noise with scale σ and the original signal has been reduced with scale $\sqrt{1-\sigma^2}$, we recover the original signal by applying the above modulation. Although we didn't train our generator directly to denoise condition $\mathbf{h}$, our new design removes characteristic artifacts introduced by additive noise.

2.2 Unconditional image generation with latent diffusion model.

When training DIC-GAN, we utilize data points in training set to generate noised condition $\mathbf{h}$. This formulation requires us additional mechanism to sample noised condition $\mathbf{h}$ at inference time for unconditional image generation. We use latent diffusion model $S(f(\mathbf{x}_i; \psi)_t, t; \omega)$ to sample from our latent distribution $f(\mathbf{x}_i; \psi), \mathbf{x}_i \sim p(\mathbf{x})$ where $f(\mathbf{x}_i; \psi)_t = \sqrt{\alpha_t} f(\mathbf{x}_i; \psi)_t + \sqrt{1-\alpha_t}\epsilon$ and α_t is variance scheduling used in diffusion model and ω is parameter for diffusion model. Training the latent diffusion model is done by optimizing follow objective:

$$\min_{\omega} \sum_{t=1}^{T} \mathbb{E}_{\mathbf{x}_i \sim p(\mathbf{x}), \epsilon_t \sim \mathcal{N}(0,I)} [\| S(f(\mathbf{x}_i; \psi)_t, t; \omega) - \epsilon_t \|_2]. \tag{7}$$

As we use spatial dimension in latent space, we can employ our works to well established diffusion model architecture such as U-Net structure without any modification. We empirically found that naive employment of diffusion model in latent space usually works well when channel dimension is not too large. Details about hyperparameter for training latent diffusion model are provided in 3



(a) Noise scale aware modulation

(b) Instance $\mathbf{x}_i$ and noised representation $\mathbf{h}$

Figure 3: Detailed parts of DIC-GAN.

	FID	*Imgs*	FID	*Imgs*	FID	*Imgs*
256^2	**Art Painting**		**Flowers**		**Pokemon**	
StyleGAN2-ADA	43.07	3.2 M	21.66	3.8 M	40.38	3.4 M
FastGAN	44.02	0.7 M	26.23	0.8 M	81.86	2.5 M
Projected GAN	27.96	0.8 M	13.86	1.8 M	**26.36**	0.8 M
DIC-GAN	**26.71**	3.2 M	**13.34**	3.4 M	27.25	2.8 M
DIC-GAN$_{ref}$	25.01	-	12.52	-	26.13	-

Table 2: **Quantitative Results.**

3 Experimental evaluation

In this section, we conduct a comprehensive analysis demonstrating the advantages of DIC-GAN with respect to state-of-the-art models. Our experiments are structured into three sections: evaluation of sample generated from noised condition from reference image (3.2), and comparisons of unconditional image generation with latent diffusion model (3.1) on benchmark datasets. We evaluate our model in unlabeled image genertion on art paintings from WikiArt (1000 images; wikiart.org), Oxford Flowers (1360 images) and Pokemon (833 images; pokemon.com). **JH: refs** We only evaluate on resolution 256^2.

Evaluation Protocol. We measure image quality using the Fréchet Inception Distance (FID). We report the FID between 50k generated and all real images. For baseline, we report results from **JH: ref** . we also report other metrics that are less benchmarked in GAN literature: KID [3], SwAV-FID [39], precision and recall [51]. **JH: refs**

Baseline For our conditional generator and conditional discriminator, we use Projected-GAN as baselines. Projected-GAN is one of the strongest generative model on most datasets in terms of sample quality, mode coverage and training speed. We implement these baselines and our DIC-GANs within the codebase provided by the authors of Projected gan. Following the implementation in Projected-GAN, we ran differentiable data-augmetnation for our experiment. For all datasets, we perform data amplification through x-flips. Unlike projected GAN, DIC-GANs use the conditional generator and discriminator architecture. Although we are using additional module compared to baseline, we adjust training hyperparameters to be similar with baseline by changing channel dimension per resolution. We use same learning rate and batch size for all experiments.

For latent diffusion model we use DDIM as baseline. DDIM is an efficient class of iterative implicit probabilistic models with the same training procedure as DDPMs. They enable faster sampling than DDPM without loss of quality of sample by using non-Markovian diffusion process. We changed several hyperparameters for architecture to fit our latent space. Just like original DDIM implementation, we utilize 1000 training steps for training.

3.1 Unconditional image generation with latent diffusion model

3.2 Sampling from reference image

In this section we would like to show several properties of DIC-GAN given reference image and demonstrate new capabilities such as cluster wise sampling and per cluster sampling. Due to our design in modulation with Z and noised condition h, we can control per cluster sampling. More over modulation from Z has different size of effect per noise scale σ.

To demonstrate

Table 1: Results for Imagenet in unlabeled setting.

Method	**Res.**	**↓FID**	**↑IS**
Self-sup. GAN	64	19.2	16.5
Uncond. BigGAN†	64	16.9 ± 0.0	14.6 ± 0.1
IC-GAN	64	10.4 ± 0.1	21.9 ± 0.1
IC-GAN + DA	64	9.2 ± 0.0	23.5 ± 0.1
DIC-GAN + DA	64	9.2 ± 0.0	23.5 ± 0.1
DIC-GAN$_{ref}$ + DA	64	6.5 ± 0.0	23.5 ± 0.1

4 Related Work

Clustered GAN training. Clustering data distribution was largely applied for improving image generation quality and diversity. These works can be classified in to two of kinds by how they implemented clustering in their generation process. The first one use clustering techinques to data manifold within feature space or data space. These approaches use pretrained feature network to gain representation for instance and cluster them by applying k-means. Information about cluster are given to either generator or discriminator to model fine-grained distribution. Instance condition GAN belongs to these. The other ones use mixture models in their design, intrisically training partitioned distribution within their model. These approaches use multi agent for generator or latent space for building such mixture distribution in latent space. Moreover, by conditioning discriminator with mixture distribution they utilized finegrained data distribution for image generation. These methods use unsupervised training techines for generated images to let them clustered within mixture distribution. Unlike these approaches, our model utilized dataset instance as condtion and build diffusion based Gaussian mixture distribution for condtion. These approaches doesn't need such complex unsupervised training for training clustered distribution.

Latent diffusion model. Diffusion Models (DM) have shown impressive results in distribution coverage as well as sample quality. Their sucess are stems from generative power of denoising process, with naturally fitting Unet style Backbon. Recently, there are several approaches to use diffusion model on latent space. Most of these are focussing on decreasing sampling cost by implementing diffusion model on Autoencoder's latent space. Although autoencoder's latent space can be semantic compression space, They can only model such with generator's inductive bias. On the otherhand, our model build such semantic compression with not only generator but also discriminator. By using destylized discriminator for such compression, we could gain better sample quality than these counterparts.

Instancewise prior in GAN training. Recent approaches such as instance conditioned gan or data instance priors utilized instance wise feature vector to transfer knowledge about data distribution. These approaches requires rich source of pre-trained feature extractor and have to maintain subset of instance wise feature vectors. Unlike these approaches, our model use latent diffusion model to gain instance wise feature vector. Due to this difference we could use larger size of instance condition unlike previous works.

5 Discussion

References

Mohammadreza Armandpour, Ali Sadeghian, Chunyuan Li, and Mingyuan Zhou. Partition-guided gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5099–5109, 2021.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Arantxa Casanova, Marlene Careil, Jakob Verbeek, Michal Drozdzal, and Adriana Romero Soriano. Instance-conditioned gan. *Advances in Neural Information Processing Systems*, 34:27517–27529, 2021.

Hamid Eghbal-zadeh, Werner Zellinger, and Gerhard Widmer. Mixture density generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5820–5829, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021.

Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, and Antonio Torralba. Diverse image generation via self-conditioned gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14286–14295, 2020.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pp. 3481–3490. PMLR, 2018.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. *Advances in Neural Information Processing Systems*, 34:17480–17492, 2021.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in Neural Information Processing Systems*, 33:7559–7570, 2020.