

Supplementary Materials: A Novel Confidence Guided Training Method for Conditional GANs with Auxiliary Classifier

Anonymous Authors

A HYPERPARAMETER SETUP FOR BASELINE APPROACHES

As recommended in the paper [2], the weight λ of the classifier of ADC-GAN is set to be 1.0 for CIFAR10/CIFAR100 and ImageNet, and 0.5 for Tiny-ImageNet; the parameter λ in AC-GAN is set to 0.2 for CIFAR10/CIFAR100 and Tiny-ImageNet as it yields the best performance. For the datasets Baby/Papa/Grandpa-ImageNet, we investigate the performance of AC-GAN and ADC-GAN with different values of $\lambda = \{0.1, 0.3, 0.5, 0.7, 1.0\}$. Based on the results, we set that $\lambda = 0.5$ for ADC-GAN and $\lambda = 0.3$ for AC-GAN that yield the best performance. Regarding PD-GAN and ReACGAN, we follow the default hyperparameter settings implemented by StudioGAN.

B PROOF FOR PROPOSITION 3.1.

The main idea of the proof is similar to the proof of [1, Proposition A.1]. Given Q_{XY} , the objective functions for the classifier of CG-GAN in the discriminator, as specified in Equation (7), can be written as the following form

$$\sigma_{\text{sce}}^*(x, y) = \arg \min_{\sigma_{\text{sce}}(x, y)} \mathbb{E}_{x, y \sim P_{XY}} [\sigma_{\text{sce}}(x, y)] + \mathbb{E}_{x, y \sim Q_{XY}} [m - \sigma_{\text{sce}}(x, y)]_+. \quad (22)$$

Note that $\sigma_{\text{sce}}(x, y) \geq 0$. Based on the proof of [8, Lemma 1], the optimal solution $\sigma_{\text{sce}}^*(x, y)$ can be derived as:

$$\sigma_{\text{sce}}^*(x, y) = \begin{cases} 0, & Q_{XY} < P_{XY}, \\ m, & Q_{XY} > P_{XY}, \\ \alpha_x, & Q_{XY} = P_{XY} > 0, \\ \beta_x, & Q_{XY} = P_{XY} = 0, \end{cases} \quad (23)$$

where $\alpha_x \in [0, m]$ and $\beta_x \in [0, +\infty)$. The objective functions for the classifier of CG-GAN in the generator can be formulated as a minimization problem with respect to Q_{XY} :

$$Q_{XY}^* = \arg \min_{Q_{XY}} \mathbb{E}_{x, y \sim Q_{XY}} [\sigma_{\text{sce}}^*(x, y)] - \mathbb{E}_{x, y \sim P_{XY}} [\sigma_{\text{sce}}^*(x, y)], \quad (24)$$

where the second term in Equation (24) is implicitly defined because the problem is an adversarial game between the distributions Q_{XY} and σ_{sce} .

The rest part of the proof follows the same steps as the proof of [1, Proposition A.1]. we only replace the corresponding marginal distribution with the joint distribution, and replace the discriminator function with the cross entropy function. Based on [1, Proposition A.1], the global optimum of the training objective for the classifier of CG-GAN can be achieved if and only if $Q_{XY} = P_{XY}$.

C PROOF FOR LEMMA 3.2.

Since Equation (11) consists of the softmax cross-entropy $\sigma_{\text{sce}}(x, y) = -\log \Pr(y|x)$, we derive the gradient of the softmax cross-entropy $\sigma_{\text{sce}}(x, y)$, w.r.t $w_{k \in \{1, \dots, K\}}$ as following:

$$\frac{\partial \sigma_{\text{sce}}(x, y)}{\partial w_k} = -f(x) \left(\mathbf{1}_{y=k} - \Pr(y|x) \right). \quad (25)$$

Based on Equation (11), we have:

$$L_w(i) = \begin{cases} -\log \Pr(y|x_i^r), & m + \log \Pr(y|x_i^g) < 0; \\ -\log \Pr(y|x_i^r) + m + \log \Pr(y|x_i^g), & \text{otherwise.} \end{cases}$$

Based on Equation (25), it is easy to derive the gradient of $L_w(i)$:

$$\frac{\partial L_w(i)}{\partial w_k} = \begin{cases} G_r(i, k), & \Pr(y|x_i^g) < \exp(-m); \\ G_r(i, k) - G_g(i, k), & \Pr(y|x_i^g) \geq \exp(-m), \end{cases}$$

where $G_r(i, k) = -f(x_i^r) \left(\mathbf{1}_{y=k} - \Pr(y|x_i^r) \right)$ and $G_g(i, k) = -f(x_i^g) \left(\mathbf{1}_{y=k} - \Pr(y|x_i^g) \right)$. Here, $\mathbf{1}_{y=k}$ is the indicator function that outputs 1 if $y = k$.

D PROOF FOR COROLLARY 3.3.

Based on Proposition 3.1, we have $Q_{XY} = P_{XY}$. Particularly, when $Q_{XY} = P_{XY}$, the objective functions for the classifier of CG-GAN in the discriminator, (i.e., Equation (7)), can be written as the following form

$$\begin{aligned} \varphi(\sigma_{\text{sce}}(x, y)) &= \mathbb{E}_{x, y \sim P_{XY}} [\sigma_{\text{sce}}(x, y)] + \mathbb{E}_{x, y \sim Q_{XY}} [m - \sigma_{\text{sce}}(x, y)]_+ \\ &= \mathbb{E}_{x, y \sim P_{XY}} [\sigma_{\text{sce}}(x, y)] + \mathbb{E}_{x, y \sim P_{XY}} [m - \sigma_{\text{sce}}(x, y)]_+, \end{aligned} \quad (26)$$

Then the proof is similar to the proof of [8, Lemma 1]. The derivative of the function φ is $\varphi' = P_{XY} - P_{XY} = 0$ for $\sigma_{\text{sce}}(x, y) \in [0, m]$; Meanwhile, for $\sigma_{\text{sce}}(x, y) \in (m, +\infty)$, the derivative is $\varphi' = P_{XY} > 0$. So the function φ reaches its minimum value when $\sigma_{\text{sce}}(x, y) \in [0, m]$. And the corresponding optimal confidence function $\Pr^*(y|x)$ may be any value between $\exp(-m)$ and 1.

E THE CONNECTION WITH SOME REGULARIZATION TECHNIQUES ON THE INTRODUCED KL TERM

In this section, we examine the connection between the KL term which we have introduced in Section 3.3 and various standard regularization methods used to mitigate overfitting in neural networks. Firstly, we define the softmax cross-entropy loss function as follows. Given the prior label distribution $Q = \{q_1, q_2, \dots, q_K\}$ and the prediction distribution $P(x) = \{p_1(x), p_2(x), \dots, p_K(x)\}$ by the classifier, the standard softmax cross-entropy loss can be

defined as:

$$H(Q, P(x)) = \sum_{j=1}^K -q_j \log p_j(x) = \sum_{j=1}^K -q_j \log \frac{\exp(l_j(x))}{\sum_{k=1}^K \exp(l_k(x))}. \quad (27)$$

Connection with label smoothing: Label smoothing [6] is a regularization technique for mitigating overfitting in neural networks. Given a parameter α , training a classifier with label smoothing is to minimize the cross-entropy between a prior label distribution P^{ls} and the prediction distribution of classifier $P(x)$. The P^{ls} is defined as:

$$P^{ls} = [\underbrace{\frac{\alpha}{K}, \dots, 1 - \alpha + \frac{\alpha}{K}}_{\text{The } y\text{-th item}}, \dots, \frac{\alpha}{K}].$$

Let $\alpha = (1 - \exp(-m)) \cdot \frac{K}{K-1}$, then $\tilde{P} = P^{ls}$. Consequently, we have:

$$\begin{aligned} \min \text{KL}(\tilde{P}||P(x)) &= \min \text{KL}(P^{ls}||P(x)) \\ &= \min H(P^{ls}, P(x)) - \underbrace{H(P^{ls}, P^{ls})}_{\text{Constant for optimization}} \\ &= \min H(P^{ls}, P(x)), \end{aligned} \quad (28)$$

Equation (28) indicates that it is equivalent to add one side (for generated data) label smoothing to CG-GAN when we use $\text{KL}(\tilde{P}||P(x))$. Particularly, let $\exp(-m) = \frac{1}{K}$, \tilde{P} is a uniform distribution, and $\text{KL}(\tilde{P}||P(x))$ is equivalent to the KL divergence between the uniform distribution and the predicted distribution $P(x)$ of classifier.

Connection with confidence penalty: The confidence penalty [5] aims to penalize a low-entropy output distribution of classifier by adding the KL divergence between the predicted distribution $P(x)$ of the classifier and the uniform distribution u . When $\exp(-m) = \frac{1}{K}$, $\text{KL}(P(x)||\tilde{P})$ is equivalent to confidence penalty (i.e., $\text{KL}(P(x)||u)$). Moreover, we have

$$\begin{aligned} \min \text{KL}(P(x)||u) &= \min H(P(x), u) - H(P(x), P(x)) \\ &= \min -\log(u) - H(P(x), P(x)) \\ &= \max H(P(x), P(x)), \end{aligned} \quad (29)$$

where $H(P(x), P(x))$ is the conditional entropy of the input.

F DISCUSSION OF DIFFERENT EVALUATION PROTOCOLS.

Please note that some papers, such as the StudioGAN paper, utilize the training dataset for evaluating metrics. However, we follow the same protocol as [3, 7] by employing the validation dataset as the default reference distribution for computing evaluation metrics. For CIFAR10 and CIFAR100, we use the test dataset due to the absence of the validation dataset. We illustrate the disparity between the evaluations on the training and validation sets in Table 1, with a particular focus on the significant inconsistencies observed in the FID metric.

G MORE DETAILS FOR DATASETS

CIFAR10/100 are commonly used benchmark datasets for evaluating GANs. CIFAR10 consists of 60,000 RGB images with resolution 32×32 of 10 classes. CIFAR-100 consists of 100 classes and contains 600

Table 1: Evaluation on CIFAR10 and Baby/Papa-ImageNet using training/validation datasets. *: The reported performance is evaluated using the public checkpoint provided by the StudioGAN paper.

Datasets	Methods	FID (train)↓	FID (valid)↓
CIFAR10	ReACGAN*	3.91	7.889
	ReACGAN	4.13	8.026
	rCG-GAN	3.45	7.514
Baby-ImageNet	ReACGAN*	21.558	32.994
	ReACGAN	18.647	27.5857
	rCG-GAN	11.534	21.4124
Papa-ImageNet	ReACGAN*	22.884	31.369
	ReACGAN	20.875	29.6279
	rCG-GAN	13.504	23.4174

images for each class and the resolution is 32×32 . Tiny-ImageNet (64×64) contains 120000 images of 200 classes. The ImageNet training set is composed of about 1.28 million images from 1000 different categories.

Baby/Papa/Grandpa-ImageNet (64×64) are created by StudioGAN for small-scale ImageNet experiments due to the extensive computational resources required to train GAN on the full ImageNet. These subsets of ImageNet are created based on the classification difficulty: the Baby-ImageNet is the easiest to classify, whereas the Granpa-ImageNet is the most difficult to classify. StudioGAN computes the class-wise accuracy of ImageNet 1,000 classes using the InceptionV3 network and picks the real images of the same class according to the rank on class-wise accuracy (1st~100th: Baby-ImageNet, 451st~550th: Papa-ImageNet, and 901st~1,000th: Granpa-ImageNet). For more details please refer to the StudioGAN paper [4].

H MORE RESULTS

Impact of the weight of the classifier. For simplicity, let $\lambda = \lambda_1 = \lambda_2$ within the objective functions of rCG-GAN, as shown in Equation (15) and Equation (16). We consider the FID curve with varying the parameter λ . From Figure 1 we can see that our rCG-GAN has higher robustness with respect to the large weight of the classifier λ , and AC-GAN performs substantially worse when λ becomes larger.

Impact of the desired confidence. Table 2 shows the results of the hyperparameter search for different values of the desired confidence $\exp(-m)$ on CIFAR-10, CIFAR-100, Tiny ImageNet and Baby/Papa/Grandpa-ImageNet. As depicted in Figure 2, “Density” of our rCG-GAN is improved with increasing desired confidence. The values of “FID” and “Coverage” exhibit minimal changes. As shown in Figures 2b and 2c, it is evident that there is only a slight change in FID and Coverage for different values of $\exp(-m)$ on CIFAR100. The result suggests that our rCG-GAN exhibits robustness to variations in $\exp(-m)$ in terms of FID and Coverage. However, it should be noted that Density is more susceptible to changes in confidence levels as shown in Figure 2a.

The average norm of input feature maps. It is shown in Figures 3a and 3b that the regularization term, particularly the

Table 2: Evaluation on CIFAR10, CIFAR100, Tiny-ImageNet, Baby/Papa/Grandpa-ImageNet. The “conf.” denotes the desired confidence. The CIFAR10 dataset contains 10 classes, while the Tiny-ImageNet dataset consists of 200 classes. The remaining datasets consist of 100 classes each.

Datasets	Methods	conf.	IS \uparrow	FID \downarrow	Density \uparrow	Coverage \uparrow	Precision \uparrow	Recall \uparrow
CIFAR10	fCG-GAN	0.11	10.058	8.188	0.9943	0.9281	0.7396	0.6976
		0.4	10.272	7.701	1.082	0.9356	0.773	0.675
	rCG-GAN	0.11	9.933	8.084	1.0272	0.9292	0.7527	0.6919
		0.4	10.285	7.514	1.109	0.9396	0.7759	0.6736
CIFAR100	fCG-GAN	0.011	13.4889	9.5285	0.9995	0.9192	0.7786	0.6568
		0.02	14.291	9.505	1.048	0.9233	0.8008	0.6307
	rCG-GAN	0.011	13.5334	9.479	1.0149	0.9185	0.7894	0.6603
		0.02	14.0678	9.46	1.0721	0.9248	0.8092	0.6222
Tiny-ImageNet	rCG-GAN	0.0051	17.23	17.959	0.8575	0.773	0.7122	0.6138
		0.007	19.657	16.83	0.8965	0.8146	0.7344	0.5981
Baby-ImageNet	rCG-GAN	0.011	32.1185	21.6787	0.8080	0.7972	0.7237	0.6967
		0.02	31.5075	21.4124	0.7792	0.7644	0.7289	0.6831
Papa-ImageNet	rCG-GAN	0.011	24.0439	25.0798	0.8327	0.789	0.7210	0.6348
		0.02	26.9556	23.4174	0.8396	0.8086	0.7288	0.6352
Grandpa-ImageNet	rCG-GAN	0.011	20.4053	24.2695	0.8851	0.8274	0.6996	0.5724
		0.02	22.445	22.679	0.9006	0.856	0.7248	0.579

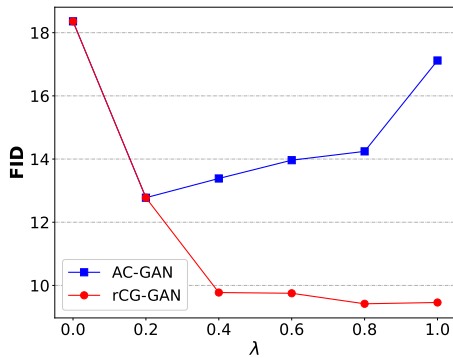


Figure 1: FID on CIFAR-100 with different λ .

KL term we have introduced in Section 3.3, effectively lowers the feature norm for both real and generated data compared to that of the CG-GAN. The reduced feature norm leads to an improvement in the stability of the rCG-GAN over the basic CG-GAN, as depicted in Figure 3c.

I QUALITATIVE RESULTS

Figure 4 illustrates the images generated by our rCG-GAN for ImageNet. Figures 5 to 8 illustrate some qualitative results of our rCG-GAN and the baseline approaches.

REFERENCES

- [1] Zihang Dai, Amjad Almahairi, Philip Bachman, Eduard H. Hovy, and Aaron C. Courville. 2017. Calibrating Energy-based Generative Adversarial Networks. In *ICLR (Poster)*. OpenReview.net.
- [2] Liang Hou, Qi Cao, Huawei Shen, Siyuan Pan, Xiaoshuang Li, and Xueqi Cheng. 2022. Conditional GANs with Auxiliary Discriminative Classifier. In *ICML (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, 8888–8902.
- [3] Minguk Kang, Woohyeon Shim, Minsu Cho, and Jaesik Park. 2021. Rebooting ACGAN: Auxiliary Classifier GANs with Stable Training. In *NeurIPS*. 23505–23518.
- [4] Minguk Kang, Joonghyuk Shin, and Jaesik Park. 2023. StudioGAN: a taxonomy and benchmark of GANs for image synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
- [5] Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. 2017. Regularizing Neural Networks by Penalizing Confident Output Distributions. In *ICLR (Workshop)*. OpenReview.net.
- [6] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. In *CVPR*. IEEE Computer Society, 2818–2826.
- [7] Han Zhang, Zizhao Zhang, Augustus Odena, and Honglak Lee. 2019. Consistency Regularization for Generative Adversarial Networks. In *International Conference on Learning Representations*.
- [8] Junbo Jake Zhao, Michaël Mathieu, and Yann LeCun. 2017. Energy-based Generative Adversarial Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*.

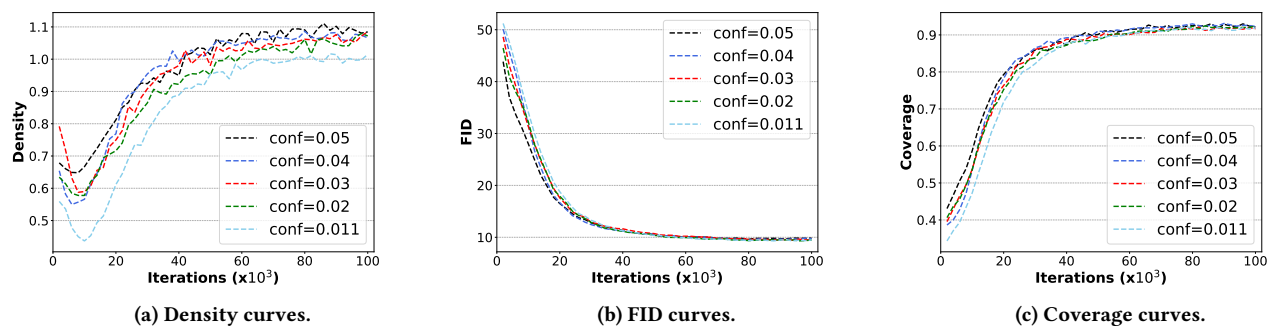


Figure 2: The experiments on CIFAR100.

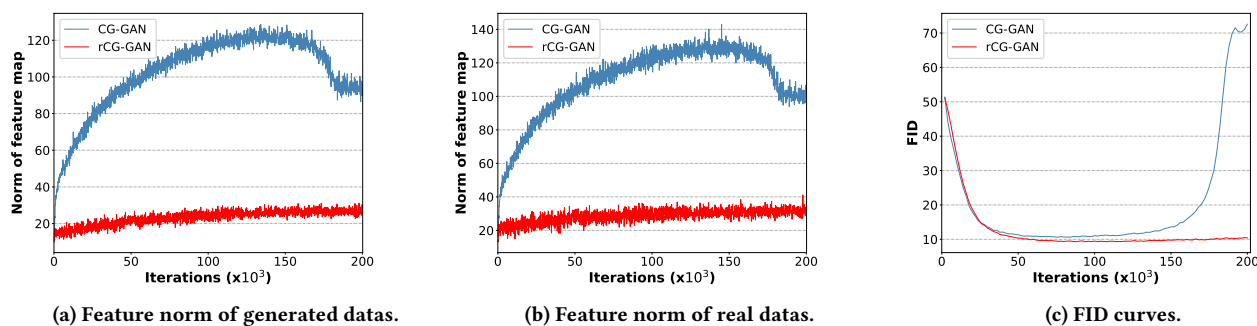


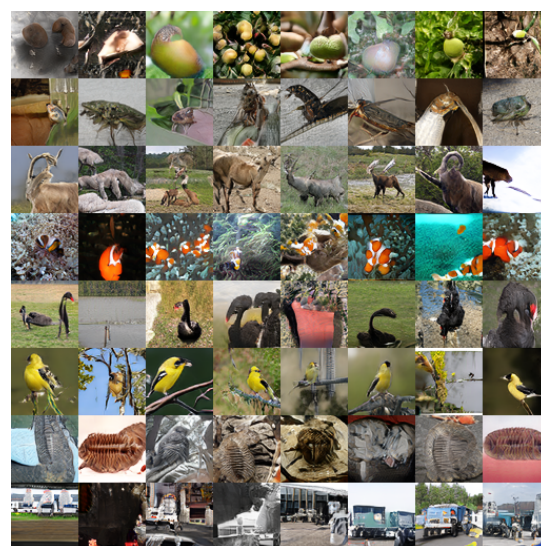
Figure 3: The average norm of input feature maps on CIFAR100.



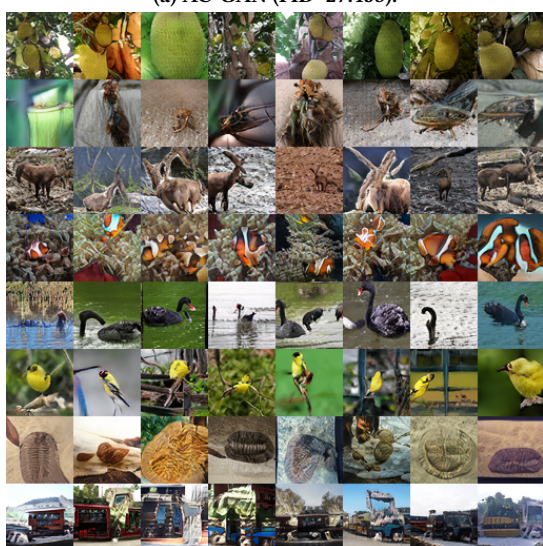
Figure 4: Generated images on ImageNet dataset using rCG-GAN (FID=5.187).



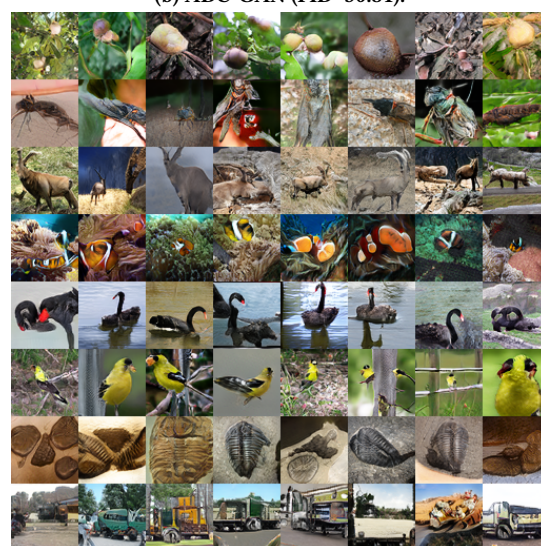
(a) AC-GAN (FID=27.453).



(b) ADC-GAN (FID=30.81).

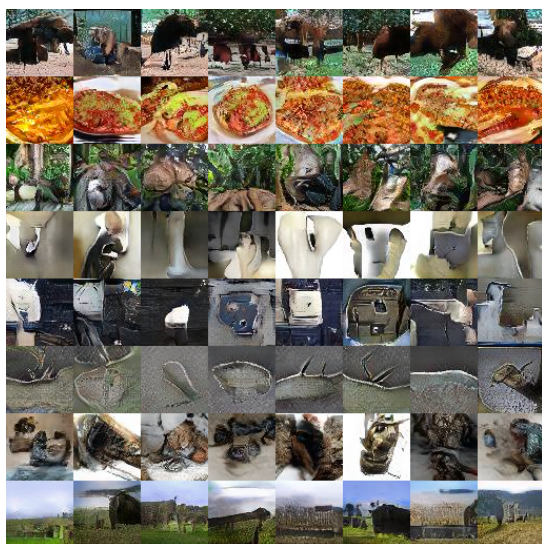


(c) ReACGAN (FID=27.5857).

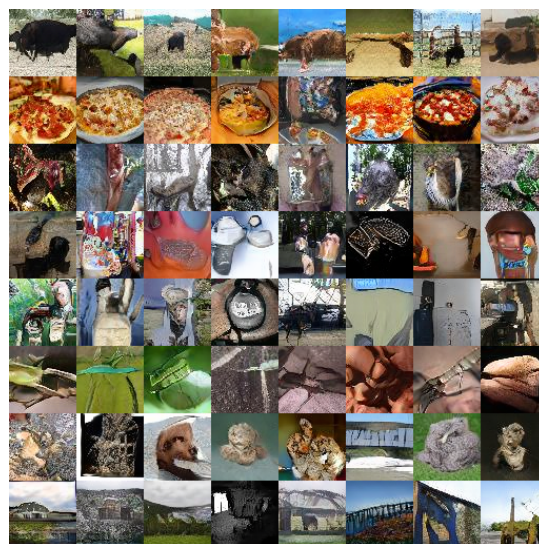


(d) rCG-GAN (FID=21.41).

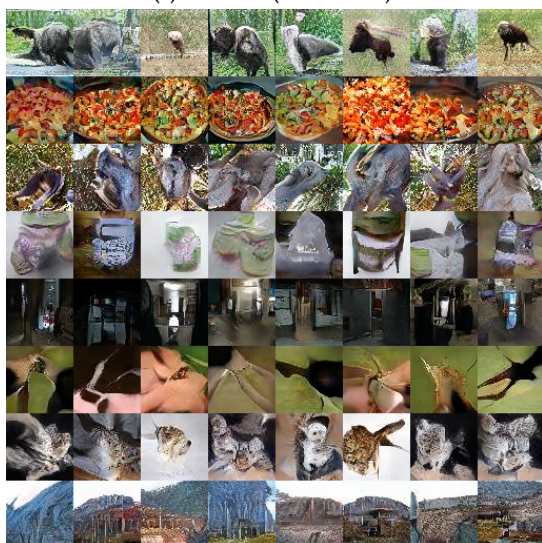
Figure 5: Generated images on Baby-ImageNet dataset.



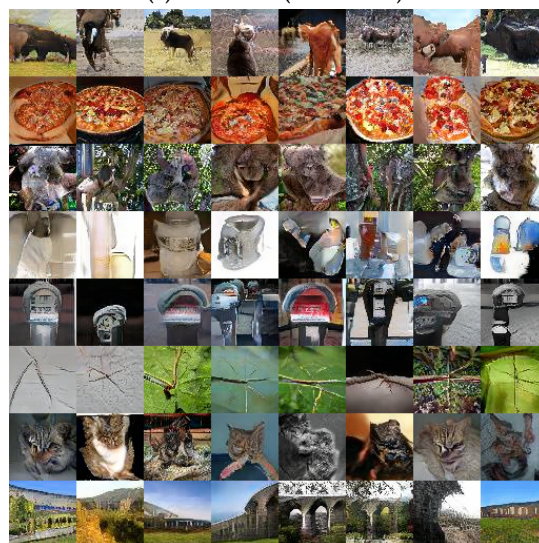
(a) AC-GAN (FID=36.799).



(b) ADC-GAN (FID=26.682).

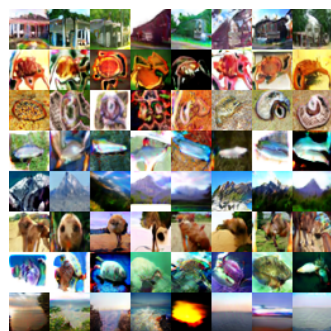


(c) ReACGAN (FID=30.4484).

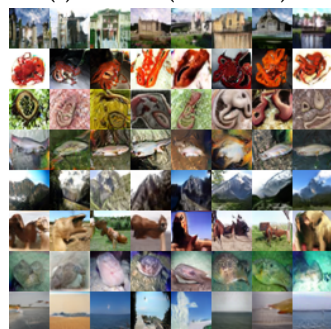


(d) rCG-GAN (FID=16.83).

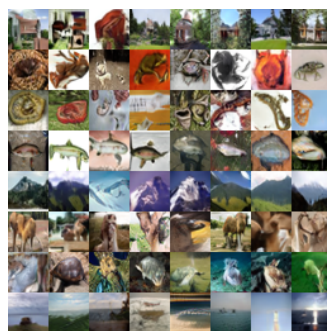
Figure 6: Generated images on Tiny-ImageNet dataset.



(a) AC-GAN (FID=12.777).



(c) ReACGAN (FID=12.1964).

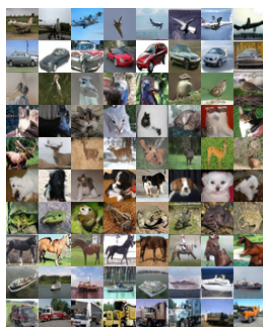


(b) ADC-GAN (FID=10.7903).

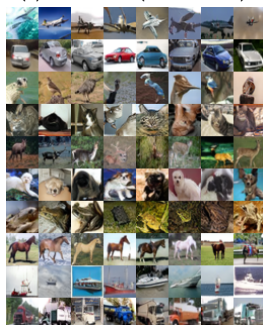


(d) rCG-GAN (FID=9.46).

Figure 7: Generated images on CIFAR100 dataset.



(a) AC-GAN (FID=8.342).



(c) ReACGAN (FID=8.026).



(b) ADC-GAN (FID=8.0266).



(d) rCG-GAN (FID=7.514).

Figure 8: Generated images on CIFAR10 dataset.