NICE: NoIse-modulated Consistency rEgularization for Data-Efficient GANs

Yao Ni[†], Piotr Koniusz[§,†]
[†]The Australian National University [§]Data61♥CSIRO
yao.ni@anu.edu.au, piotr.koniusz@data61.csiro.au

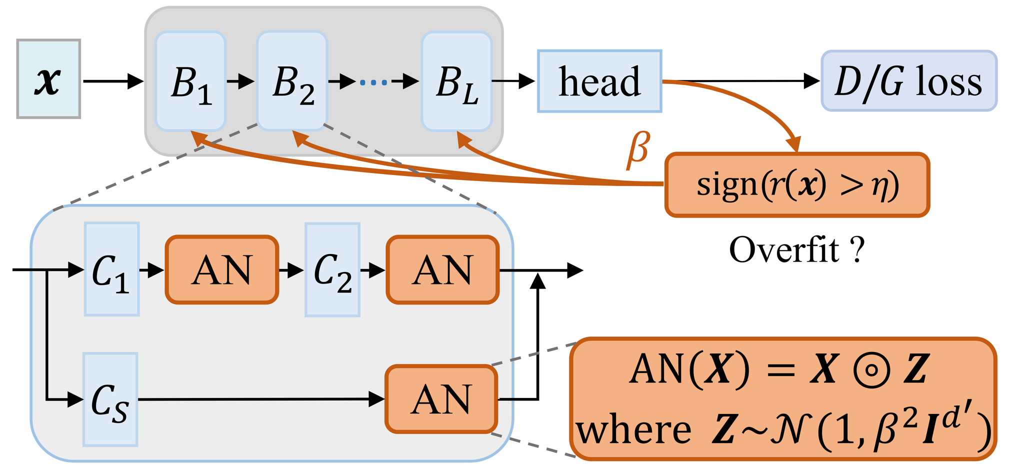
Summary

Background: Limited data in GAN training causes discriminator overfitting and training instability.
Goal: To improve the generalization of GANs.
Contributions:

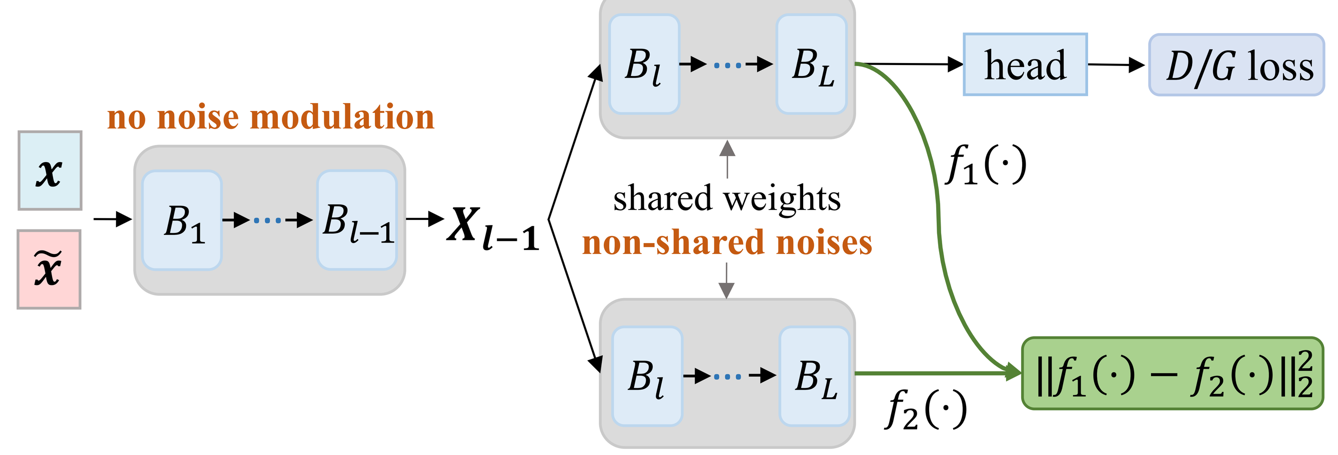
- We use an adaptive multiplicative noise to modulate latent features of discriminator to boost generalization of GAN.
- We introduce NICE to enforce the discriminator consistency across varying noise modulations, implicitly penalizing first and second-order gradients of discriminator latent features to improve the stability of training.
- We showcase theoretical and practical effectiveness of NICE in preventing discriminator overfitting. We achieve superior performance in image generation with limited data.

Pipeline

Discriminator with adaptive noise (AN) [a] :



NoIse-modulated Consistency rEgularization (NICE):



Update β: control the variance of noise by monitoring $r(\boldsymbol{x})=\mathbb{E}[\text{sign}(D(\boldsymbol{x}))]$. Update $\beta_{t+1}=\beta_t+\Delta_\beta\cdot\text{sign}\,(r(\boldsymbol{x})>\eta)$.

NICE: weight regularization → better generalization

NICE: gradient penalization → stable training

[a] d': feature size. $\circledcirc$: expands $\boldsymbol{Z}$ to $d'\times d^H\times d^W$ then performs element-wise multiplication. B_l: l-th block. C_S: Convolution in skip branch. f: feature extractor. $\boldsymbol{x}/\tilde{\boldsymbol{x}}$: real/fake image. η: a threshold.

Method

Reducing the weight norms of D improves generalization:

n: dataset size. $\mathcal{H}/\mathcal{G}$: D/G sets. $\forall h\in\mathcal{H}$, $\|h\|_\infty\leq\Delta$. μ/ν: measures on real/fake data. $\hat{\mu}_n/\nu_n$: empirical measures. Assume $d_{\mathcal{H}}(\hat{\mu}_n,\nu_n)-\inf_{\nu\in\mathcal{G}} d_{\mathcal{H}}(\hat{\mu}_n,\nu)\leq\epsilon$.

$$\underbrace{d_{\mathcal{H}}(\mu,\nu_n)-\inf_{\nu\in\mathcal{G}} d_{\mathcal{H}}(\mu,\nu)}_{\text{Generalization error of GAN.}}\leq 2\underbrace{R_n^{(\mu)}(\mathcal{H})}_{\text{Complexity of } D.}+2\Delta\sqrt{2\log(1/\delta)/n}+\epsilon$$

For $\forall i\in\{1,\dots,n\}$, $\|\boldsymbol{x}^{(i)}\|_2\leq q$ and a t-layer fully-connected network parameterized from set $\mathcal{V}=\{v_{\boldsymbol{\theta}}:\|\boldsymbol{W}_i\|_{\text{lip}}\leq k_i, \|\boldsymbol{W}_i^T\|_{2,1}\leq b_i\}$:

$$\underbrace{R_n^{(\mu)}(\mathcal{V})}_{\text{Rademacher complexity.}}\leq\frac{q}{\sqrt{n}}\cdot\Big(\prod_{i=1}^t k_i\Big)\cdot\Big(\sum_{i=1}^t \underbrace{b_i^{2/3}}_{\text{Weight norm.}}/k_i^{2/3}\Big)^{3/2}$$

Multiplicative noise modulation reduces weight norms:

$\boldsymbol{w}_k$: the k-th column vector of the second layer weight $\boldsymbol{W}_2$. $\hat{a}_k$: mean feature norm ≥ 0. β^2: variance of noise. $\boldsymbol{y}$: label. Multiplicative noise modulation $\boldsymbol{z}$ on the latent feature $\boldsymbol{a}^{(i)}$ in a two-layer net induces weight regularization.

$$\hat{L}_{\text{noise}}(w):=\hat{\mathbb{E}}_i\mathbb{E}_{\boldsymbol{z}}\big[\|\boldsymbol{y}^{(i)}-\boldsymbol{W}_2(\boldsymbol{z}\odot\boldsymbol{a}^{(i)})\|_2^2\big]$$
$$=\hat{\mathbb{E}}_i\big[\|\boldsymbol{y}^{(i)}-\boldsymbol{W}_2\boldsymbol{a}^{(i)}\|_2^2\big]+\underbrace{\beta^2\sum_k\hat{a}_k\|\boldsymbol{w}_k\|_2^2}_{\text{Implicit regularization on } \|\boldsymbol{w}_k\|_2.}$$

Noise modulation causes gradient norm amplification:

$$\min_{\boldsymbol{\theta}_d} L_D^{\text{AN}}:=\mathbb{E}_{\tilde{\boldsymbol{a}}}\mathbb{E}_{\boldsymbol{z}}\big[h(\boldsymbol{z}\odot\tilde{\boldsymbol{a}})\big]-\mathbb{E}_{\boldsymbol{a}}\mathbb{E}_{\boldsymbol{z}}\big[h(\boldsymbol{z}\odot\boldsymbol{a})\big]$$
$$\approx\mathbb{E}_{\tilde{\boldsymbol{a}}}[h(\tilde{\boldsymbol{a}})]-\mathbb{E}_{\boldsymbol{a}}[h(\boldsymbol{a})]$$
$$+\frac{\beta^2}{2}\Big(\mathbb{E}_{\tilde{\boldsymbol{a}}}\Big[\sum_k\tilde{a}_k^2H_{kk}^{(h)}(\tilde{\boldsymbol{a}})\Big]-\mathbb{E}_{\boldsymbol{a}}\Big[\sum_k a_k^2H_{kk}^{(h)}(\boldsymbol{a})\Big]\Big)$$

$$\min_{\boldsymbol{\theta}_g} L_G^{\text{AN}}:=-\mathbb{E}_{\boldsymbol{z}}\mathbb{E}_{\tilde{\boldsymbol{a}}}\big[h(\boldsymbol{z}\odot\tilde{\boldsymbol{a}})\big]$$
$$\approx-\mathbb{E}_{\tilde{\boldsymbol{a}}}\big[h(\tilde{\boldsymbol{a}})\big]-\frac{\beta^2}{2}\mathbb{E}_{\tilde{\boldsymbol{a}}}\Big[\sum_k\tilde{a}_k^2H_{kk}^{(h)}(\tilde{\boldsymbol{a}})\Big]$$

$\boldsymbol{a}/\tilde{\boldsymbol{a}}$: real/fake feature. $H^{(h)}(\boldsymbol{a})$: Hessian of h at $\boldsymbol{a}$. $\odot$: element-wise product.

Consistency regularization (NICE) lowers gradient norm:

$$\ell^{\text{NICE}}(\boldsymbol{a}):=\mathbb{E}_{\boldsymbol{z}_1,\boldsymbol{z}_2}\Big[\big(f(\boldsymbol{z}_1\odot\boldsymbol{a})-f(\boldsymbol{z}_2\odot\boldsymbol{a})\big)^2\Big]$$
$$\approx 2\beta^2\sum_k a_k^2\nabla_k^2 f(\boldsymbol{a})+\beta^4\sum_{j,k}a_j^2a_k^2(H_{jk}^{(f)}(\boldsymbol{a}))^2$$

$\nabla f(\boldsymbol{a})$, $H^{(f)}(\boldsymbol{a})$: gradient and Hessian matrix of feature extractor f at $\boldsymbol{a}$. $H_{jk}^{(f)}$: (j,k)-th entry of $H^{(f)}$.

We apply NICE on real & fake images when training G & D.

Experimental Results

Comparison with the state of the art:

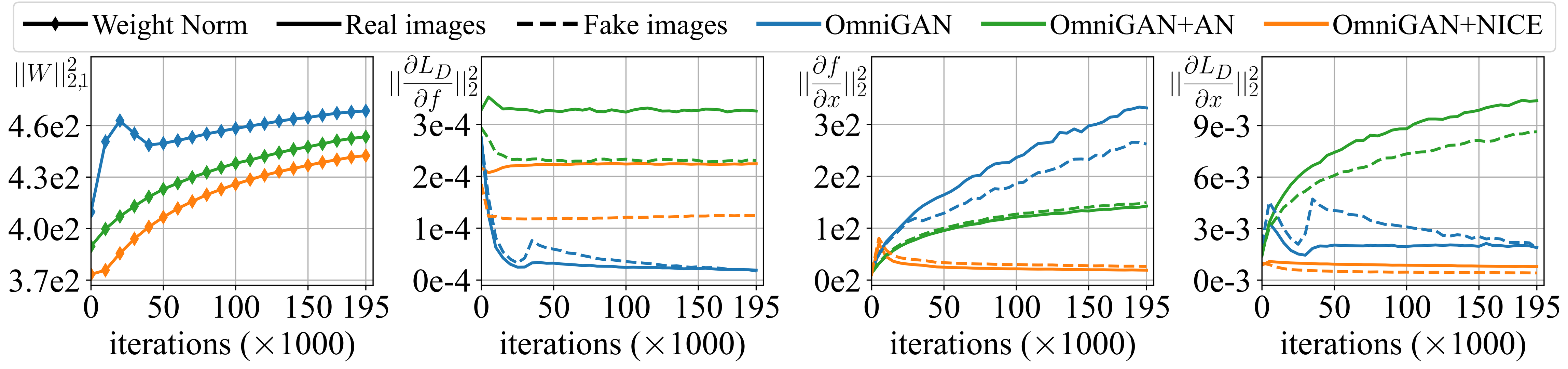
Method	CIFAR-10			CIFAR-100		
	100% data	20% data	10% data	100% data	20% data	10% data
	IS↑/tFID↓	IS↑/tFID↓	IS↑/tFID↓	IS↑/tFID↓	IS↑/tFID↓	IS↑/tFID↓
BigGAN	9.21/5.48	8.74/16.20	8.24/31.45	11.02/7.86	9.94/25.83	7.58/50.79
+NICE	**9.50/4.19**	**8.96/8.51**	**8.73/13.65**	10.99/**6.31**	**10.32/13.17**	**8.96/19.53**
LeCam+DA	9.45/4.32	9.01/8.53	8.81/12.64	11.25/6.45	10.12/15.96	9.17/22.75
+NICE	**9.52/3.72**	**9.12/6.92**	**8.99/9.86**	**11.28/5.72**	**10.54/10.02**	**9.35/14.95**
OmniGAN+ADA	10.24/4.95	9.41/27.04	7.86/40.05	13.07/6.12	12.07/13.54	8.95/44.65
+NICE	**10.38/2.25**	**10.18/4.39**	**10.08/5.49**	**13.82/3.78**	**12.75/6.28**	**12.04/9.32**

Method	FID ↓ on ImageNet		
	10%	5%	2.5%
BigGAN	38.30	91.16	133.80
ADA	31.89	43.21	56.83
DA	32.82	56.75	63.49
MaskedGAN	26.51	35.70	38.62
KDDLGAN	20.32	22.35	28.79
NICE	21.44	24.72	31.45
ADA+NICE	**18.29**	**20.07**	**24.41**

Method (FID↓)	Obama	GrumpyCat	Panda	AnimalCat	AnimalDog
StyleGAN2	80.20	48.90	34.27	71.71	131.90
StyleGAN2+NICE	**24.56**	**18.78**	**8.92**	**25.25**	**46.56**
ADA	45.69	26.62	12.90	40.77	56.83
LeCam+KDDLGAN	29.38	19.65	8.41	31.89	50.22
ADA+NICE	**20.09**	**15.63**	**8.18**	**22.70**	**28.65**

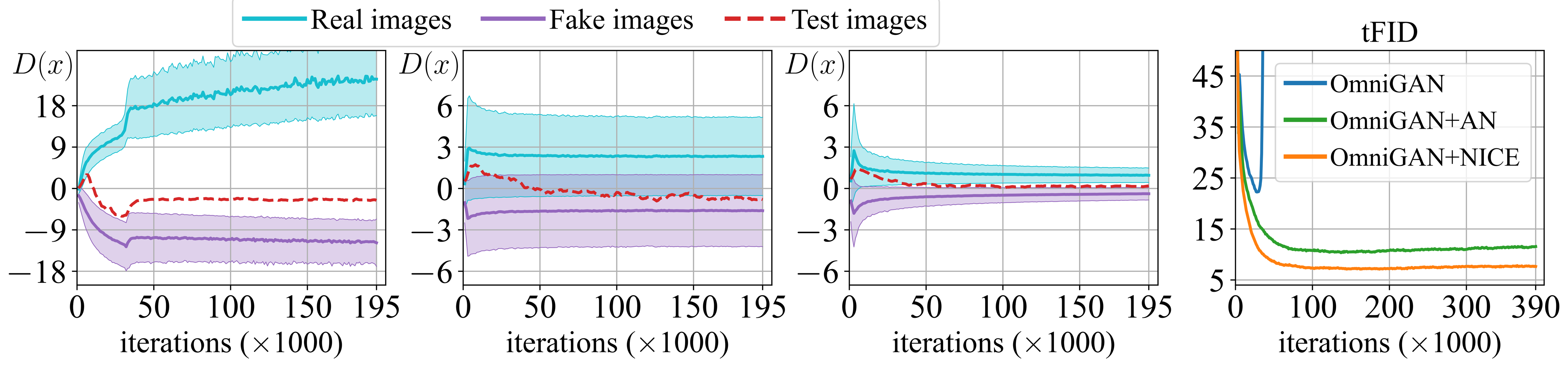
Method (FID↓ on FFHQ)	100	$1K$	$2K$	$5K$
StyleGAN2	179	100.16	54	49.68
ADA	85.8	21.29	15.39	10.96
ADA-Linear	82	19.86	13.01	9.39
InsGen	45.75	18.21	11.47	7.83
FakeCLR	42.56	15.92	9.90	7.25
ADA+NICE	**38.42**	**14.57**	**8.85**	**6.48**

NICE reduces weight and gradient norms in the discriminator:



Generated Images:

NICE minimizes the discrepancy between training and test images:



OmniGAN | OmniGAN+AN | OmniGAN+NICE | tFID curves