SPATIALLY DECOMPOSED HINGE ADVERSARIAL LOSS BY LOCAL GRADIENT AMPLIFIER

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative Adversarial Networks (GANs) have achieved large attention and great success in various research areas, but it still suffers from training instability. Recently hinge adversarial loss for GAN is proposed that incorporates the SVM margins where real and fake samples falling within the margins contribute to the loss calculation. In a generator training step, however, fake samples outside of the margins that partially include unrealistic local patterns are ignored. In this work, we propose local gradient amplifier(LGA) which realizes spatially decomposed hinge adversarial loss for improved generator training. Spatially decomposed hinge adversarial loss applies different margins for different spatial regions extending overall margin space toward all fake samples asymmetrically. Our proposed method is evaluated on several public benchmark data sets compared to state of the art methods showing outstanding stability in training GANs.

1 INTRODUCTION

Generative Adversarial Networks (Goodfellow et al., 2014) have achieved large attention and great success in various research areas since it was introduced. GAN consists of two adversarial networks (Generator and Discriminator) that are trained alternately. Discriminator is trained to distinguish between real and generated fake samples. On the other hand, generator is trained based on the feedback from the discriminator to make fake samples that are classified as real by the discriminator. Thanks to its practical performance of adversarial training strategy, GAN evolved to various fields such as image to image translation(Isola et al., 2017; Wang et al., 2018; Park et al., 2019), super resolution(Ledig et al., 2017), text to image generation(Zhang et al., 2017), etc.

As the utilization of GAN has been expanded, it faces limited performance with images of increased complexity and diversity in its visual characteristics. As a result, huge networks and datasets are built that, in turn, make it more difficult to train GAN. WGAN(Arjovsky et al., 2017) changes *Jensen-Shannon* divergence based adversarial loss to *Earth-Mover* distance based adversarial loss. This prevents it from causing vanishing gradient when GAN has optimal Discriminator. In order to use wGAN adversarial loss, Lipschitz constraint must be satisfied. WGAN-GP(Gulrajani et al., 2017) satisfies Lipschitz constraint by adding gradient penalty to regularization term, and SNGAN(Miyato et al., 2018) satisfies Lipschitz constraint through spectral normalization. McGAN (Mroueh et al., 2017) defines wGAN adversarial loss as mean feature matching. GeometricGAN(Lim & Ye, 2017) applies hyper-plane of soft-margin SVM to wGAN (Arjovsky et al., 2017) by hinge adversarial loss so that the discriminator of GAN maximizes margin between two classes. The authors mention that GeometricGAN converges to a Nash equilibrium between discriminator and generator. Hinge adversarial loss (Tran et al., 2017; Lim & Ye, 2017) applied to the discriminator lets real and fake samples falling within the SVM margins contribute to the loss, which increases training performance. In a generator training step, however, fake samples outside of the margins that partially include unrealistic local patterns are ignored in the training.

In this work, we propose spatially decomposed hinge adversarial loss for improved generator training. Spatially decomposed hinge adversarial loss applies different margins for different spatial regions extending overall margin space toward all fake samples asymmetrically. To this end, we implement a local gradient amplifier (LGA). Spatially decomposed hinge adversarial loss works as respective gradient amplifiers in a fake sample in our method. Gradient of unrealistic region is amplified, while gradient of realistic region is sustained. Our decomposed hinge loss is a training

scheme of dividing target goal of generating a real sample into separate multiple optimization of local regions.

Consider a case in which more than a half of generated image including most of foreground region looks realistic, however, remaining part is unrealistic. Practical policy for faster and stable convergence is encouraging update for such unrealistic region while keeping all other realistic image regions. Our proposed decomposed hinge adversarial loss locally amplifies the gradients of unrealistic regions allowing spatially adapted loss calculation and generator network update. This kind of approach is reasonable in training a generative network because there is no fixed target image that has to be globally optimal in spatial image space. Instead, perceptually acceptable local image parts could be seamlessly combined building a local optimal generated image.

The contributions of our proposed work are as follows. 1) We propose spatially decomposed hinge adversarial loss for generator training that improves the convergence and stability of GAN maintaining perceptual image generation performance. 2) We propose Local Gradient Amplifier(LGA) to effectively implement our spatially decomposed hinge adversarial loss in GAN. Since the structure of LGA is simple and works with any given networks, it can be easily adopted to various other GANs. 3) Extensive evaluation experiments show that spatially decomposed hinge adversarial loss makes generator converge faster and represent complex and diverse images better.

2 Related Works

2.1 wGAN and Lipschitz Constraint

Training GAN is min-max game in which generator G and discriminator D compete with objective function below. Discriminator tries to distinguish between real and fake samples, and generator tries to deceive the discriminator.

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim P_r}[log(D(x))] - \mathop{\mathbb{E}}_{z\sim P_z}[log(1-D(G(z)))] \tag{1}$$

where x and z indicate real samples and random latent vectors, respectively. In general, if discriminator is too accurate, generator is difficult to acquire direction of training due to vanishing gradient problem. wGAN shows that *Jensen-Shannon* divergence causes gradient vanishing when GAN has optimal discriminator. wGAN design a discriminator loss using *Earth-Mover* distance so that GAN is capable of reflecting distance required to move fake distribution to real distribution.

$$EMD = \inf_{\gamma\in\prod(\mathbb{P}_r,\mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}||x-y|| \tag{2}$$

where $\prod(\mathbb{P}_r,\mathbb{P}_g)$ denotes the set of all joint distributions, and $\gamma(x,y)$ indicates joint distribution. However it is difficult to calculate joint distribution for all real and fake samples for calculating EMD. Minimizing the EMD transforms nice equation through *Kantovich-Rubinstein* duality to exclude joint distribution.

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(x)] - \mathop{\mathbb{E}}_{z\sim\mathbb{P}_z}[D(G(z))] \tag{3}$$

In wGAN, critic (Discriminator of wGAN) no longer calculates probabilities of samples. Critic searches for decision boundary which can distinguish between two classes. This solves gradient vanishing that occurs when GAN has optimal discriminator. In order to use minimizing problem of EMD as maximizing problem, discirminator must satisfy $||f||_{Lip} \leq 1$(*kantovich-Rubinstein* duality). Lipschitz constraint for critic f is as follows.

$$\frac{||f(x)-f(x')||}{||x-x'||} \leq 1 \tag{4}$$

wGAN constrains weights of Discriminator to exist in $[-c,c]$ space through weight clipping to satisfy Lipschitz constraint. WGAN-GP(Gulrajani et al., 2017) argue that weight clipping frequently causes over fitting, and training is affected too much by hyper-parameter c. WGAN-GP suggests gradient penalty to satisfy Lipschitz constraint. They add gradient penalty as regularization term of critic loss.

$$L = \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[D(x)] - \mathop{\mathbb{E}}_{z\sim\mathbb{P}_z}[D(G(z))] + \lambda \mathop{\mathbb{E}}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}[(||\nabla_{\hat{x}}D(\hat{x})||_2 - 1)^2] \tag{5}$$

Gradient penalty finds a random sample $\hat{x}$ between real and fake samples. This $\hat{x}$ is located in direction in which fake sample is to be learned. So making $\nabla_{\hat{x}} D(\hat{x}) \leq 1$ for all random samples $\hat{x}$, Lipschitz constraint can be satisfied through entire learning process. SNGAN(Miyato et al., 2018) focuses on Lipschitz's inequality $||g_1 \circ g_2||_{Lip} \leq ||g_1||_{Lip} \circ ||g_2||_{Lip}$. They argue that if each layer of discriminator satisfies Lipschitz constraint, discriminator can satisfy Lipschitz constraint as well.

$$||f||_{Lip} \leq \prod_{l=1}^{L+1} ||h_{l-1} \mapsto W^l h_{l-1}||_{Lip} \tag{6}$$

For a linear layer $g(h_l) = W^l h_l$, Lipschitz norm of g is $||g||_{Lip} = sup_{h_l} \sigma(W^l) = \sigma(W^l)$ and $\sigma(W^l)$ is the largest singular value of W^l. SNGAN satisfies Lipschitz constraint by dividing all Discriminator's layer by its $\sigma(W^l)$.

$$\bar{W}_{SN}(W^l) := W^l / \sigma(W^l) \tag{7}$$

2.2 Hinge Adversarial Loss

wGAN makes Discriminator build a decision boundary to distinguish between two classes, not probability. This prevents GAN from having gradient vanishing when Discriminator is optimal. However, it is difficult to find an appropriate decision boundary when real and fake samples are not clearly separable. Tran et al. (2017), Lim & Ye (2017) propose hinge adversarial loss with soft margin SVM (Schölkopf et al., 2002), (Cortes & Vladimir, 1995) that is applied to the decision boundary of Discriminator. When designing critic of wGAN, channel of last convolution layer is set to 1. Thus, output of critic is a single $n \times n$ feature map. McGAN (Mroueh et al., 2017), which inspired GeoGAN, fixes output of critic as a single 1×1 feature map, so that the final convolution layer w acts as a weighted sum. The critic of McGAN can be expressed as the product of the final convolution layer v and $\Phi_\zeta(g_\theta(z_i))$, remaining layers of critic excluding v. Loss function of McGAN is as follows.

$$L(v, \zeta, \theta) = \left\langle v, \frac{1}{n} \sum_{i=1}^{n} \Phi_\zeta(x_i) - \frac{1}{n} \sum_{i=1}^{n} \Phi_\zeta(g_\theta(z_i)) \right\rangle \tag{8}$$

where v, ζ, θ are weight parameters of Last conv layer, discriminator, generator respectively. $\langle\ \rangle$ indicates inner product. As a result, final output of McGAN's critic is transformed into an equation comparing mean features of $\Phi_\zeta(x_i)$ and $\Phi_\zeta(g(z_i))$ through weighted sum v. GeoGAN sets last convolution layer w as decision function of SVM. Discriminator learns to maximize margin between two classes, and generator learns to move fake samples toward real class plane. However, because $L(v, \zeta, \theta)$ searches for decision boundary with hard margin, this makes it difficult to search for optimal decision boundary when fake and real samples are intermixed. GeoGAN adapts soft margin SVM to resolve the problem. The primal form of the soft margin SVM is as follows.

$$\min_{v,b} \quad \frac{1}{2}||v||^2 + C\sum\nolimits_i (\xi_i + \xi_i{}') \tag{9}$$

$$\begin{aligned} \text{subject to} \quad & \langle v, \Phi_\zeta(x_i) \rangle + b \geq 1 - \xi_i, && i = 1, \cdots, n \\ & \langle v, \Phi_\zeta(g_\theta(z_i)) + b \leq \xi_i' - 1, && i = 1, \cdots, n \\ & \xi_i, \xi_i' \geq 0, && i = 1, \cdots, n \end{aligned} \tag{10}$$

where ξ_i is value of how far each sample is out of class plane, and C is a hyper-parameter that determines how much classification mistakes are reflected. We can make the problem of optimizing objective function $f(x)$ under the constraint $g(x) = 0$ into an equation through *Lagrange Multiplier* method $F(x, y) = f(x) + y * g(x)$.

$$\begin{aligned} \min_{v,b} \quad & \frac{1}{2Cn}||v||^2 + \frac{1}{n}\sum_{i=1}^{n} max(0, 1 - \langle v, \Phi_\zeta(x_i) \rangle) \\ & + \frac{1}{n}\sum_{i=1}^{n} max(0, 1 + \langle v, \Phi_\zeta(g_\theta(z_i)) \rangle) \end{aligned} \tag{11}$$

Hinge adversarial loss constrains $D(x)$ to greater than 1 and $D(G(z))$ to less than -1. Such constraints ignore values of samples properly classified in each class. However, it gives penalties for

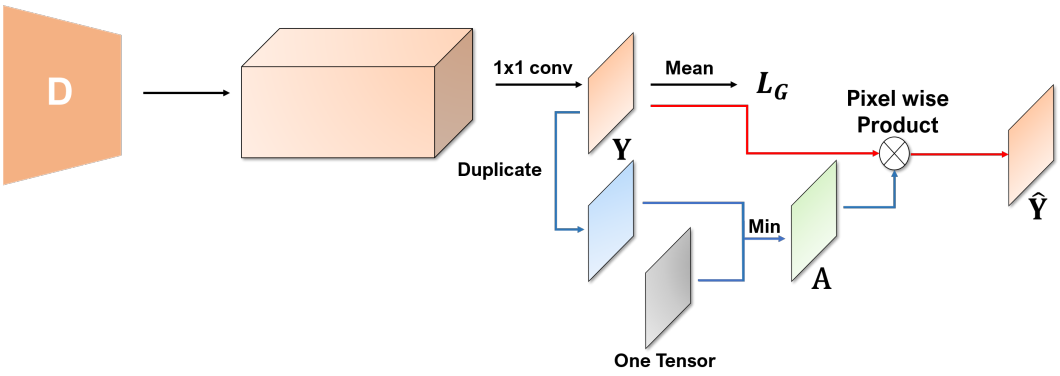

Figure 1: Discriminator with Local Gradient Amplifier (LGA) for the implementation of spatially decomposed hinge adversarial loss

misclassified samples proportional to the distance from the class boundary. This hinge adversarial loss not only makes discriminator more focused on misclassified samples, but also makes the loss easier to control than conventional adversarial loss by setting the decision boundary to 0.

3 Proposed Method

Our proposed spatially decomposed hinge adversarial loss is defined as follows.

$$\begin{aligned} L_D &= - \mathop{\mathbb{E}}_{x \sim P_r} [min(0, -1 + D(x))] - \mathop{\mathbb{E}}_{z \sim P_z} [min(0, -1 - D(G(z))] \\ L_G &= - \mathop{\mathbb{E}}_{z \sim P_z} [D_{i,j}(G(z))] \approx - \mathop{\mathbb{E}}_{z \sim P_z} [D_{LGA}(G(z))] \end{aligned} \tag{12}$$

where discriminator loss L_D is identical to the traditional hinge adversarial loss for GAN. In generator loss L_G, $D_{i,j}$ represents discriminator output calculated from $(i,j)th$ feature value of the final feature map $Y \in \mathbb{R}^{n \times n}$. we approximate $D_{i,j}$ by D_{LGA} where LGA represents our Local Gradient Amplifier as illustrated in Figure 1. In the back-propagation process of traditional GANs, gradients are equally propagated backward from the feature map of discriminator. In this case, entire spatial regions of the generator that created current fake images will be updated without any spatial priority and, as a result, regardless of whether particular spatial region in the generator contributed to the unrealistic part of the fake image or not. The output value of discriminator can be either a simple scalar value of 1×1 or a feature map of $n \times n$. If the last layer of discriminator is a 1×1 convolution layer with a single filter, output will be a single $n \times n$ feature map. We focus on this last layer v that operates as a decision function determining real or fake locally. By using v, we implement our Local Gradient Amplifier.

D_{LGA} in the calculation of L_G, we build an amplification map $A \in \mathbb{R}^{n \times n}$ by duplicating final feature map $Y \in \mathbb{R}^{n \times n}$. And we adjust A so that a value greater than 1 becomes 1.

$$A_{i,j} = min(Y_{i,j}, 1) \tag{13}$$

We perform pixel-wise multiplication with amplification map and original final feature map.

$$\hat{Y} = Y \odot A \tag{14}$$

Note that $\hat{Y}$ is created to calculate only L_G and train generator. Based on the amplification, gradient of Y is calculated as follows, with which we amplify gradients that propagate to locally unrealistic area in a generated image.

$$\frac{dL_G}{dY_{i,j}} = \frac{dL_G}{dY} \times A_{i,j} \tag{15}$$

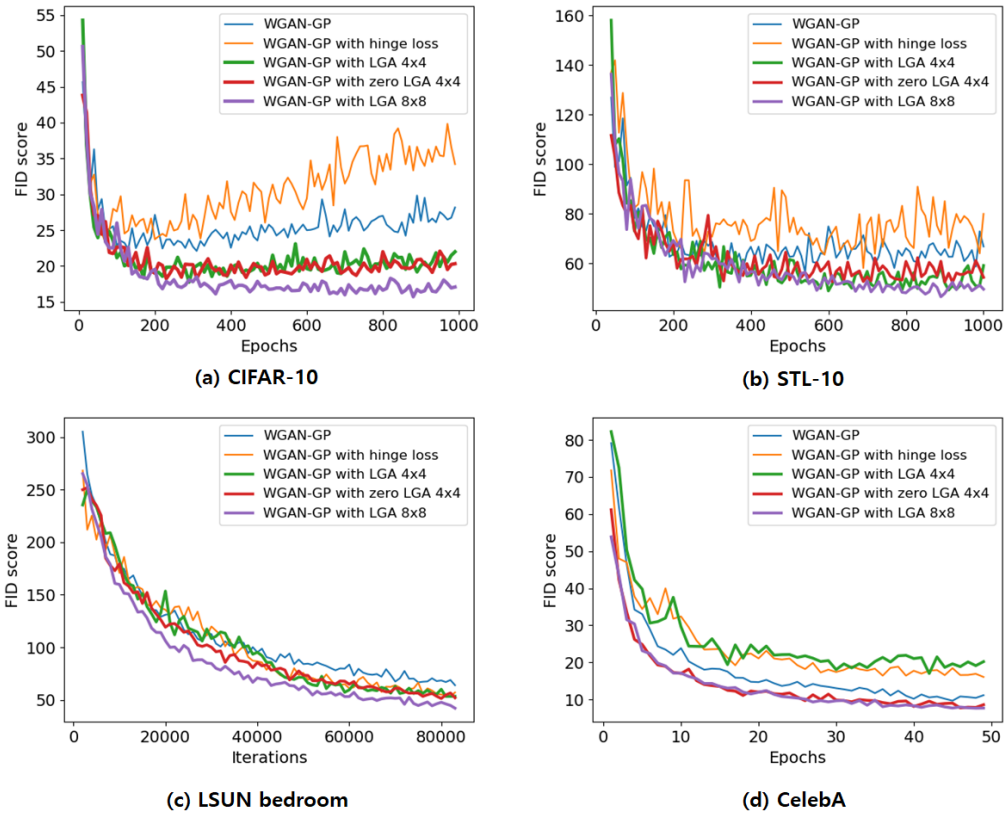

Figure 2: FID scores on various data sets. All trainings are conducted with WGAN-GP and RMSProp. The resolutions of generated images are 32×32 for CIFAR10 and STL10 and 64×64 for CelebA and LSUN bedroom.

4 Experimental Evaluation

4.1 Implementation Details

We conduct experimental evaluation on various benchmark data sets; CIFAR-10, STL-10, CelebA, and LSUN bedroom. For LSUN bedroom and CelebA, we set generator to make 64×64 images. For CIFAR-10 and STL-10, 32×32 images are generated. We use Adam optimizer with $\beta_1 = 0, \beta_2 = 0.9$. We have observed that learning process becomes unstable when momentum decay term β_1 is large, so we only use RMSProp term in our tests. We use gradient penalty and spectral normalization to satisfy Lipschitz constraint. Weight clipping is used only to reproduce GeoGAN(Lim & Ye, 2017). The hyperparameter λ of gradient penalty was set to 10 in default WGAN-GP only, and 1 for other experiments. WGAN-GP with LGA 4×4 and 8×8 mean final output of critic is set to 4×4 and 8×8 feature map, respectively. We apply our LGA to these final feature maps. WGAN-GP with zero LGA 4×4 means an experiment in which gradient, which is propagated to filters of Generator that contributed realistic area, is completely blocked by setting $A_{i,j} = min(Y_{i,j}, 0)$. For spectral normalization, we experiment only LGA 4×4. Resnet based SNGAN makes $\Phi_\zeta(x)$ which is input of last layer v into a $c \times 1 \times 1$ feature through global sum pooling. We apply 2×2 average pooling with stride 2 to apply LGA, maintaining the role of global sum pooling of SNGAN. To maintain the role of global sum pooling of SNGAN, we multiply the output of average pooling by 64. For quantitative evaluation, we calculate Fréchet inception distance (Heusel et al., 2017) and Inception score. Experiments are performed with 10k real samples and 10k fake samples except experiments on STL-10. For STL-10, we evaluate networks with 5k real samples and 10k fake samples due to the lack of training data. Table (1) and (2) summarize all of our experimental results.

Table 1: FID score comparison on four benchmark data sets

Method	CIFAR-10	STL-10	CelebA	LSUN bedroom
GeoGAN	36.14	49.95	13.96	48.58
wGAN-GP	22.59	55.40	11.15	64.01
WGAN-GP + hinge loss	23.30	69.72	16.07	57.12
WGAN-GP + LGA 4×4	19.31	43.41	18.85	53.46
WGAN-GP + zero LGA 4×4	20.20	45.61	8.59	52.08
WGAN-GP + LGA 8×8	**16.61**	**35.83**	**7.68**	**42.17**

Table 2: Inception score and FID score of SNGAN based evaluation

	CIFAR-10		STL-10	
Method	Inception score	FID	Inception score	FID
SNGAN	8.3	**17.33**	7.2	36.05
SNGAN + LGA	**8.5**	18.16	**7.3**	**34.02**

4.2 Results on Cifar-10 and Stl-10

We conduct experiments on CIFAR-10 and STL-10 with WGAN-GP based networks for 100 epochs. Figure 2 (a), (b) show FID scores on CIFAR-10 and STL-10. We measure FID score every 10 epochs. For generating 32×32 image, after around 50 epochs, networks without LGA do not decrease the score as the training progressed. On the other hand, networks with LGA show better and stable FID score decrease as the number of epochs increased. In Figure 2, prior methods diverge after they show converging curves on Cifar10. We calculate average and std of FID scores within 20% of lowest FID from respective lowest FID score positions for Stl-10: wGAN-GP(64.6%(3.6)), wGAN-GP with hinge(74.0%(6.6)), LGA 4×4(54.6%(3.8)), LGA zero4×4(58.3%(3.3)), LGA 8×8(52.5%(3.4)). After 200 epochs, some tests without LGA look to have mode collapse. These mode collapses are found not only with WGAN-GP based previous networks but also in SNGAN based previous networks. Figure 3 shows both mode collapsed and unstable sample results appeared over 100 epochs. Figure 5 shows results after 1000 epochs. WGAN-GP with hinge loss produces blurry images, but other networks with LGA generate clean and diverse samples at 1000 epochs. (You can check the occurrence of mode collapse at Figure 6 in appendix.)

4.3 Results on CelebA and LSUN bedroom

For LSUN bedroom, we train our networks 2 epochs, and measure FID score every 1000 iterations. And for CelebA, we train our networks 50 epochs, and measure FID score every epoch. Figure 2 (c), (d) show FID score on LSUN bedroom and CelebA. For generating 64×64 image, we observe the networks with LGA converge faster than others. In training LSUN bedroom, WGAN-GP with LGA 8×8 takes 21000 iterations until it reach 100 FID score, but WGAN-GP takes 37000 iterations. In training CelebA, WGAN-GP converges 5 epochs slower than WGAN-GP with LGA 8×8. Figure 4 shows generated image samples. WGAN-GP with LGA for both data sets shows faster convergence with higher diversity in generated images.

4.4 Results on Spectral Normalization

SNGAN(Miyato et al., 2018) shows remarkable result on CIFAR-10 and STL-10 through resnet based networks. We incorporate LGA 4×4 to the networks. Table 2 shows quantitative evaluation results of original SNGAN and SNGAN with LGA 4×4. In these experiment we evaluate using Inception Score(Salimans et al., 2016) and FID. For CIFAR-10, we train our networks 50000 iterations to reproduce SNGAN's results. For STL-10, we train our networks only 10000 iterations,

because we observe mode collapse after 20000 iterations on original SNGAN. Figure 6 in appendix shows mode collapse on STL-10. We generates 32×32 images both CIFAR-10 and STL-10.

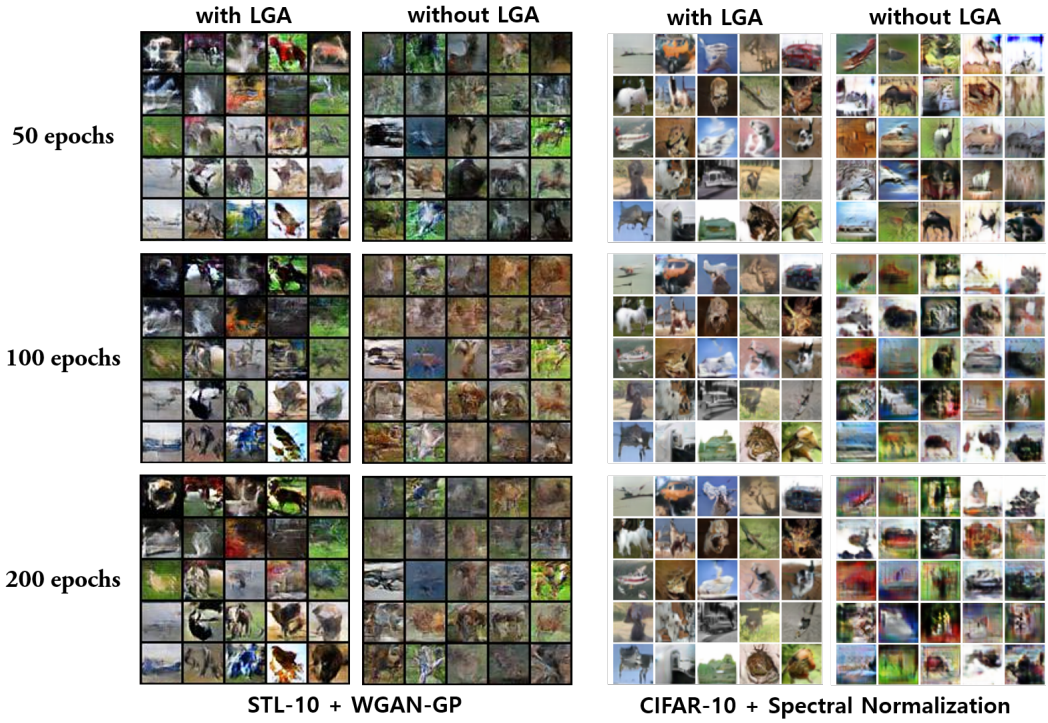

Figure 3: Intermediate results of Training with STL10 and CFIAR10. When training highly complex dataset without LGA, Generator often produces patterns which look like mode collapse.

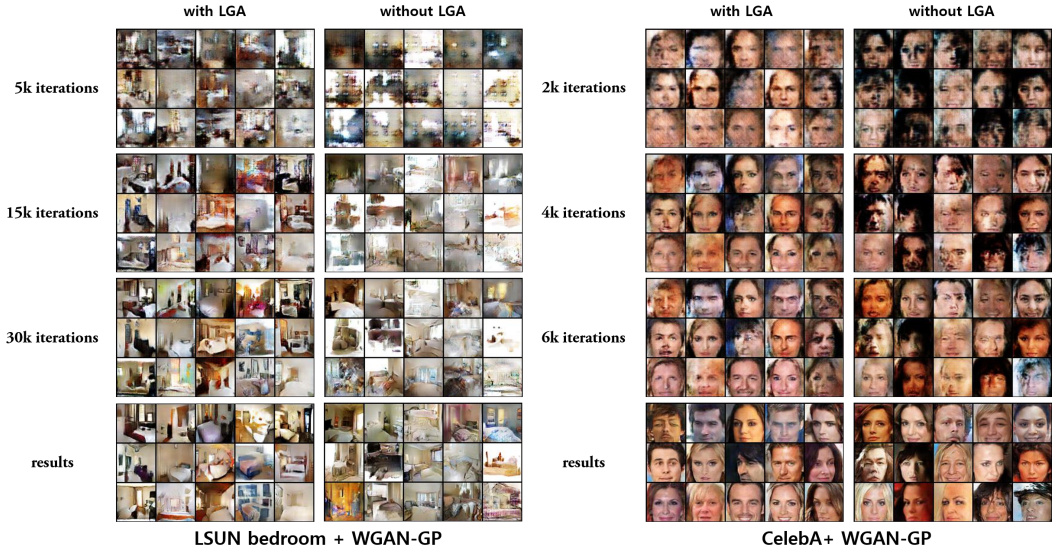

Figure 4: Intermediate results of Training with LSUN bedroom and CelebA. When generating 64×64 images, LGA shows much faster convergence than previous methods.

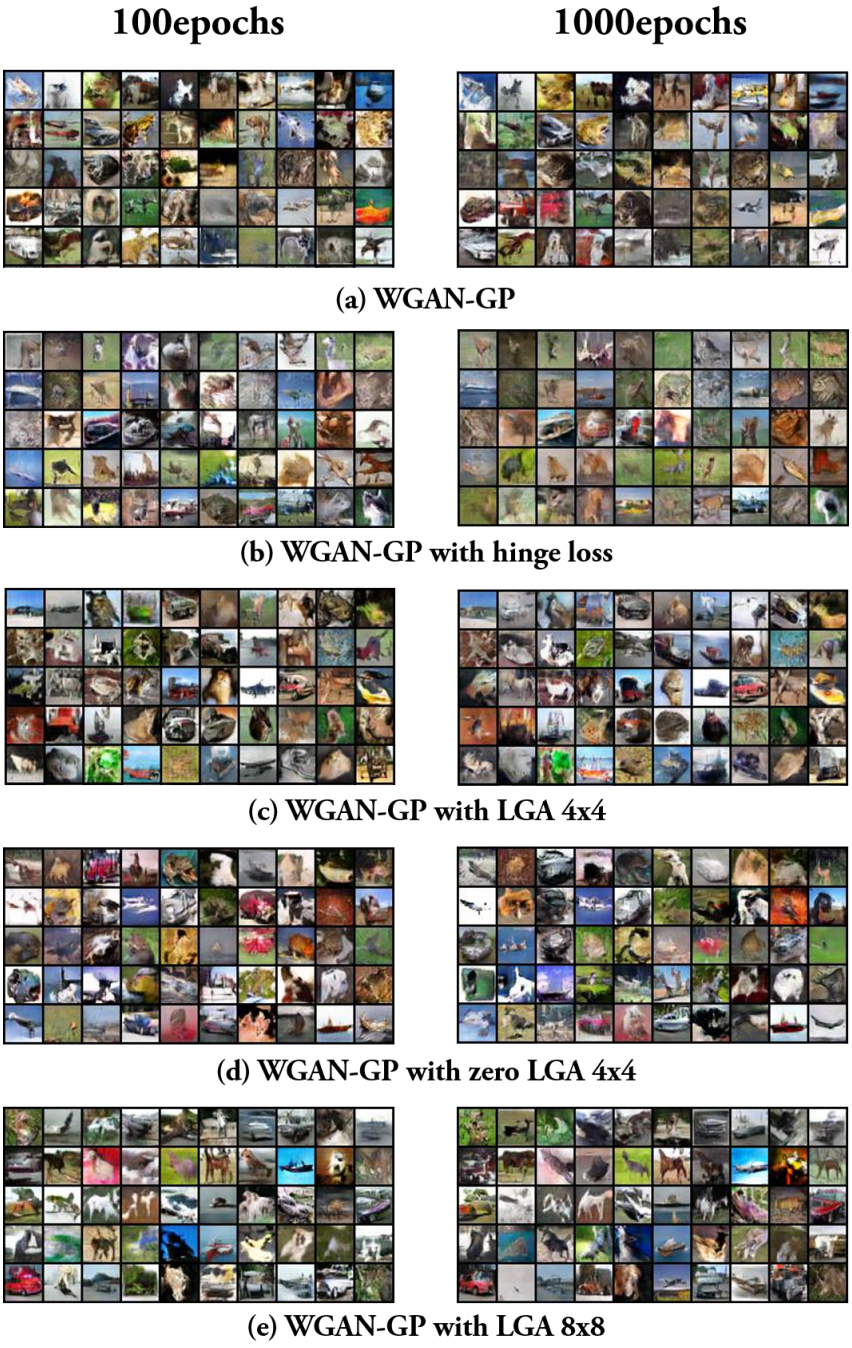

Figure 5: Generated images with WGAN-GP based network on CIFAR-10. Without LGA, Generator produces image with poor representation.

5 Conclusion

In this paper, we propose Local Gradient Amplifier(LGA) which realizes spatially decomposed hinge adversarial loss for improved generator training. Structure of LGA is simple and easy to adopt in various networks. By using LGA, we can train GAN more quickly and stably.

REFERENCES

Martin Arjovsky, Chintala Soumith, and Bottou Léon. Wasserstein gan. arxiv:1701.07875, 2017.

Corinna Cortes and Vapnik Vladimir. Support-vector networks. *Machine learning*, 20.3:273–297, 1995.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. NIPS, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. NIPS, 2017.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. NIPS, 2017.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. CVPR, 2017.

Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. CVPR, 2017.

JaeHyun Lim and JongChul Ye. Geometric gan. arxiv:1705.02894, 2017.

Takero Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. ICLR, 2018.

Youssef Mroueh, Tom Sercu, and Vaibhava Goel. Mcgan: Mean and covariance feature matching gan. ICML, 2017.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semanitc image synthesis with spatially-adaptive normalization. CVPR, 2019.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. NIPS, 2016.

Bernhard Schölkopf, Alexander J. Smola, and Francis Bach. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT Press, 2002.

Dustin Tran, Rajesh Ranganath, and David M. Blei. Deep and hierarchical implicit models. arXiv:1702.08896, 2017.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. CVPR, 2018.

Han Zhang, Tao Xu, Hongsheng Li, shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitirs N. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. CVPR, 2017.

A ADDITIONAL EXPERIMENT RESULTS

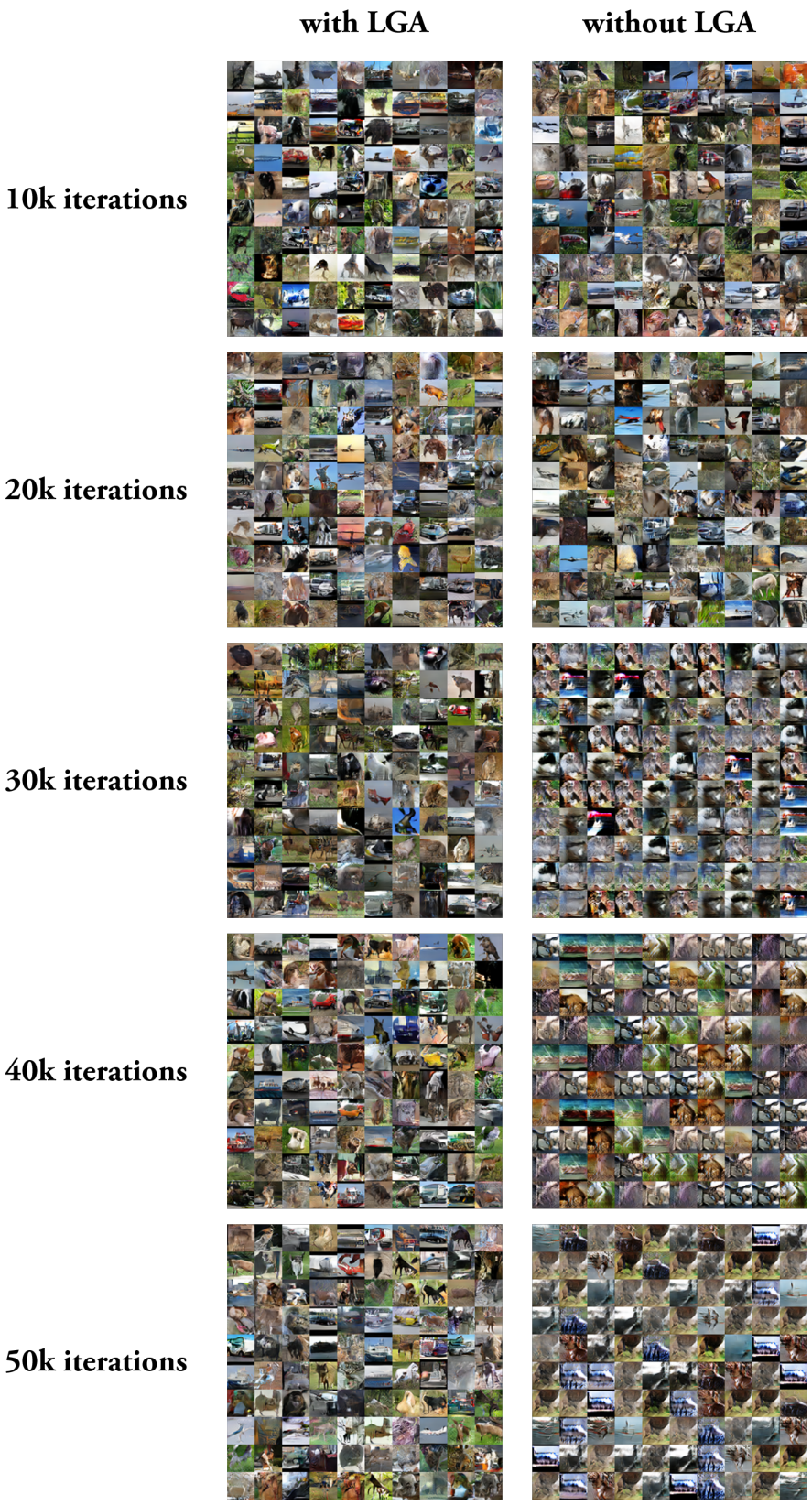

Figure 6: Generated images with SNGAN based network on STL-10. We can see original SNGAN occurs mode collapse when training with long iterations.

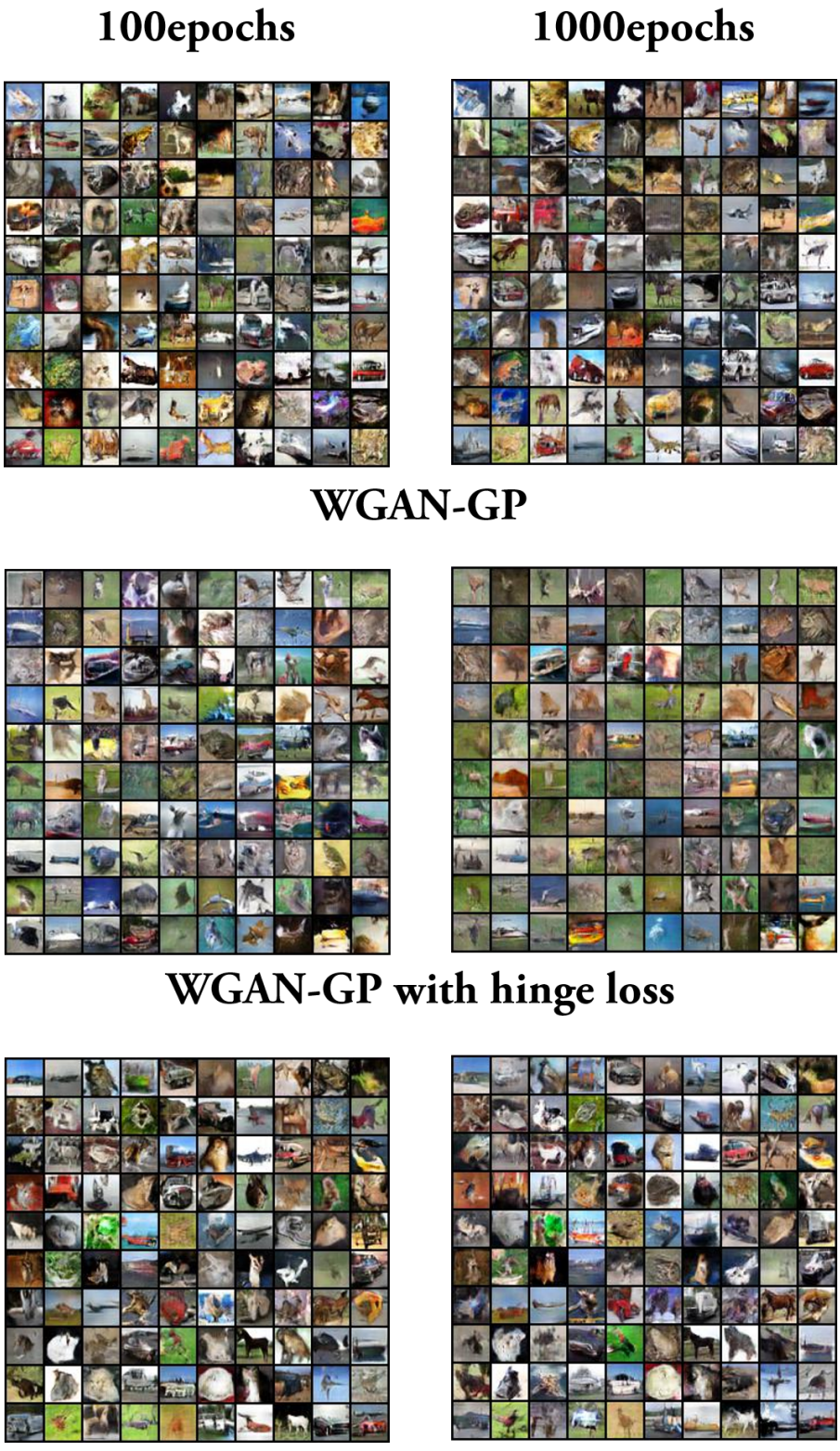

Figure 7: Generated images with WGAN-GP based network on CIFAR-10. For CIFAR-10, we train our networks with 1000 epochs.

Figure 7: Generated images with WGAN-GP based network on CIFAR-10. For CIFAR-10, we train our networks with 1000 epochs.

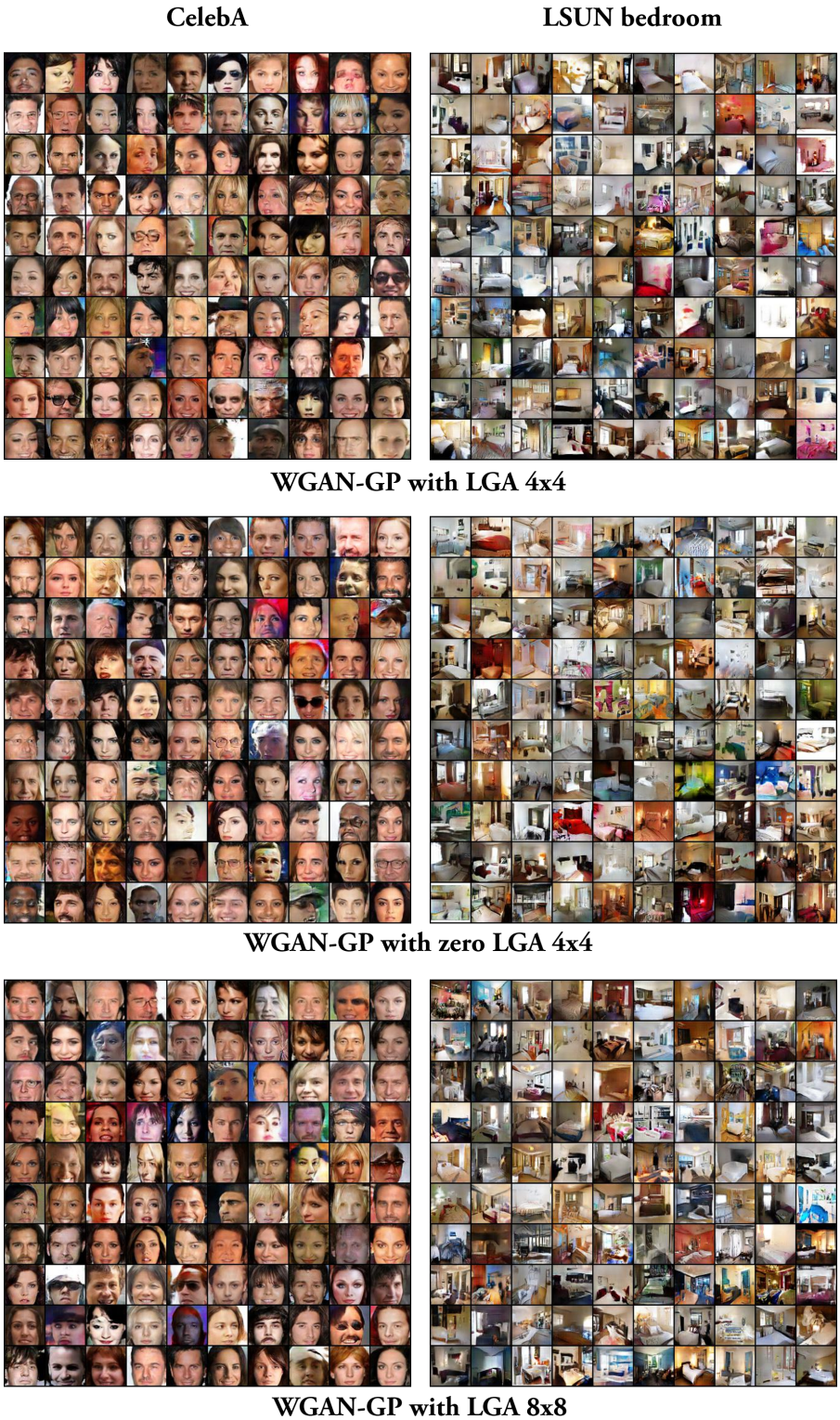

Figure 8: Generated images with WGAN-GP based network on CelebA and LSUN bedroom. For CelebA, we train our networks with 50 epochs. For LSUN bedroom, we train our networks with 2 epochs.