Progressive Growing of GANs for Improved Quality, Stability, and Variation

Tero Karras
NVIDIA
tkarras@nvidia.com

Timo Aila
NVIDIA
taila@nvidia.com

Samuli Laine
NVIDIA
slaine@nvidia.com

Jaakko Lehtinen
NVIDIA and Aalto University
jlehtinen@nvidia.com

arXiv YouTube GitHub

Introduction

Problem

- GANs become seriously unstable as the output resolution increases
- How to enable high-quality image synthesis at megapixel resolutions?

Approach

- New training methodology
- Grow both the generator and discriminator progressively
- Add new layers to model increasingly fine details as training progresses
- Several tweaks to increase variation and avoid mode collapse
- New metric for assessing result quality

Benefits

- Considerably faster and more stable training, especially at high resolutions
- Able to produce images of unprecedented quality at 1024×1024
- Achieves record inception score of 8.80 in unsupervised CIFAR10

References

Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. *CoRR*, abs/1707.09405, 2017.

Zihang Dai, Amjad Almahairi, Philip Bachman, Eduard H. Hovy, and Aaron C. Courville. Calibrating energy-based generative adversarial networks. In *ICLR*, 2017.

Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martin Arjovsky, Olivier Mastropietro, and Aaron Courville. Adversarially learned inference. *CoRR*, abs/1606.00704, 2016.

Ishan P. Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative multi-adversarial networks. *CoRR*, abs/1611.01673, 2016.

Guillermo L. Grinblat, Lucas C. Uzal, and Pablo M. Granitto. Class-splitting generative adversarial networks. *CoRR*, abs/1709.07359, 2017.

Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of Wasserstein GANs. *CoRR*, abs/1704.00028, 2017.

Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, and Zhen Wang. Least squares generative adversarial networks. *CoRR*, abs/1611.04076, 2016.

Marco Marchesi. Megapixel size image creation using generative adversarial networks. *CoRR*, abs/1706.00082, 2017.

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

David Warde-Farley and Yoshua Bengio. Improving generative adversarial networks with denoising feature matching. In *ICLR*, 2017.

Jianwei Yang, Anitha Kannan, Dhruv Batra, and Devi Parikh. LR-GAN: layered recursive generative adversarial networks for image generation. In *ICLR*, 2017.

Contributions

Progressive growing: *stabilize training and speed up convergence*

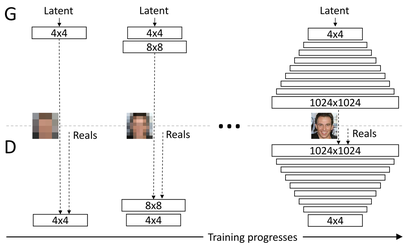


Figure 1: We start with a low resolution of 4×4 pixels. As the training advances, we incrementally introduce new layers to increase the resolution.

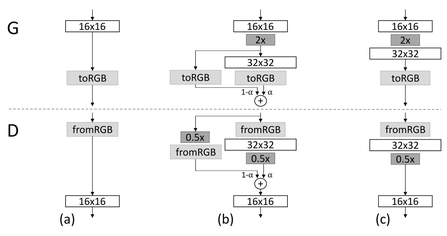


Figure 2: When doubling the resolution, we fade in the new layers smoothly by treating them like a residual block, whose weight increases linearly from 0 to 1.

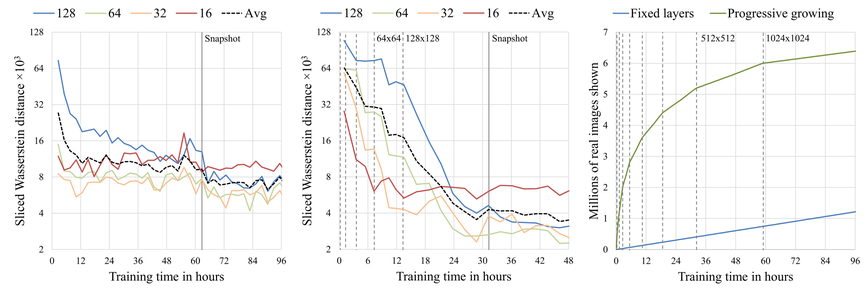


Figure 3: *Left:* Statistical similarity over time for different scales using the CELEBA dataset. *Center:* Progressive growing speeds up the convergence. *Right:* Raw training speed at 1024×1024 using Tesla P100.

Minibatch standard deviation: *improve variation*

- Special layer in discriminator measures variation across minibatch
- Discourages generator from producing too homogeneous results

Equalized learning rate: *make layers learn at same pace*

- Initialize weights to unit variance, re-scale at runtime
- Effective learning rate becomes independent of layer dimensions

Pixelwise feature vector normalization: *avoid collapse*

- Normalize generator activations to unit length at each pixel
- Prevents generated pixel values from veering off to infinity

Sliced Wasserstein distance (SWD): *assess quality*

- Take small 7×7 pixel patches from generated & training images
- Compare distributions on multiple scales (full res., half res., ...)

CELEBA-HQ: *novel dataset with 30,000 high-res images*

CELEBA in-the-wild | De-JPEG, super-res. | Mirror padding | Gaussian filter | Landmark alignment | Final image 1024×1024

Generator	Act.	Output shape	Params
Latent vector	–	512 × 1 × 1	–
Conv 4 × 4	LReLU	512 × 4 × 4	4.2M
Conv 3 × 3	LReLU	512 × 4 × 4	2.4M
Upsample	–	512 × 8 × 8	–
Conv 3 × 3	LReLU	512 × 8 × 8	2.4M
Conv 3 × 3	LReLU	512 × 8 × 8	2.4M
⋮	⋮	⋮	⋮
Upsample	–	64 × 512 × 512	–
Conv 3 × 3	LReLU	32 × 512 × 512	18k
Conv 3 × 3	LReLU	32 × 512 × 512	9.2k
Upsample	–	32 × 1024 × 1024	–
Conv 3 × 3	LReLU	16 × 1024 × 1024	4.6k
Conv 3 × 3	LReLU	16 × 1024 × 1024	2.3k
Conv 1 × 1	linear	3 × 1024 × 1024	51
Total trainable parameters			**23.1M**

Discriminator	Act.	Output shape	Params
Input image	–	3 × 1024 × 1024	–
Conv 1 × 1	LReLU	16 × 1024 × 1024	64
Conv 3 × 3	LReLU	16 × 1024 × 1024	2.3k
Conv 3 × 3	LReLU	32 × 1024 × 1024	4.6k
Downsample	–	32 × 512 × 512	–
Conv 3 × 3	LReLU	32 × 512 × 512	9.2k
Conv 3 × 3	LReLU	64 × 512 × 512	18k
Downsample	–	64 × 256 × 256	–
⋮	⋮	⋮	⋮
Conv 3 × 3	LReLU	512 × 8 × 8	2.4M
Conv 3 × 3	LReLU	512 × 8 × 8	2.4M
Downsample	–	512 × 4 × 4	–
Minibatch stddev	–	513 × 4 × 4	–
Conv 3 × 3	LReLU	512 × 4 × 4	2.4M
Conv 4 × 4	LReLU	512 × 1 × 1	4.2M
Fully-connected	linear	1 × 1 × 1	513
Total trainable parameters			**23.1M**

Figure 4: Full-resolution generator and discriminator that we use to generate 1024×1024 images. Both networks consist mainly of replicated 3-layer blocks that we introduce one by one during the course of the training.

Results

Figure 5: 1024×1024 images generated using the CELEBA-HQ dataset. While megapixel GAN results have been shown before (Marchesi, 2017), our results are vastly more varied and of higher perceptual quality.

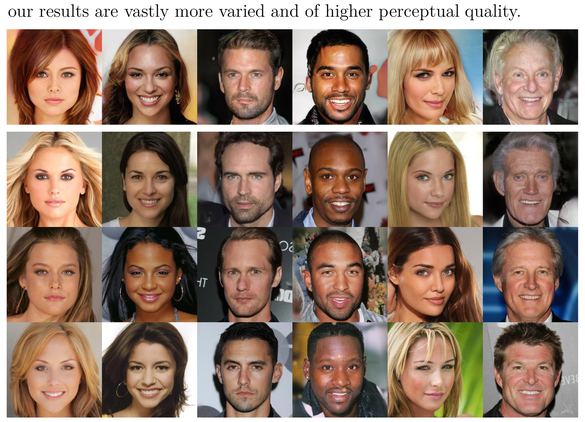
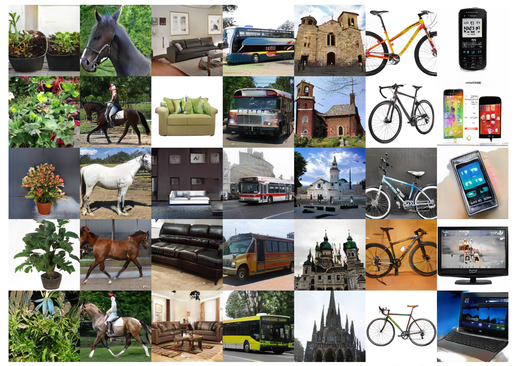


Figure 6: *Top:* Generated images. *Next three rows:* Nearest neighbors found in the training data, based on feature-space distance (Chen & Koltun, 2017). The generator does not appear to overfit to the data.

Mao et al. (2016) (128 × 128) | Gulrajani et al. (2017) (128 × 128) | Our (256 × 256)

Figure 7: Visual quality comparison between our solution and earlier techniques in the LSUN BEDROOM dataset. Pictures copied from the cited articles.



POTTED-PLANT | HORSE | SOFA | BUS | CHURCH-OUTDOOR | BICYCLE | TV-MONITOR

Figure 8: Selection of 256×256 images generated from different LSUN categories. We performed the training separately for each category using the first 100,000 images.

Training configuration	Sliced Wasserstein distance ×10³					MS-SSIM
	128	64	32	16	Avg	
(a) Gulrajani et al. (2017)	12.99	7.79	7.62	8.73	9.28	0.2854
(b) + Progressive growing	4.62	**2.64**	3.78	6.06	4.28	**0.2838**
(c) + Small minibatch	75.42	41.33	41.62	26.57	46.23	0.4065
(d) + Revised training parameters	9.20	6.53	4.71	11.84	8.07	0.3027
(e*) + Minibatch discrimination	10.76	6.28	6.04	16.29	9.84	0.3057
(e) Minibatch stddev	13.94	5.67	2.82	5.71	7.04	0.2950
(f) + Equalized learning rate	4.42	3.28	2.32	7.52	4.39	0.2902
(g) + Pixelwise normalization	**4.06**	3.04	**2.02**	**5.13**	**3.56**	0.2845
(h) Converged	2.42	2.17	2.24	4.99	2.96	0.2828

Figure 9: SWD and MS-SSIM for several training setups using CELEBA at 128×128. For SWD, each column represents one scale level and the last one is their average.

(a) (b) (c) (d) (e*) (e) (f) (g) (h)

Figure 10: Example images corresponding to rows in Fig. 9. Note that (a)-(g) are intentionally non-converged.

Method		Inception score
ALI	(Dumoulin et al., 2016)	5.34 ± 0.05
GMAN	(Durugkar et al., 2016)	6.00 ± 0.19
Improved GAN	(Salimans et al., 2016)	6.86 ± 0.06
CEGAN-Ent-VI	(Dai et al., 2017)	7.07 ± 0.07
LR-AGN	(Yang et al., 2017)	7.17 ± 0.17
DFM	(Warde-Farley & Bengio, 2017)	7.72 ± 0.13
WGAN-GP	(Gulrajani et al., 2017)	7.86 ± 0.07
Splitting GAN	(Grinblat et al., 2017)	7.90 ± 0.09
Our (best run)		**8.80 ± 0.05**
Our (computed from 10 runs)		**8.56 ± 0.06**

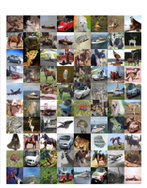


Figure 11: Unsupervised inception scores using CIFAR10, higher is better. We achieve a record score of 8.80.