LARGE SCALE GAN TRAINING FOR HIGH FIDELITY NATURAL IMAGE SYNTHESIS

Mikhailov Nikita, 161

Higher School of Economics

30.01.2020

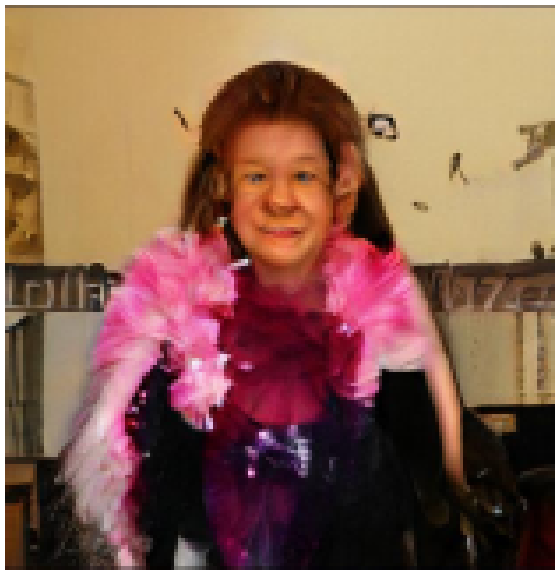
State-of-the-art GAN

Self-Attention Generative Adversarial Networks (SA-GAN)

Inception Score=52.52 (больше – лучше)
Frechet Inception Distance=18.65 (меньше – лучше)

Что хочется улучшить?

- Метрики
- Разрешение изображений
- Качетво генерирования мелких деталей



Лицо человека

Подходы

- Увеличение BatchSize
- Увеличение ширины сети

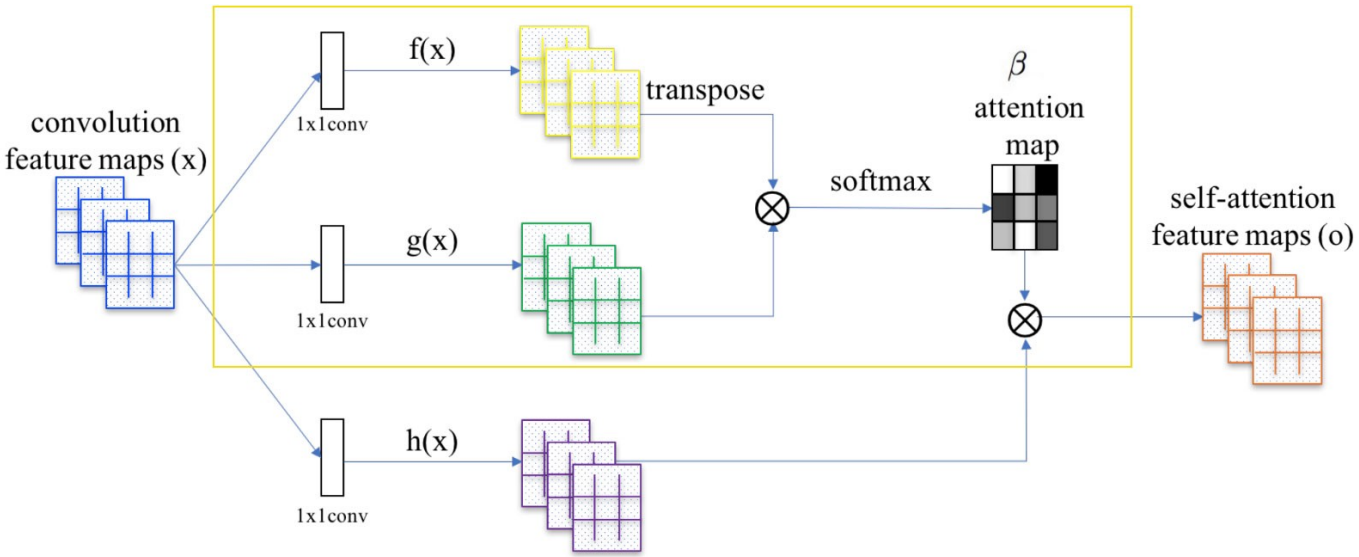
Batch	Ch.	Param (M)	Shared	Skip-z	Ortho.	Itr $\times 10^3$	FID	IS
256	64	81.5	SA-GAN Baseline			1000	18.65	52.52
512	64	81.5	✗	✗	✗	1000	15.30	58.77($\pm$1.18)
1024	64	81.5	✗	✗	✗	1000	14.88	63.03($\pm$1.42)
2048	64	81.5	✗	✗	✗	732	12.39	76.85($\pm$3.83)
2048	96	173.5	✗	✗	✗	295($\pm$18)	9.54($\pm$0.62)	92.98($\pm$4.27)

SA-GAN

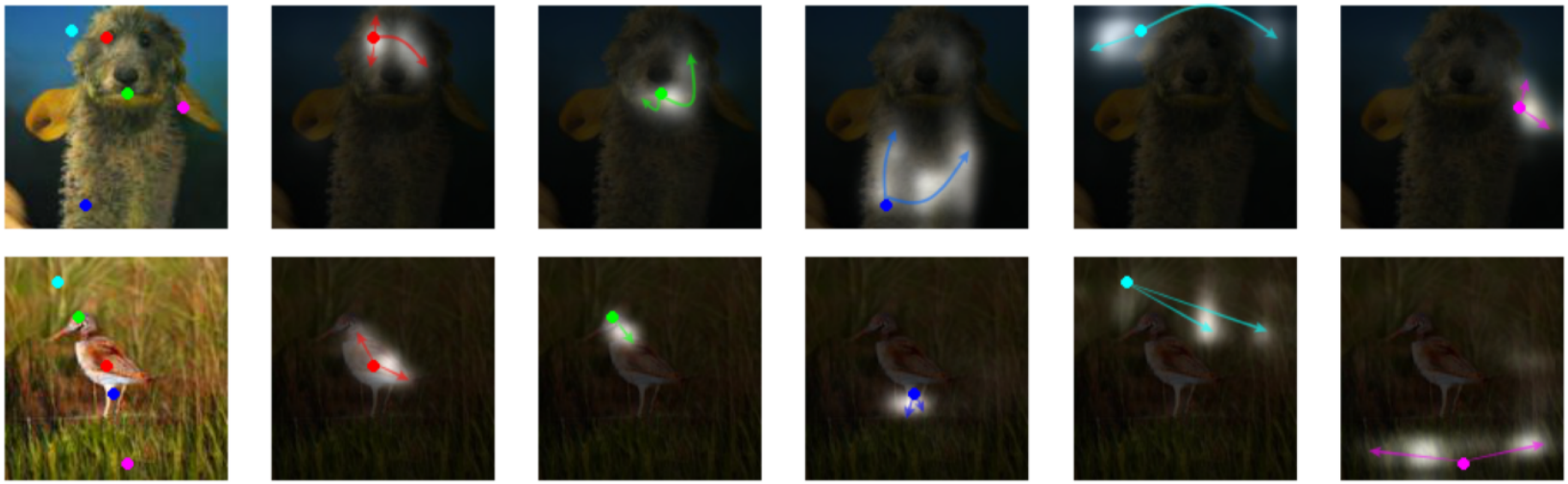
- self-attention для изображений
- conditional batch normalization
- Информация о классе в D и G
- Hinge-loss в GAN

Self-Attention для изображений

- $f : [C, HW], g : [C, HW], h : [C_0, HW]$
- $m = f^T \cdot g : [HW, HW] \rightarrow softmax(m)$
- $feature\_map = h \cdot m : [C_0, HW]$
- $Attention = \gamma \cdot feature\_map + x$



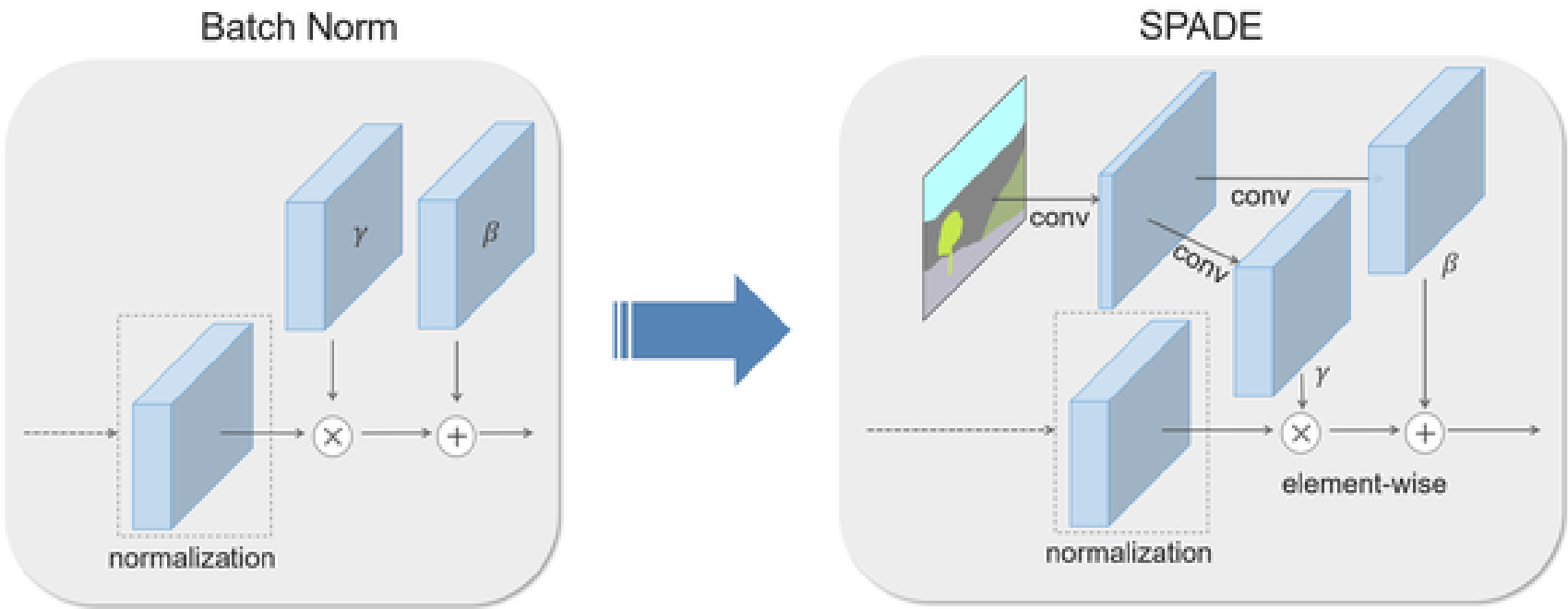
Self-Attention для изображений



Conditional Batch-Normalization

$$x_i \to \gamma_i \hat{x}_i + \beta_i$$

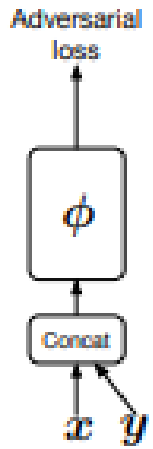
Теперь γ и β функции от x



ImageNet

GAN учится на ImageNet датасете

- Вход $x = [z \sim P(z), OneHot(y)]$
- Генератор знает, какой класс он должен выдать
- Дискриминатор имеет полное представление о том, что на изображении



HingeLoss

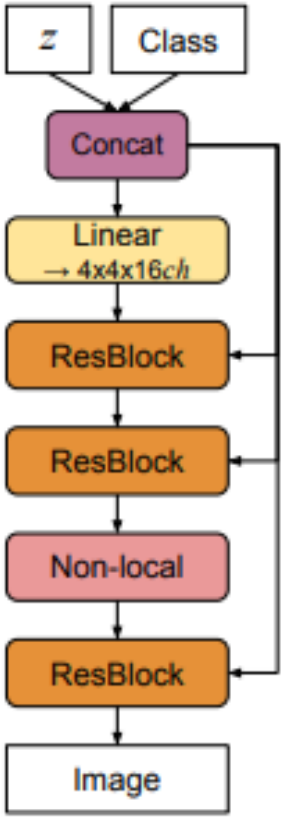
SA-GAN использует такой лосс для дискриминатора и генератора:

$$L_D = -\mathbb{E}_{(x,y)\sim p_{data}}\left[min(0, -1+D(x,y)\right] - \mathbb{E}_{z\sim p_z y\ \sim p_{data}}\left[min(-1-D(G(z,y))\right]$$

$$L_G = -\mathbb{E}_{z\sim p_z y\ \sim p_{data}}\left[D(G(z,y)\right]$$

Улучшения от авторов статьи

- Вместо OneHot(y) использовать эмбеддинги классов.
  Таким образом можно сильно уменьшить число параметров
- $OneHot(y) \in \mathbb{R}^{1000}$, а $Embed(y) \in \mathbb{R}^{128}$
- "Проброс"исходной переменной z в середину сети
- Truncation Trick
- Orthogonal Regularization при отсутствии результата с Truncation Trick



Truncation Trick

Семплируем переменную z из $\mathbb{N}(0, I)$. Если значение выпадает за некоторые пределы, то семплируем заново

- Приводит к улучшению всех метрик
- Уменьшает вероятность генерирования выбросов
- Но понижает вариативность

Orthogonal Regularization

Идея в том, чтобы сделать G гладким, что все из нашего распределения имело хорошие изображения

$$R_\beta(W) = \beta||WW^T - I||_F^2$$

W – матрица весов, β – гиперпараметр

Orthogonal Regularization

Но это слишком сильное ограничение, поэтому авторы статьи придумали это:

$$R_\beta(W) = \beta||WW^T \odot (1 - I)||_F^2$$

Минимизирует попарное косинусное расстояние между фильтрами

Результаты с нововведениями

Batch	Ch.	Param (M)	Shared	Skip-z	Ortho.	Itr $\times 10^3$	FID	IS
256	64	81.5	SA-GAN Baseline			1000	18.65	52.52
512	64	81.5	✗	✗	✗	1000	15.30	58.77($\pm$1.18)
1024	64	81.5	✗	✗	✗	1000	14.88	63.03($\pm$1.42)
2048	64	81.5	✗	✗	✗	732	12.39	76.85($\pm$3.83)
2048	96	173.5	✗	✗	✗	295($\pm$18)	9.54($\pm$0.62)	92.98($\pm$4.27)
2048	96	160.6	✓	✗	✗	185($\pm$11)	9.18($\pm$0.13)	94.94($\pm$1.32)
2048	96	158.3	✓	✓	✗	152($\pm$7)	8.73($\pm$0.45)	98.76($\pm$2.84)
2048	96	158.3	✓	✓	✓	165($\pm$13)	8.51($\pm$0.32)	99.31($\pm$2.10)
2048	64	71.3	✓	✓	✓	371($\pm$7)	10.48($\pm$0.10)	86.90($\pm$0.61)

Model	Res.	FID/IS	(min FID) / IS	FID / (valid IS)	FID / (max IS)
SN-GAN	128	27.62/36.80	N/A	N/A	N/A
SA-GAN	128	18.65/52.52	N/A	N/A	N/A
BigGAN	128	$8.7 \pm .6/98.8 \pm 3$	$7.7 \pm .2/126.5 \pm 0$	$9.6 \pm .4/166.3 \pm 1$	$25 \pm 2/206 \pm 2$
BigGAN	256	$8.7 \pm .1/142.3 \pm 2$	$7.7 \pm .1/178.0 \pm 5$	$9.3 \pm .3/233.1 \pm 1$	$25 \pm 5/291 \pm 4$
BigGAN	512	8.1/144.2	7.6/170.3	11.8/241.4	27.0/275
BigGAN-deep	128	$5.7 \pm .3/124.5 \pm 2$	$6.3 \pm .3/148.1 \pm 4$	$7.4 \pm .6/166.5 \pm 1$	$25 \pm 2/253 \pm 11$
BigGAN-deep	256	$6.9 \pm .2/171.4 \pm 2$	$7.0 \pm .1/202.6 \pm 2$	$8.1 \pm .1/232.5 \pm 2$	$27 \pm 8/317 \pm 6$
BigGAN-deep	512	7.5/152.8	7.7/181.4	11.5/241.5	39.7/298

Картинки

Figure 5: Samples generated by our BigGAN model at 256×256 resolution.

Вопросы

- Как работает Conditional Batch-Normalization?
- Напишите HingeLoss для обучения генератора и дискриминатора
- Что такое Truncation Trick?