

Towards Faster and Stabilized GAN Training for High-fidelity Few-shot Image Synthesis

Ильяс Адыгамов

НИУ Высшая школа экономики

ishadygamov@edu.hse.ru

11 мая 2021 г.

План

- 1 О чём эта работа?
- 2 Проблемы предъявляемых требований
- 3 Идея решения
 - Генератор
 - SLE
 - Self-supervised discriminator
 - Формулы
 - Архитектура
 - Про реализацию
- 4 Результаты
 - FID
 - Сравниваемые модели
 - Few-shot генерация
 - Большие датасеты
 - Визуальное сравнение
- 5 Критика
- 6 Заключение

О чём эта работа?

Проблема:

- Обучение GAN'ов требует (a) много вычислительных ресурсов и (b) много данных и это количество пропорционально разрешению создаваемых изображений.

One-sentence Summary

"converge on single gpu with few hours' training, on 1024 resolution sub-hundred images"

Красивая картинка



Рис.: Результаты на разрешении 1024^2 модели, обученной на одной RTX 2080-Ti GPU, используя 1000 изображений. Слева: 20 часов на пейзажах; Справа: 10 часов на FFHQ.

Проблемы предъявляемых требований

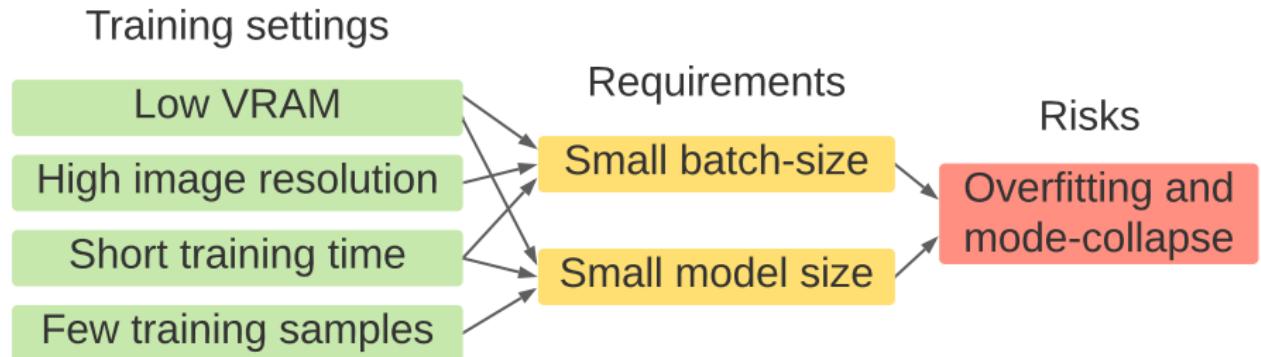


Рис.: Взаимосвязь предъявляемых требований, причин и возникающих сложностей.

Mode collapse



Рис.: Пример mode collapse.

Идея решения

Решение основано на двух идеях:

- Skip-Layer channel-wise Excitation(SLE) - очень похоже на *residual connection*
- Self-supervised disctiminator - регуляризация, которая заставляет дискриминатор смотреть на изображение в целом

Идея решения: Генератор

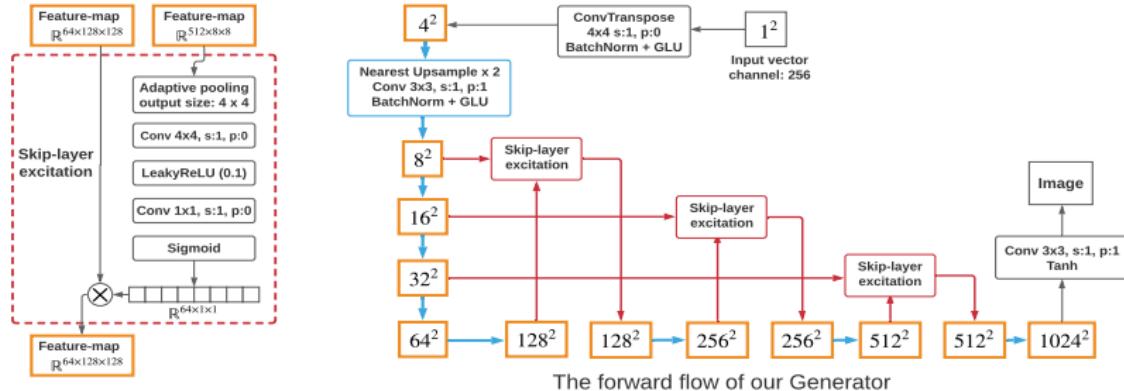


Рис.: Строение SLE блока слева и архитектура генератора справа. Здесь желтые блоки означают карту признаков с пропущенной глубиной. Голубые блоки и стрелки обозначают одинаковый процесс повышения размерности, который можно увидеть слева сверху.

Идея решения: SLE

Итого:

- Вносим правки в большие изображения на основе маленьких
- Похоже на внимание(attention)
- Дает более устойчивое течение градиента, как и ResBlock
- Приводит к выделению стиля

Отличия от ResBlock:

- Поканальное умножение вместо полного сложения
- ResBlock между одинаковыми разрешениями, а здесь большие с маленькими
- Возможность использования при разных разрешениях приводит к меньшему потреблению ресурсов

Идея решения: Self-supervised discriminator

Идея очень простая:

- ① Рассмотрим дискриминатор как энкодер
- ② Добавим декодер
- ③ Допишем в функционал ошибку восстановления

Идея решения: Self-supervised discriminator

Если быть точнее:

- ① Рассмотрим дискриминатор как энкодер
- ② Добавим несколько декодеров: на 2 разных разрешения и на часть изображения
- ③ Допишем в функционал ошибку восстановления(в авторской реализации LPIPS)

Идея решения: Self-supervised discriminator. Формулы

$$\mathcal{L}_{recons} = \mathbb{E}_{\mathbf{f} \sim D_{encode}(x), x \sim I_{real}} [||\mathcal{G}(\mathbf{f}) - \mathcal{T}(x)||], \quad (1)$$

\mathbf{f} - значения с промежуточных слоев дискриминатора,

\mathcal{G} - препроцессинг(вырезание, масштабирование) + декодер,

\mathcal{T} - просто препроцессинг

Идея решения: Self-supervised discriminator. Формулы

$$\begin{aligned}\mathcal{L}_D = & -\mathbb{E}_{x \sim I_{real}} [\min(0, -1 + D(x))] \\ & -\mathbb{E}_{\hat{x} \sim G(z)} [\min(0, -1 - D(\hat{x}))] + \mathcal{L}_{recons}\end{aligned}\tag{2}$$

$$\mathcal{L}_G = -\mathbb{E}_{z \sim \mathcal{N}} [D(G(z))]\tag{3}$$

Идея решения: Self-supervised discriminator. Архитектура

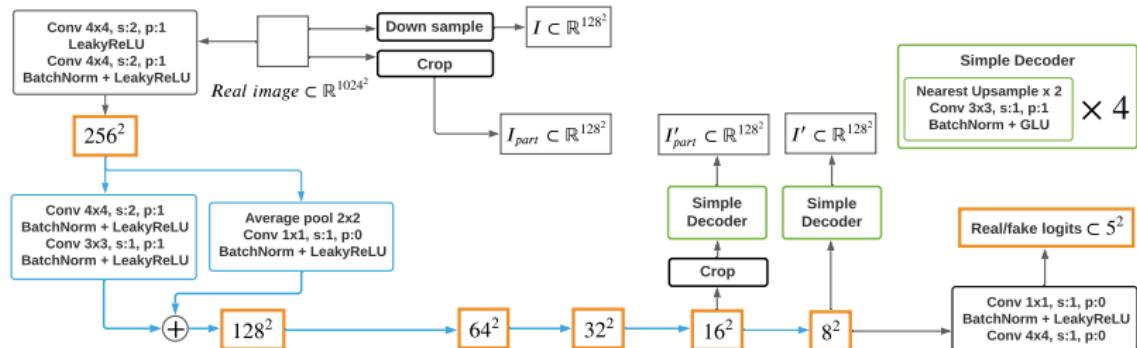


Рис.: Архитектура дискриминатора. Голубые блоки и стрелки означают один и тот же процесс понижения размерности, изображенный слева.

Идея решения: Self-supervised discriminator. Реализация

Про реализацию:

- Авторы используют два типа голубых блоков, чередуя такой же, как на изображении ранее и еще более простой
- Также они в дискриминаторе используют SLE блоки
- Используют ошибку декодера еще и на промежуточном разрешении 128^2

Результаты. Fréchet Inception Distance(FID)

Fréchet Distance

Расстояние Фреше между двумя распределениями вычисляется как

$$FD = \|\mu_x - \mu_g\|_2^2 + \text{Tr}(\Sigma_x + \Sigma_g - 2\sqrt{\Sigma_x \cdot \Sigma_g}) \quad (4)$$

Для того, чтобы получить *Fréchet Inception Distance(FID)* нужно прогнать настоящие и поддельные изображения через предобученный InceptionV3, построить распределение на внутренних представлениях(вычислить среднее, ковариации) и посчитать расстояние Фреше по формуле выше.

Результаты. Сравниваемые модели

① StyleGAN2

- Уменьшают число параметров до $\frac{1}{4}$ (как?)
- Для разрешения 1024^2 используют версию с $\frac{1}{2}$ параметрами

② Baseline - комбинация лучших техник для данной задачи

- Спектральная нормализация
- Экспоненциальное скользящее среднее на G
- Дифференцируемые аугментации
- *Gated linear unit(GLU)* вместо ReLU в G

Результаты. Сравниваемые модели

① StyleGAN2

- Уменьшают число параметров до $\frac{1}{4}$ (как?)
- Для разрешения 1024^2 используют версию с $\frac{1}{2}$ параметрами

② Baseline - комбинация лучших техник для данной задачи

- На самом деле проще понимать baseline как авторскую модель без модулей SLE и регуляризации в виде self-supervision на дискриминатор.

Результаты. Few-shot генерация

Таблица: Сравнение FID на изображениях 256^2 на небольших наборах данных.

		Animal Face - Dog		Animal Face - Cat		Obama	Panda	Grumpy-cat
Image number		389	160	100	100	100	100	100
Training time on one RTX 2080-Ti	20 hour	StyleGAN2	58.85	42.44	46.87	12.06	27.08	
		StyleGAN2 finetune	61.03	46.07	35.75	14.5	29.34	
	5 hour	Baseline	108.19	150.3	62.74	15.4	42.13	
		Baseline+Skip	94.21	72.97	52.50	14.39	38.17	
		Baseline+decode	56.25	36.74	44.34	10.12	29.38	
		Ours (B+Skip+decode)	50.66	35.11	41.05	10.03	26.65	

Результаты. Few-shot генерация

Таблица: Сравнение FID на изображениях 1024^2 на небольших наборах данных.

		Art Paintings	FFHQ	Flower	Pokemon	Anime Face	Skull	Shell
Image number		1000	1000	1000	800	120	100	60
Training time on one RTX TITAN	24 hour	StyleGAN2	74.56	25.66	45.23	190.23	152.73	127.98
		StyleGAN2 finetune	N/A	N/A	36.72	60.12	61.23	107.68
	8 hour	Baseline	62.27	38.35	42.25	67.86	101.23	186.45
		Ours	45.08	24.45	25.66	57.19	59.38	130.05

Результаты. Большие датасеты

Таблица: Сравнение FID на изображениях 1024^2 на больших наборах данных.

Model	Dataset	Art Paintings			FFHQ			Nature Photograph				
		Image number	2k	5k	10k	2k	5k	10k	70k	2k	5k	10k
StyleGAN2			70.02	48.36	41.23	18.38	10.45	7.86	4.4	67.12	41.47	39.05
Baseline			60.02	51.23	49.38	36.45	27.86	25.12	17.62	71.47	66.05	62.28
Ours			44.57	43.27	42.53	19.01	17.93	16.45	12.38	52.47	45.07	43.65

Результаты. Большие датасеты

Таблица: FID для разных версий self-supervision на D

	Art paintings	Nature photos
a. contrastive loss	47.14	57.04
b. predict aspect ratio	49.21	59.22
c. auto-encoding	42.53	43.65
d. a+b	46.02	54.23
e. a+b+c	44.21	47.65

Результаты. Визуальное сравнение

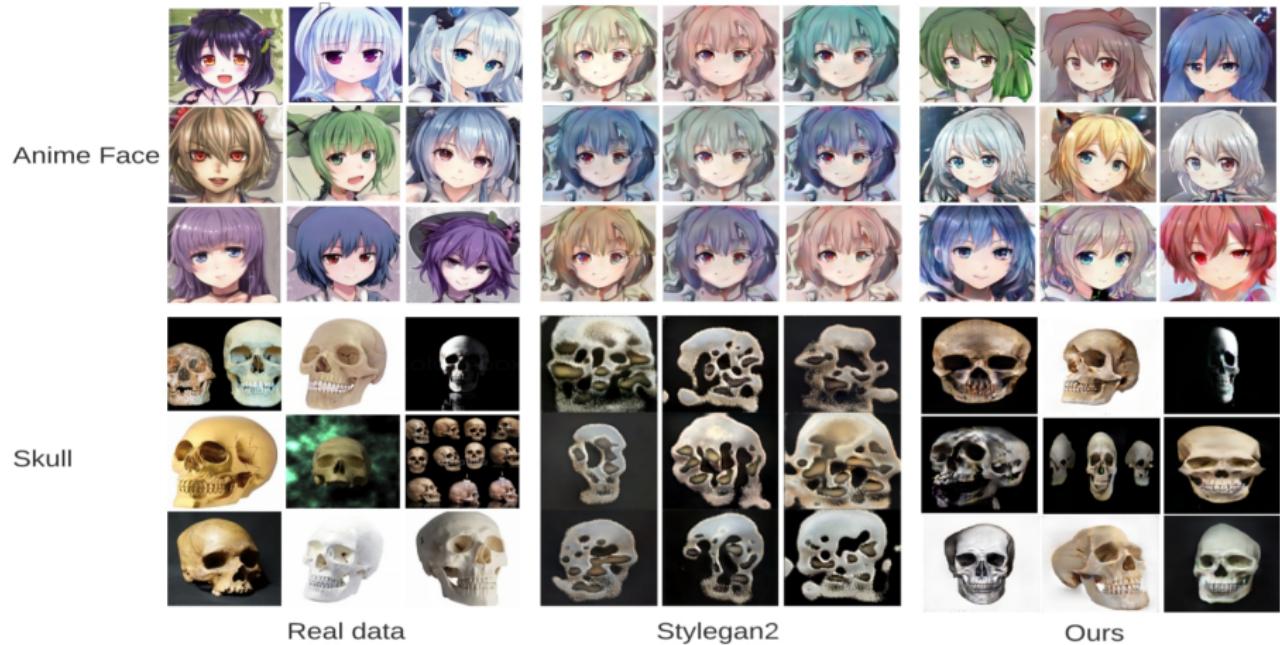


Рис.: Сравнение StyleGAN2 и авторской модели, обученных в течение 10 часов с размером батча равным 8, на изображениях 1024^2 . Примеры получены с помощью checkpoint'a с минимальным FID.

Результаты. Визуальное сравнение

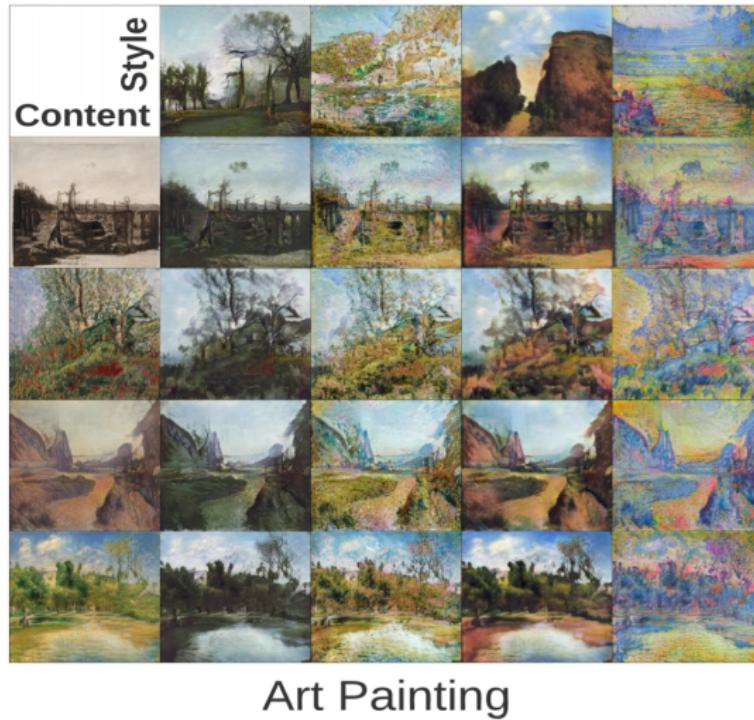


Рис.: Изменение стиля с помощью модели обученной за 5 часов.

Критика

- Недостаточно сравниваемых моделей, архитектур. Особенно с похожими методами
- Используется только одна метрика - FID, которая слабочувствительна к переобучению и плохо работает на небольших наборах данных
- Мало изучено переобучение
- Нет сравнения между разными масштабами данной архитектуры
- Почему бы не использовать SLE в дискриминаторе(уже есть в реализации)

Заключение

Кратко о том, что было

- Новый вид модуля *Skip-Layer channel wise Excitation*
- Регуляризация дискриминатора *self-supervised* стратегией
- Впечатляющие, но не полные результаты

References

Оригинальная статья: <https://arxiv.org/abs/2101.04775>
(оттуда же все изображения и таблицы)

Авторская реализация:

<https://github.com/odegeasslbc/FastGAN-pytorch>

Ревью на сайте *openreview.net*:

<https://openreview.net/forum?id=1Fqg133qRaI>

Ликбез по FID: <https://jonathan-hui.medium.com/gan-how-to-measure-gan-performance-64b988c47732>