

StyleGAN2

这篇论文也是英伟达的，解决了stylegan的特征滞留和水滴状伪影的问题，另外提出了PPL损失（感知路径长度损失），为了解决计算时间复杂度，提出了lazy regularization。这篇论文是StyleCLIP的基础（想从StyleCLIP里面找找感觉，看有没有办法通过文本控制对抗样本的生成）

StyleGAN2

2022年5月18日 9:44

CVPR 2022, NVIDIA

主要是为了解决 styleGAN 存在的 2 个问题

① 水滴状伪影（在 64×64 以上分辨率的图及特征图上都有水滴状伪影，并且越来越明显）

② 特征滞留（部分特征如人脸牙齿不会随着人脸的转动而转动）

作者发现：FID 和 P&R 指标关注的是图像的纹理而不是图像的质量，PPL（感知路径长度）与图像质量有关

一、水滴状伪影

实验：styleGAN 生成的图像都有水滴状伪影（始于 64×64 分辨率图像上，不仅图像上有，特征图上也有，并且会增强）

↓ 实验发现

原因出现在 AdaIN 上，该操作去除了特征图上特定信息，生成器通过这个伪影来补偿一个特定信息（?）



⇒ 水滴状伪影

↓ 优化方法

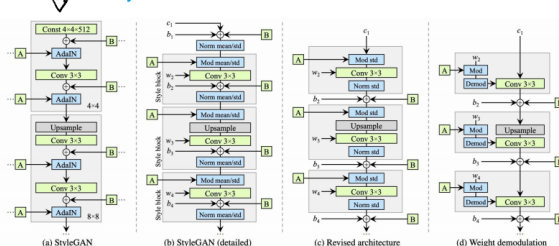


Figure 2. We redesign the architecture of the StyleGAN synthesis network. (a) The original StyleGAN, where \tilde{A} denotes a learned affine transform from \mathcal{V} that produces a style and \tilde{B} is a noise broadcast operation. (b) The same diagram with full detail. Here we have broken the AdaIN to explicit normalization followed by modulation, both operating on the mean and standard deviation per feature map. We have also annotated the learned weights (w), biases (b), and constant input (c), and redrawn the gray boxes so that one style is active per box. The activation function (leaky ReLU) is always applied right after adding the bias. (c) We make several changes to the original architecture that are justified in the main text. We remove some redundant operations at the beginning, move the addition of b and \tilde{B} to be outside active area of a style, and adjust only the standard deviation per feature map. (d) The revised architecture employs a “demodulation” operation, which we apply to the weights associated with each convolution layer.

(d)：逐步优化

(思想：对特征进行缩放可以近似通过对权重和偏置进行缩放实现)

$$\begin{cases} W'_{ijk} = S_r \cdot W_{ijk} & (W \text{ 是原始权重, } W' \text{ 是调制权重, } S_r \text{ 是对特征图的缩放}) \\ W''_{ijk} = \frac{W'_{ijk}}{\sqrt{\sum_{ik} W'_{ijk} + \epsilon}} & (\text{目的是将输出恢复为标准差}) \end{cases}$$

二、特征滞留（对渐进式增长方法的重思考）

实验：渐进式增长方法虽然可以生成高分辨率图像，但会造成特定的缺点（特征滞留）



⇒ 生成器对一些细节有一些固定的位置偏好。

如图，牙齿对应的中心（蓝线）一直没有随着人脸的转动而改变。

Figure 6. Progressive growing leads to “phase” artifacts. In this example the teeth do not follow the pose but stay aligned to the



Figure 6. Progressive growing leads to "phase" artifacts. In this example the teeth do not follow the pose but stay aligned to the camera, as indicated by the blue line.

→ 生成器对一些细节有一些固定的位置偏好。
如图，牙齿对应的中心（蓝线）一直没有随着人脸的转动而改变。

↓ 原因

在渐进式增长方法中，每个分辨率都可能作为最终分辨率，迫使它生成最大的频率细节，导致训练的网总在中层有过高的频率，损害了鲁棒性

↓ 改进法：跳跃连接，残差网络，分层网络

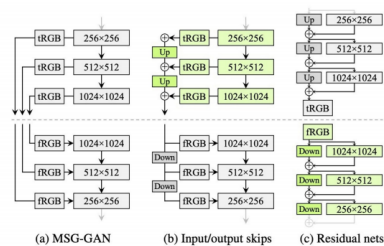


Figure 7. Three generator (above the dashed line) and discriminator architectures. [Up] and [Down] denote bilinear up and down-sampling, respectively. In residual networks these also include 1×1 convolutions to adjust the number of feature maps. [tRGB] and [rRGB] convert between RGB and high-dimensional per-pixel data. Architectures used in configs E and F are shown in green.

↓ 思想

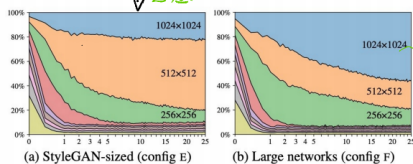


Figure 8. Contribution of each resolution to the output of the generator as a function of training time. The vertical axis shows a breakdown of the relative standard deviations of different resolutions, and the horizontal axis corresponds to training progress, measured in millions of training images shown to the discriminator. We can see that in the beginning the network focuses on low-resolution images and progressively shifts its focus on larger resolutions as training progresses. In (a) the generator basically outputs a 512^2 image with some minor sharpening for 1024^2 , while in (b) the larger network focuses more on the high-resolution details.

(a): 跳跃连接匹配分辨率，思想来自 MSG-GAN

(b): 对输出的图像进行上采样并相加，在判别器中进行下采样（采样方法用的都是双线性滤波）

(c): 使用残差结构

↓ 实验：(3个生成器 × 3个判别器 = 9种方法)

结论：跳跃连接有利于生成器，残差结构有利于判别器（不再用洋片式方法）

渐进式增长：生成器首先关注低分辨率特征，再慢慢转移到细节。

↓
生成器的跳跃连接也可以做到这一点

↓
更进一步：它没有洋片式方法所强制执行的，减弱了上采样在细节的产生。

→ 没有饱和，可以生成更高分辨率图像

三. 对图片质量的重新思考.



Figure 14. Uncarved examples from two generative models trained on LSUN Cars without truncation. FID, precision, and recall are similar for models 1 and 2, even though the latter produces car-shaped objects more often. Perceptual path length (PPL) indicates a clear preference for model 2. Model 1 corresponds to configuration A in Table 3, and model 2 is an early training snapshot of (B) (iterations 1.2).

FID和P&R评价指标关注的是纹理信息，PPL(感知距离)
 → 与图像的质量相同。

如图: Model1和Model2有相同的FID, $PPL_1 > PPL_2$, 明显图1没有图2的图像质量好

↓ 更低的PPL对应于更高的图片质量

但如果只是简单的最小化PPL, 可能会导致零回报率 (P&R的评价指标计算方法需要看一下)

↓ 方法

Path Length regularization

↓ 目的

使得在 w 中固定大小的步长会导致图像中非零的固定的变化 (与 w 的梯度有关)

↓ 计算

$g(w)$: 生成器, $J_w = \frac{\partial g(w)}{\partial w}$

正则化: $E_{w,y \sim N(0,I)} (\|J_w y\|_2 - \alpha)^2$
 ↓ 与梯度调整

计算: $J_w y = \nabla_w (g(w) \cdot y) \rightarrow$ 可以直接通过反向传播计算

↓ 思想来源

无论朝哪个方向变化, 都要保证向量的期望长度

↓ 解决正则化计算代价太高的问题

Lazy regularization

↓ 具体

以前的方案都是同时优化损失函数和正则化项, 但是经发现, 每16个批次只进行一次正则化也可以达到不错的结果, 同时减少了计算代价。

① 逆: 如何通过图片生成 w ?

方法: 先随机生成一个 w , 然后计算生成图片与原始图片的高维空间距离, 再一遍遍反向传播。

<< Image2StyleGAN: How to Embed Images Into the styleGAN Latent >> ICCV 2019 ★

② 感知损失

LPIPS: 学习感知图像相似性 (比L1/L2损失更好地反映图像质量)