최근 Computer Vision 분야에서 Transformer가 각광받고있다.

그러면 어려운 task인 GAN에서도 될까?

그래서 완전히 convolution을 사용하지 않은 GAN을 만들어 봤다. (오직 순수한 Transformer만 사용)

TransGAN은 memory-friendly transformer를 기반으로한

생성자 이다. 임베딩 차원이 감소하는동안 feature의 resolution은 점진적으로 증가

패치 수준의 discriminator 도 transformer 기반.

Data Augmentation에서 기존의 GAN보다 좋은 이점을 보였다.

+생성자에는 multi-task co-training 전략

+ 이미지의 국부영역을 강조하는 self-attention

또, 고화질에 대해서도 좋을 것이다.

SoTA에 있는 convolution 기반의 GAN이랑 비교했을 때에도 꽤 좋은 성능을 보였다.

STL-10 dataset에서 SoTA 달성

Introduction

GAN은 생성, 이미지 번역도 생성, 이미지 edit도 생성

그러나 GAN은 훈련 불안정이 문제여서 정규화, 더좋은 loss function, 더 나은 훈련방법 연구 진행

그리고 다양한 GAN에대한 연구...

그러다가 NAS를 GAN에 도입하고 backbone 설계도 GAN을 especially 개선하는게 중요하다고함

하지만 이런 연구들을 다 convolution 기반의 backbone 을 사용해야 한다는 상식을 깨뜨림.

그럼 우린 컨볼루션 아예 안쓰고 강한 GAN을 만들수 있을까?

- Transformer의 강점

1. 인간 지식의 편향이 적고 표현력 풍부

2. Ad-hoc 넥링 부족 제거가능

- **Model Architecture:** We build the first GAN using purely transformers and no convolution. To avoid overwhelming memory overheads, we create a memory-friendly generator and a patch-level discriminator, both transformer-based without bells and whistles. TransGAN can be effectively scaled up to larger models.

- **Training Technique:** We study a number of techniques to train TransGAN better, ranging from data augmentation, multi-task co-training for generator with self-supervised auxiliary loss, and localized initialization for self-attention. Extensive ablation studies, discussions, and insights are presented. None of them requires any architecture change.

- **Performance:** TransGAN achieves highly competitive performance compared to current state-of-the-art CNN-based GANs. Specifically, it sets new state-of-the-art IS score of 10.10 and FID score of 25.32 on STL-10 and also reaches competitive 8.63 IS score and 11.89 FID score on CIFAR-10, and 12.23 FID score on CelebA 64×64, respectively. We also summarize the current limitations and future potential of TransGAN.

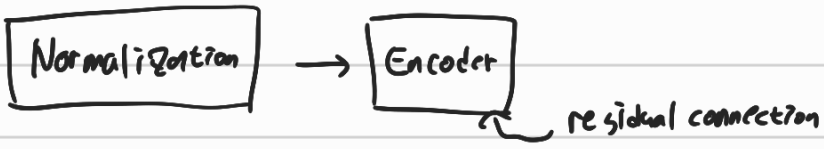
← Contribution

Method

- Basic Block

: Transformer encoder 를 basic block 으로 사용

encoder 구성 { multi-head self-attention module
MLP (feed forward) with GELU



- Memory friendly generator

점점 작은 Seq 늘리는 방법

Input : random noise

→ MLP 통과시킴 → H x W x C (vector 만큼 → feature map은 학습가능한 positional encoding과 결합

비슷하게 transformer encoder를 인풋으로 넣어주고 상단 관계 없이 따라 각각 계산

고해상도 이미지를 합성하기 위해 각 stage 뒤에 pixel shuffle module로 구성된 upsampling 모듈을 넣는다

pixel shuffle

$H \times W \times C \rightarrow 2H \times 2W \times C/4 \rightarrow 4H \times 4W \times C/16 \rightarrow 4H \times 4W \times 3$

↳ 동일 resolution