TransGAN:
Two Transformers Can Make One Strong GAN

School of Industrial and Management Engineering, Korea University

Sae Rin Lim

KOREA UNIVERSITY

DMQA hcai Human-Centered Artificial Intelligence Center

Contents

Background : GAN

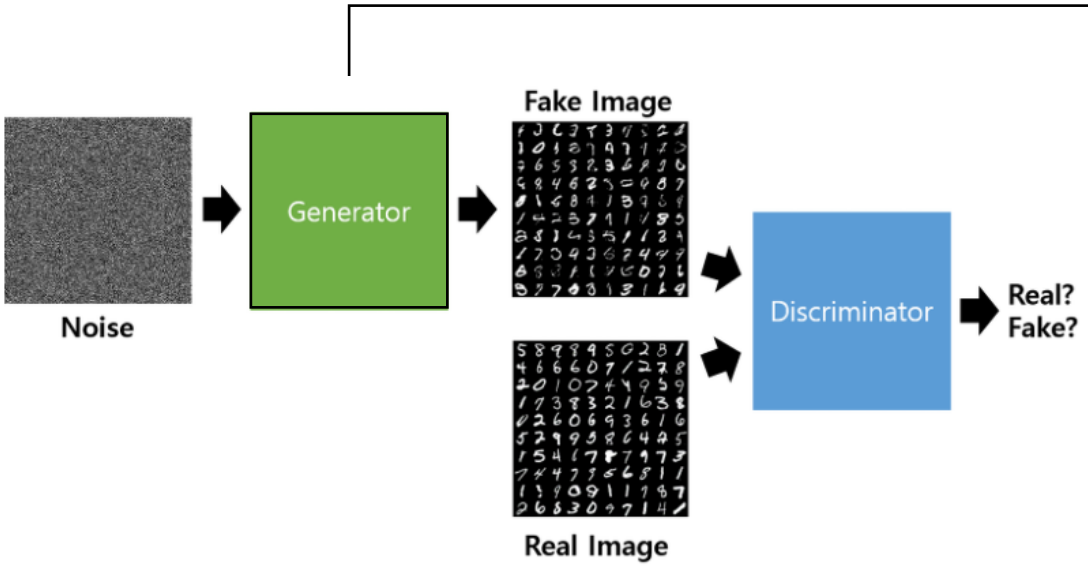
- **GAN Keywords**

❖ Generator : 정규분포를 따르는 random noise vector를 실제 데이터 분포로 mapping하여 **가짜 이미지를 생성**하는 모델

❖ Discriminator : 실제 이미지와 Generator가 생성한 이미지를 입력 받아 **진짜인지 가짜인지 판별**하는 모델

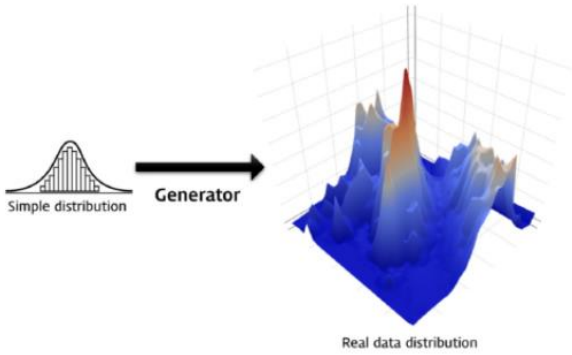
❖ Min-Max Game : Generator는Discriminator 를 속임으로써 **목적함수를 감소**시키고 Discriminator는 정확히 판별함으로써 **목적함수를 증가**시키는 Generator와 Discriminator의 Min-Max Game

❖ Generative Adversarial Nets : **Generator**와 **Discriminator**가 하나의 **목적함수(Adversarial Loss)**를 통해 적대적으로 학습하는 모델



GAN의 모델 구성도 및 학습 과정

[출처 : https://hyeongminlee.github.io/post/gan001_gan/]



Generator를 통한 noise mapping

[출처 : https://wordbe.tistory.com/entry/GAN]

Overview of TransGAN

- **TransGAN**

❖ Department of Electronic and Computer Engineering, University of Texas at Austin [1]

❖ USA MIT−IBM Watson AI Lab[2]

❖ 2021년 2월 14일 arXiv Computer Vision and Pattern Recognition

❖ 2021년 6월 28일 기준 25회 인용

❖ **'Transformer가 Computer Vision의 다른 Task인 GAN을 해낼 수 있을까?'**에 대한 최초의 pilot study

---

TransGAN: Two Transformers Can Make One Strong GAN

Yifan Jiang [1] **Shiyu Chang** [2] **Zhangyang Wang** [1]

Contributions of TransGAN

- **Issues and Contributions**

❖ 기존 이미지를 출력하는 Transformer 모델들은 CNN-based encoder 또는 feature extractor를 활용

❖ 잘 구성된 CNN-based GAN도 학습이 불안정하고 모델 collapse의 위험이 있음

❖ Vision Transformer 또한 inductive bias를 학습하기 위해 많은 데이터가 필요한 data-hungry 모델임

위 이슈들을 해결하는 방향으로 Contribution을 정의

1. **Model Architecture**
   - Convolution없이 순순하게 transformer만을 이용한 GAN model
   - Memory overheads를 피하기 위한 memory-friendly generator and discriminator 구성
2. **Training Technique**
   - 기존 CNN-based GAN과 비교할 때 Data Augmentation의 효과 증명
   - 불안정한 GAN의 학습을 안정화 시키기 위한 co-training with self-supervised auxiliary loss
   - CNN처럼 지역적인 특성을 잘 학습하기 위한 locally initialized self-attention
3. **Performance**
   - CNN-based SOTA Model과 비슷하거나 더 좋은 성능 증명

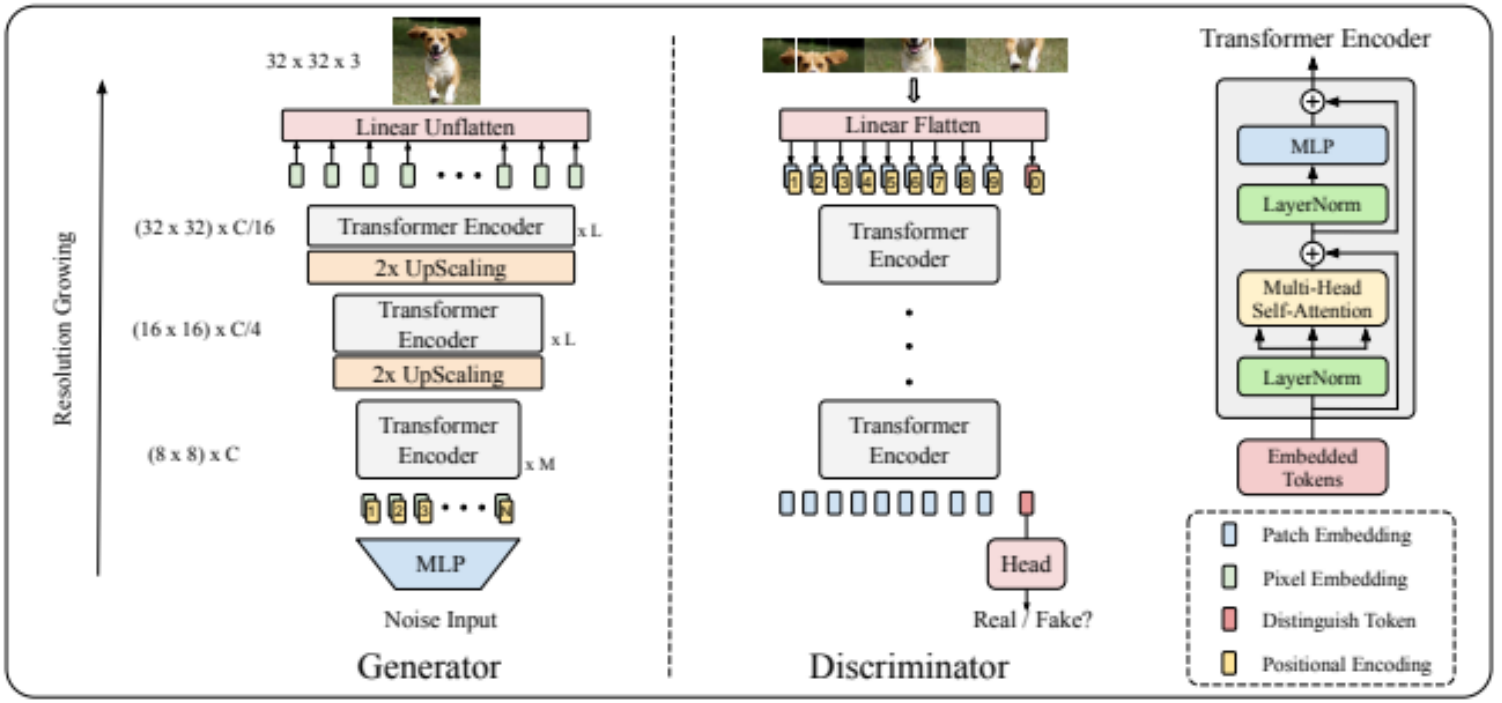
Contributions of TransGAN

- **Model Architecture**

❖ Transformer의 Encoder를 활용하여 Generator와 Discriminator 모델 구축

❖ **Pixel 단위**로 이미지를 생성하는 Generator

❖ **Patch 단위**로 이미지를 분류하는 Discriminator



TransGAN 모델 구조

Contributions of TransGAN

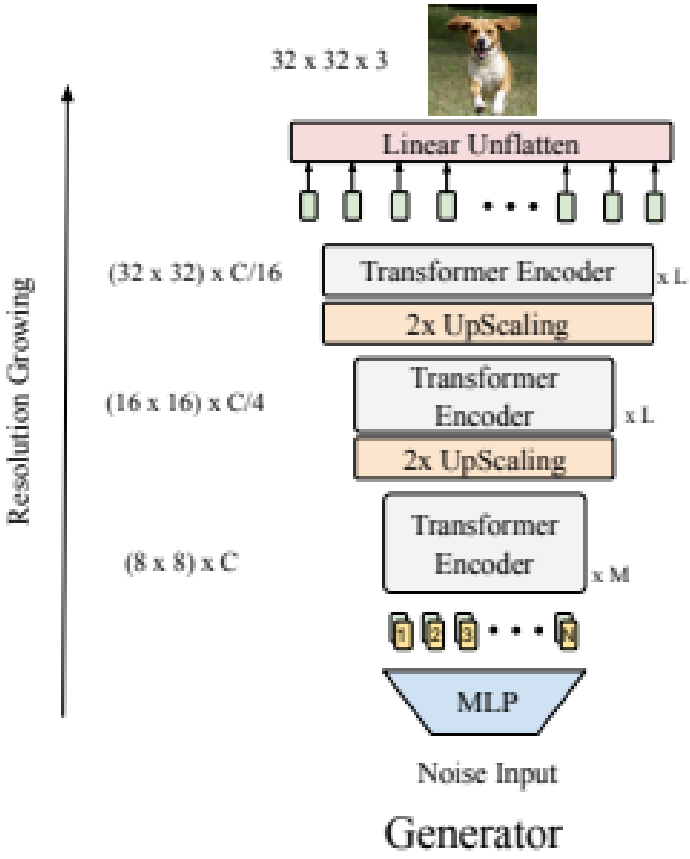
- **Model Architecture : Memory-friendly Generator**

❖ Generator에서는 **이미지를 pixel 단위로 생성**해야 하기 때문에 낮은 해상도의 이미지라도 입력 시퀀스 크기가 커짐 (ex. 32 by 32 image의 pixel 수 = generator의 입력 = 1024 pixel)

❖ 입력 시퀀스의 크기에 따라 self-attention **계산 비용 역시 quadratic하게 증가**하기 때문에 이를 해결하기 위한 **Memory-friendly generator** 구축

❖ 입력 시퀀스를 늘리는 동시에(image resolution) embedding dimension을 줄이는 방법을 stage별로 target image size가 될 때까지 반복하여 이미지 생성함으로써 계산 비용을 줄임

❖ 이러한 특성을 이용하여 큰 이미지에 대해 효율적으로 모델을 확장할 수 있음

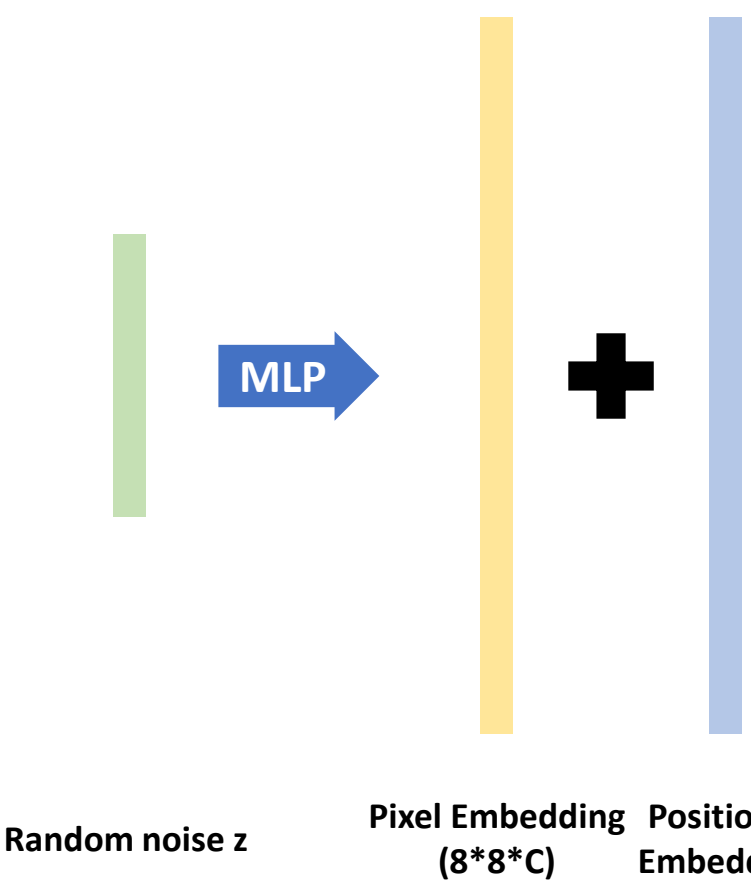


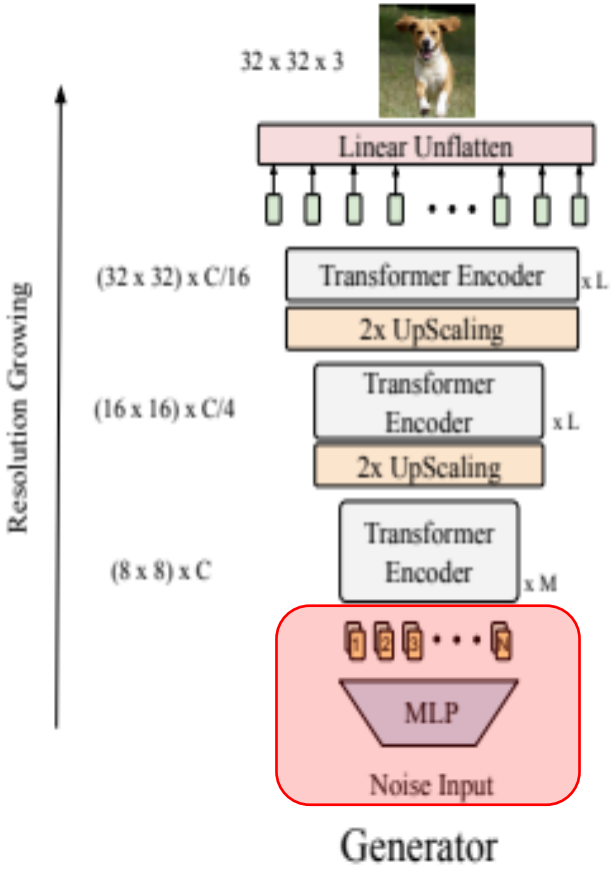
Generator

Contributions of TransGAN

- **Example of stage flow : 8 by 8 to 16 by 16**

1. Pixel Embedding과 Positional Embedding vector를 더해 Transformer encoder의 입력 벡터 생성

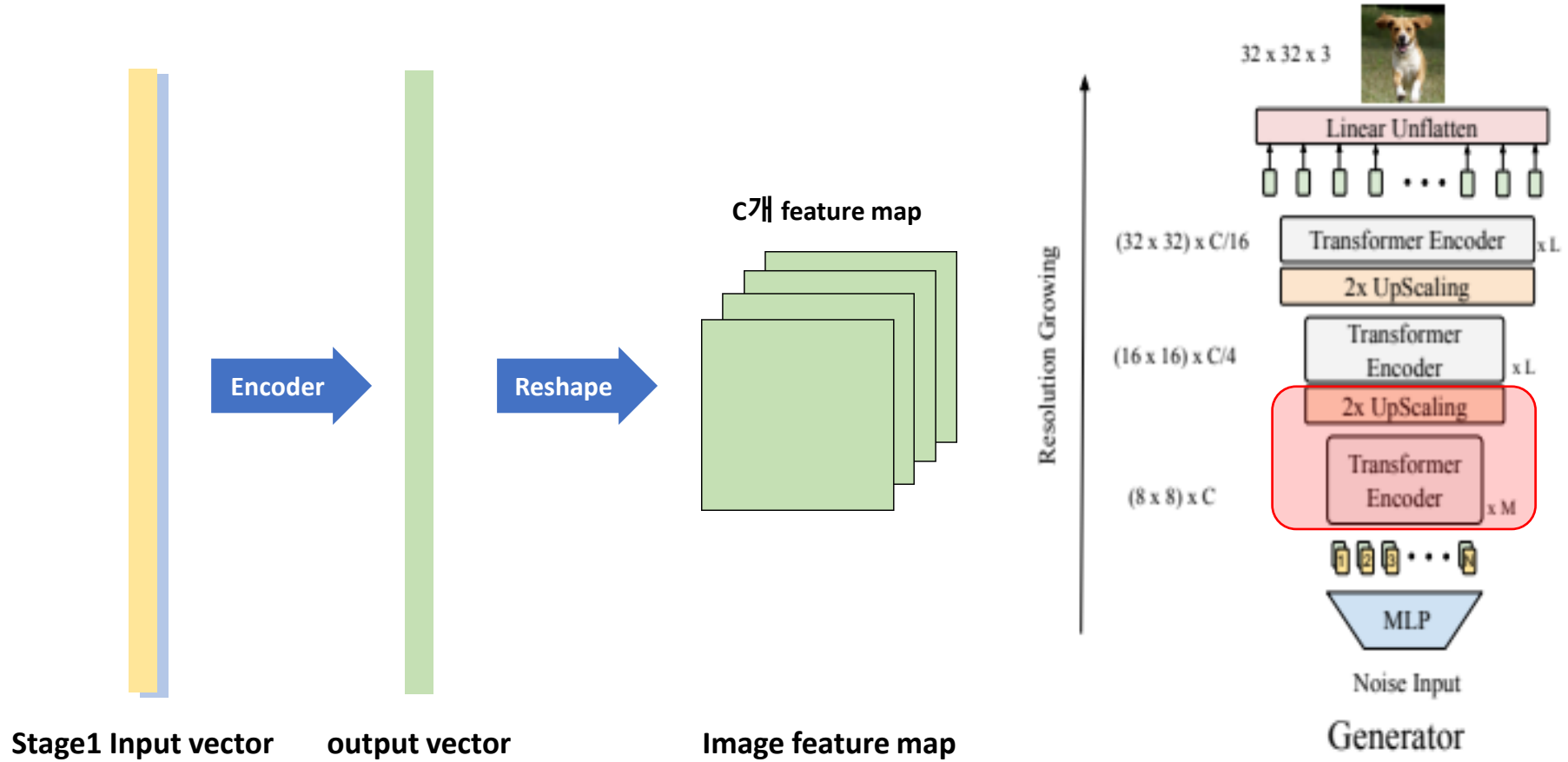




Contributions of TransGAN

- **Example of stage : 8 by 8 to 16 by 16**

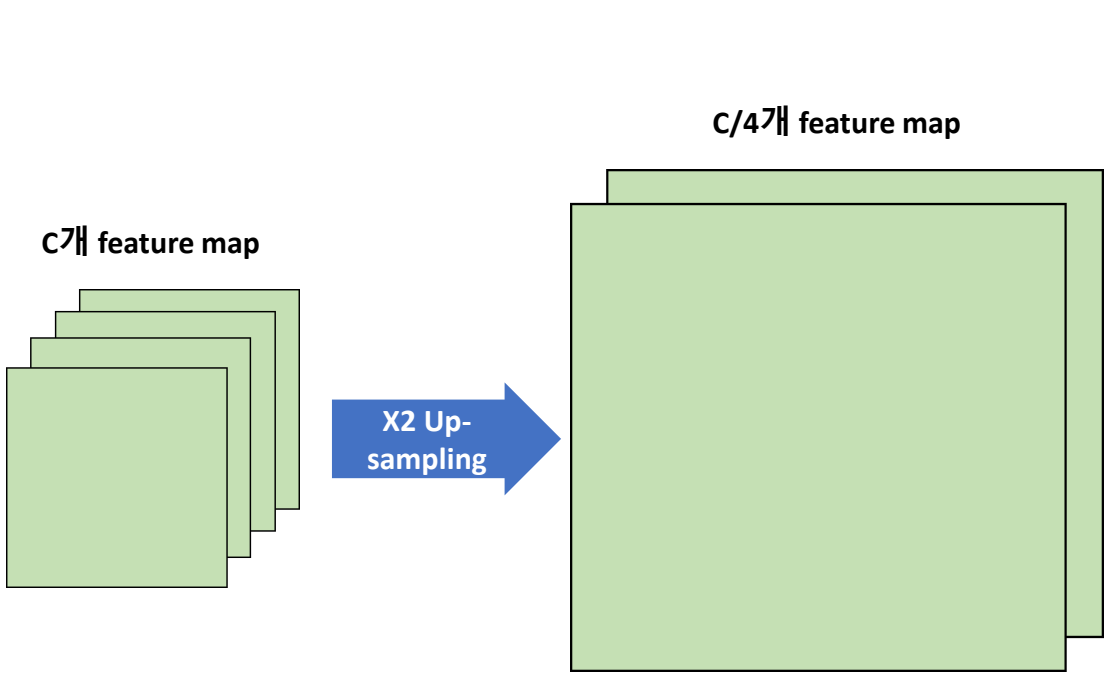
2. 입력 벡터를 transformer encoder에 태워 출력 벡터를 얻고 이를 reshape하여 image feature map 생성

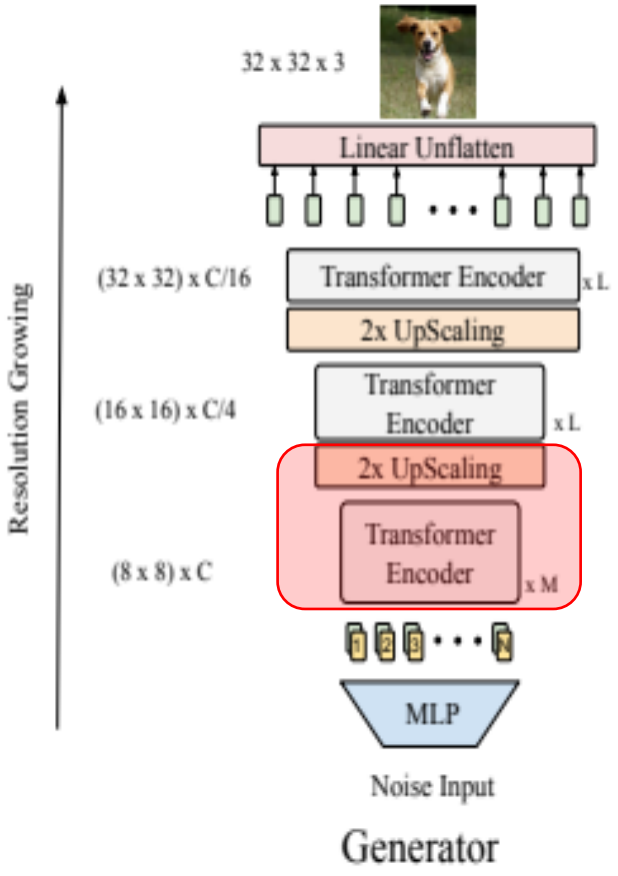


Contributions of TransGAN

- **Example of stage : 8 by 8 to 16 by 16**

3. Feature map을 up-sampling하여 크기를 2배로 늘리면서 pixel embedding size(=feature map 개수)를 ¼로 줄임





Contributions of TransGAN

- **Example of stage : 8 by 8 to 16 by 16**

  4. Up-sampling한 feature map을 flatten하여 다시 1-D vector로 변환하여 다음 stage의 encoder 입력 벡터 생성

  5. 2~4 를 하나의 stage로 반복하여 원하는 target size가 될 때까지 진행

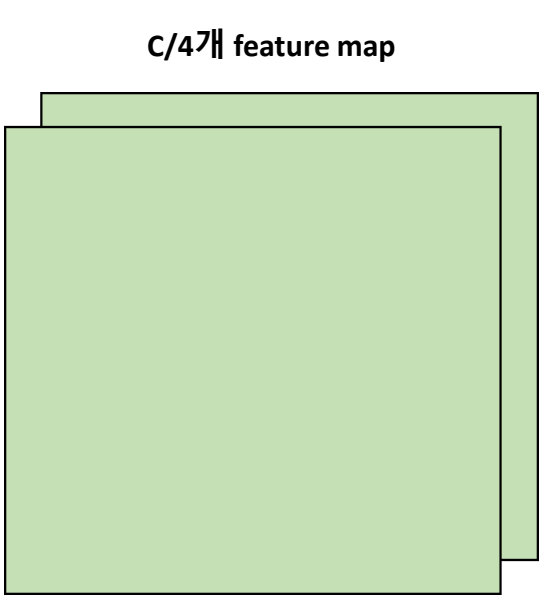
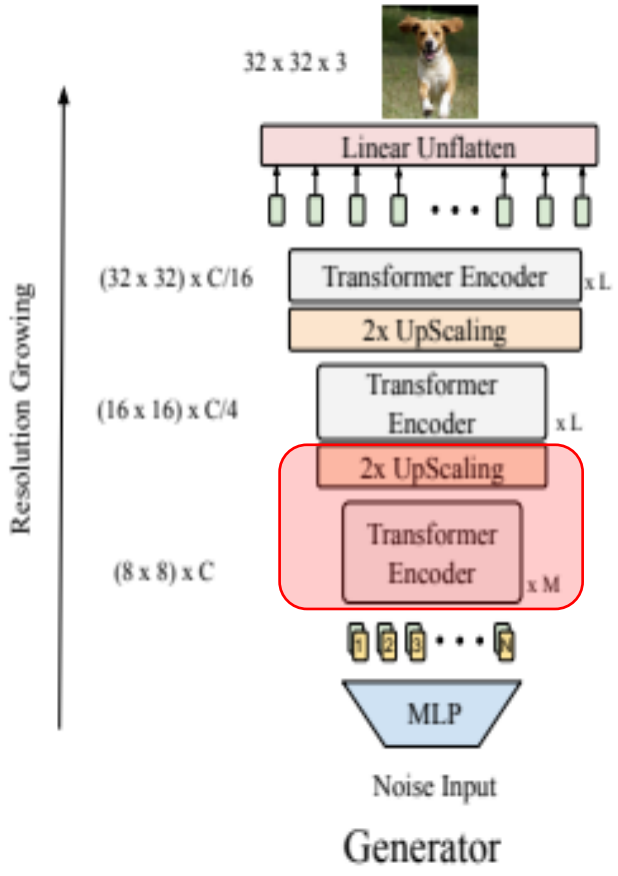


Image feature map
(16,16,C/4)

Flatten

Stage2 input vector
(16*16*C/4) = (8*8*C)



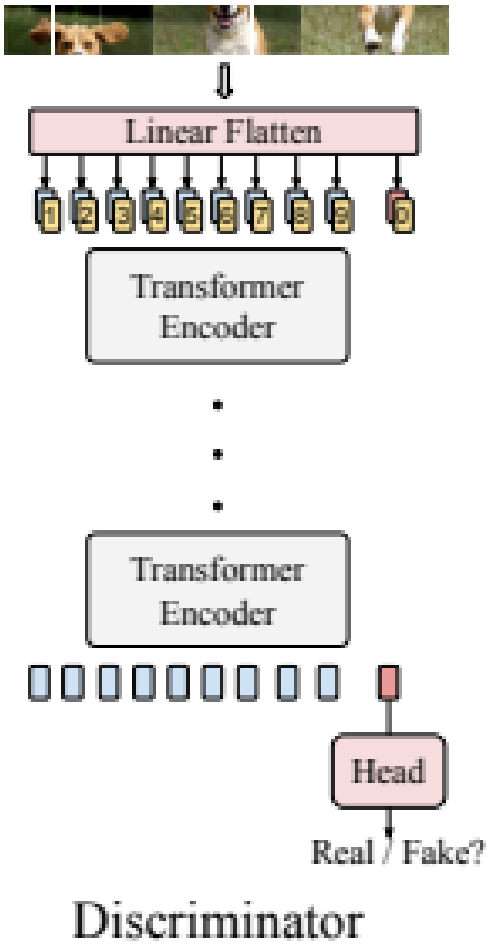
Generator

Contributions of TransGAN

- **Model Architecture : ViT-based Discriminator**

❖ Discriminator에서는 ViT와 마찬가지로 이미지를 patch 단위로 잘라서 한 patch를 단어처럼 Transformer Encoder에 입력 변수로 사용

❖ 여러 Encoder 층을 통과한 최종 출력 변수에서 CLS Token에 해당하는 Token을 통해 진짜 이미지인지 가짜 이미지인지를 판단



Contributions of TransGAN

- **Training Technique : Data Augmentation**
  - ❖ CNN-based SOTA모델인 AUTOGAN의 generator와 discriminator를 TransGAN으로 바꿔가며 실험 진행
  - ❖ TransGAN Generator는 좋은 성능을 보였지만 Discriminator는 성능이 좋지 않음을 발견
  - ❖ 저자들은 ViT에서 transformer-based classifier가 inductive bias를 학습하기 어려워 많은 데이터를 필요로 하는 data-hungry 한 모델이기 때문에 TransGAN Discriminator도 같을 것이라 생각하고 **Data Augmentation(DA)기법**을 도입
  - ❖ CNN-based SOTA모델 3개와 TransGAN 모델을 가지고 DA의 효과 비교
  - ❖ CNN-based 모델에서는 큰 성능 향상이 없었지만 TransGAN은 큰 성능 향상을 보임으로써 DA의 중요성 확인

GENERATOR	DISCRIMINATOR	IS↑	FID↓
AUTOGAN	AUTOGAN	8.55± 0.12	**12.42**
TRANSFORMER	AUTOGAN	**8.59± 0.10**	13.23
AUTOGAN	TRANSFORMER	6.17± 0.12	49.83
TRANSFORMER	TRANSFORMER	6.95 ± 0.13	41.41

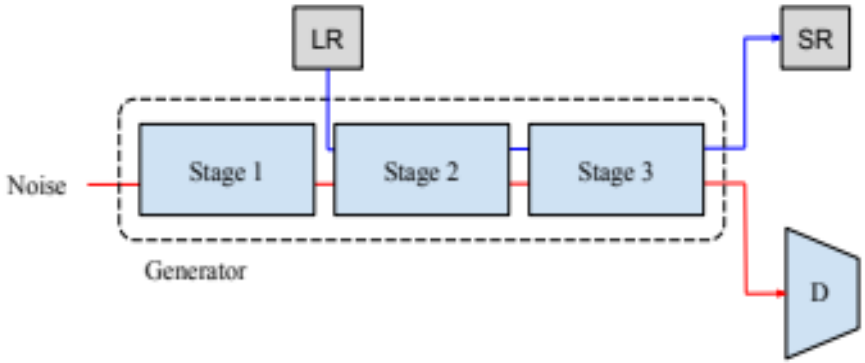
CNN-based와 transformer-based의 비교실험

METHODS	DA	IS ↑	FID ↓
WGAN-GP (GULRAJANI ET AL., 2017)	×	**6.49** ± 0.09	39.68
	√	6.29 ± 0.10	**37.14**
AUTOGAN (GONG ET AL., 2019)	×	8.55 ± 0.12	**12.42**
	√	**8.60** ± **0.10**	12.72
STYLEGAN V2 (ZHAO ET AL., 2020B)	×	9.18	11.07
	√	**9.40**	**9.89**
TRANSGAN	×	6.95 ± 0.13	41.41
	√	**8.15** ± **0.14**	**19.85**

Data Augmentation 비교실험

Contributions of TransGAN

- **Training Technique : Co-Training with Self-Supervised Auxiliary Task**
  - ❖ NLP 분야의 BERT와 같은 모델에서 여러 Task를 동시에 학습하는 방법이 효과가 있음을 증명
  - ❖ 또한, GAN에서 self-supervised auxiliary task를 활용하는 것이 GAN 학습에 안정성을 더해준다는 연구가 있음
  - ❖ 위 두 연구를 기반으로 TransGAN에 **Co-Training with Self-Supervised Auxiliary Task**를 추가
  - ❖ 기존 이미지에 Down-Sampling한 것을 Low Resolution으로 사용하는 Resolution task도입하여 추가 데이터가 필요 없음
  - ❖ 목적함수에 Resolution에 대한 MSE loss term을 추가하고 가중치 $\lambda = 50$ 으로 설정
  - ❖ 실험결과 조금의 성능 향상이 있음을 확인



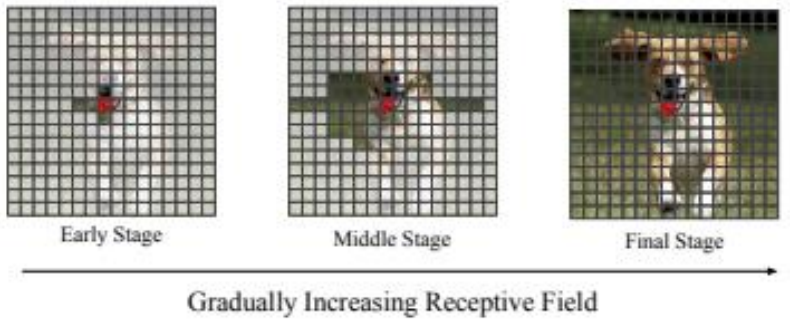
Resolution Task를 추가한 TransGAN 학습 flow

MODEL	IS↑	FID↓
TRANSGAN + DA (*)	8.15± 0.14	19.85
(*) + MT-CT	8.20± 0.14	19.12
(*) + MT-CT + LOCAL INIT.	**8.22± 0.12**	**18.58**

Multi-Task Co-Training 효과 검증 실험

Contributions of TransGAN

- **Training Technique : Locality-Aware Initialization for Self-Attention**
  - ❖ CNN의 convolution 연산은 receptive field가 가중치를 공유하기 때문에 이동에 동변하고 이미지의 지역적 특성을 잘 학습할 수 있음
  - ❖ 하지만 receptive field가 특정 크기만을 보기 때문에 global한 특징을 학습하려면 layer를 깊게 쌓아야 하며 이는 디테일한 특징을 잘 잡지 못하며 그 특징을 손실할 수도 있고 수렴의 어려움을 야기함
  - ❖ 반면에 Transformer는 self-attention을 통해 global한 특징을 잘 잡을 수 있는 CNN보다 더 genera한 모델이지만 local한 특성을 학습하기 어려움
  - ❖ 이 이슈를 해결하기 위해 self-attention을 **현재 pixel의 이웃만 계산**하도록 Mask를 씌운 locality-aware initialization 도입하여 일종의 규제 효과를 줌
  - ❖ 학습 초반에 적용해 self-attention이 지역적 특성을 학습하도록 규제함으로써 학습 초반 변동성을 줄이고 점점 Mask 크기를 줄여 최종적으로 self-attention을 통해 global한 특성을 학습
  - ❖ 실험결과 조금의 성능 향상이 있음을 확인



Model	IS↑	FID↓
TransGAN + DA (*)	8.15± 0.14	19.85
(*) + MT-CT	8.20± 0.14	19.12
(*) + MT-CT + Local init.	**8.22± 0.12**	**18.58**

Locality-Aware Initialization for Self-Attention 효과 검증 실험

Contributions of TransGAN

- **Performance : Model Size**
  - ❖ 논문에서 Pixel embedding dimension(= feature map 수)와 transformer의 encoder block을 늘려가며 ablation study 진행
  - ❖ Model Size가 증가할 수록 성능이 좋아지는 것을 확인

✓ DEPTH : 각 stage 별 transformer encoder block 개수

✓ DIM : Pixel embedding dimension

MODEL	DEPTH	DIM	IS ↑	FID ↓
TRANSGAN-S	{5,2,2}	384	8.22 ± 0.14	18.58
TRANSGAN-M	{5,2,2}	512	8.36 ± 0.12	16.27
TRANSGAN-L	{5,2,2}	768	8.50 ± 0.14	14.46
TRANSGAN-XL	{5,4,2}	1024	$\mathbf{8.63 \pm 0.16}$	**11.89**

TransGAN 모델 크기 별 CIFAR-10 dataset에 대한 성능 비교

Contributions of TransGAN

- **Performance**
  - ❖ TransGAN-XL 모델을 다른 CNN-based 모델과 비교
  - ❖ 순수한 transformer 구조로는 처음으로 SOTA에 준하는 성능을 내거나 SOTA보다 뛰어난 성능을 달성
  - ❖ 저자들은 더 많은 training technique과 tuning을 통해 더 좋은 성능을 낼 수 있을 것이라고 주장

METHODS	IS	FID
WGAN-GP (GULRAJANI ET AL., 2017)	6.49 ± 0.09	39.68
LRGAN (YANG ET AL., 2017)	7.17 ± 0.17	-
DFM (WARDE-FARLEY & BENGIO, 2016)	7.72 ± 0.13	-
SPLITTING GAN (GRINBLAT ET AL., 2017)	7.90 ± 0.09	-
IMPROVING MMD-GAN (WANG ET AL., 2018A)	8.29	16.21
MGAN (HOANG ET AL., 2018)	8.33 ± 0.10	26.7
SN-GAN (MIYATO ET AL., 2018)	8.22 ± 0.05	21.7
PROGRESSIVE-GAN (KARRAS ET AL., 2017)	8.80 ± 0.05	15.52
AUTOGAN (GONG ET AL., 2019)	8.55 ± 0.10	12.42
STYLEGAN V2 (ZHAO ET AL., 2020B)	**9.18**	**11.07**
TRANSGAN-XL	8.63 ± 0.16	11.89

CIFAR-10 dataset

METHODS	IS ↑	FID ↓
DFM (WARDE-FARLEY & BENGIO, 2016)	8.51 ± 0.13	-
D2GAN (NGUYEN ET AL., 2017)	7.98	-
PROBGAN (HE ET AL., 2019)	8.87 ± 0.09	47.74
DIST-GAN (TRAN ET AL., 2018)	-	36.19
SN-GAN (MIYATO ET AL., 2018)	9.16 ± 0.12	40.1
IMPROVING MMD-GAN (WANG ET AL., 2018A)	9.23 ± 0.08	37.64
AUTOGAN (GONG ET AL., 2019)	9.16 ± 0.12	31.01
ADVERSARIALNAS-GAN (GAO ET AL., 2020)	9.63 ± 0.19	26.98
TRANSGAN-XL	**10.10 ± 0.17**	**25.32**

STL-10 dataset

Limitations and future potential of TransGAN

- 이미지 생성이라는 어려운 task와 GAN의 학습 불안정성을 해결하고 최초로 pure transformer-based GAN 모델을 통해 CNN-based SOTA 모델과 비슷한 성능을 낸 것에 저자들은 큰 의의를 두고 있음
- 순수한 Transformer만을 이용함으로써 NLP 분야에서 Transformer를 기반으로 다양한 Task를 한 번에 해결할 수 있는 universal한 모델이 많이 등장하고 있는 것처럼 Vision 분야에서도 이를 시도하기 위한 기반을 닦는 연구라고 주장
- 반대로 단순히 CNN-based의 모델을 transformer-based 모델로 바꾸는 것을 목표로 한 연구이기 때문에 디테일한 tuning이나 다양한 학습기법에 대한 실험이 부족
- 저자들은 semantic grouping, pretext task, stronger attention form 등의 학습 기법을 적용하여 모델을 개선할 것이라 함

Reference

1. Jiang, Y., Chang, S., & Wang, Z. (2021). Transgan: Two transformers can make one strong gan. arXiv preprint arXiv:2102.07074.
2. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.


Thank You