

논문 정리

[CycleGAN] Unpaired Image-to-Image Translation using Cycle-

DI_Lee 2021. 4. 15. 14:58

이번에 정리할 논문은 바로 Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks라는 논문이다. 이 논문에서는 Cycle GAN으로 잘 알려진 모델을 소개하고 있다.

(Cycle GAN에 대해서 공부하기 전에 pix2pix 논문을 공부하면 많이 도움이 된다!)
그럼 이제부터 cycle GAN이 뭘하는 친구인지 살펴보도록 하자.

Cycle GAN 이란?



한 도메인의 이미지를 다른 도메인으로 해석하는 IMAGE TO IMAGE TRANSLATION 모델



일단 간단히 CycleGAN이 무엇을 하는 친구인지 예시들을 통해서 살펴보자.

지난 시간에 발표를 했던 GAN이라는 모델은 어떤 데이터 분포를 입력받아 실제에 가까운 데이터를 생성하는 모델이었다. 따라서 GAN은 input image와 output image간의 연관성에 초점을 맞췄다. 기 보다는 최대한 진짜 같은 이미지를 생성하는 것에 초점이 맞춰진 모델이었다. 하지만 cycleGAN의 경우는 다르다. 여기 이미지들을 보면 굉장히 흥미로운 결과들이 많이 보이는데, 이 것들의 공통점은 모두 **어떤 한 도메인의 이미지를 다른 도메인으로 해석한 결과**라는 것이다. 여기 보이는 예시는 클로드 모네의 그림을 이렇게 사진으로 변환한 결과이다. 또한 이렇게 한방향으로만 변환이 가능한 것이 아니라 CycleGAN을 이용하면 동일한 모델로 이와 같이 사진을 다시 클로드 모네의 화풍으로 변환하는 반대의 작업도 수행하는 것이 가능해진다. 이렇게 그림을 사진으로 바꾸는 것뿐만 아니라 이와 같이 여름의 풍경사진을 겨울로 바꾸거나, 여기에는 나와있지 않지만 뭐 얼룩말의 사진을 일반 말의 사진으로 변화하는 것도 CycleGAN을 이용하면 가능해진다.

이렇게 이미지 데이터셋을 사용하여 input image와 output 이미지를 mapping하는 것을 목표로 하는 생성모델의 한 분야를 **image to image translation**이라고 한다.

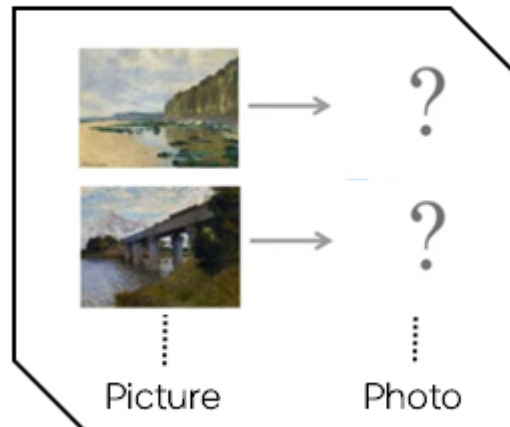
Pix2pix vs CycleGAN

그럼 지난시간에 정리했던 Pix2pix와 CycleGAN의 차이가 무엇일까??

Pix2pix



CycleGAN



먼저 앞에서 살펴보았던 PIX2PIX 모델의 데이터 특징부터 살펴보도록 하자.

앞서 말했듯이 pix2pix 모델에서 적용했던 데이터들은 모두 짝을 이루는 paired image들이었다. 하지만 모든 이미지가 다 짝으로 존재할까? 예를들어 그림을 사진으로 바꾸고 싶다고 했을 때, 모든 그림에 대해서 사진이 존재하는 것은 아닌 것처럼 모든 이미지들이 다 짝을 이루면서 존재하는 것은 아니다. Cycle GAN은 바로 이러한 PIX2PIX 모델의 한계를 극복한 모델이라고 할 수있다. 그럼 CycleGAN의 loss를 살펴보면서 cycleGAN이 어떤 방법으로 이런 한계를 극복할 수 있었을지 살펴보자.

Cycle GAN LOSS function

GAN LOSS

Cycle GAN

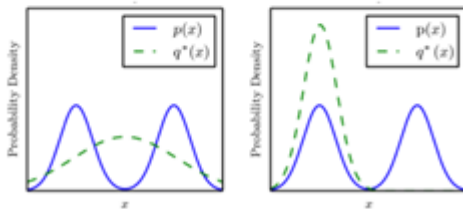
manypeople... 구독하기

Loss 1

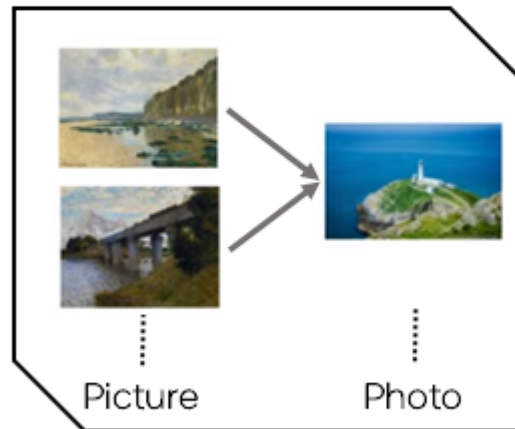
$$L_{GAN}(G(x), y)$$



Mode collapse 발생

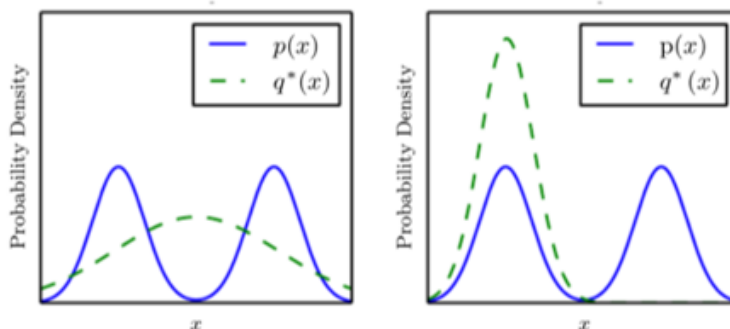


CycleGAN



먼저 cycle gan도 결국에는 input 이미지와 mapping되는 ‘진짜 같은’ 이미지를 생성을 하는 것이기 때문에 첫번째로 GAN loss를 이용하게 된다. 여기서는 conditional GAN이 아닌 그냥 GAN을 이용하게 되는데, 하지만 이때 LOSS를 GAN으로만 구성하게 되면 MODE COLLAPSE라는 문제가 발생하게 된다.

Mode collapse 발생

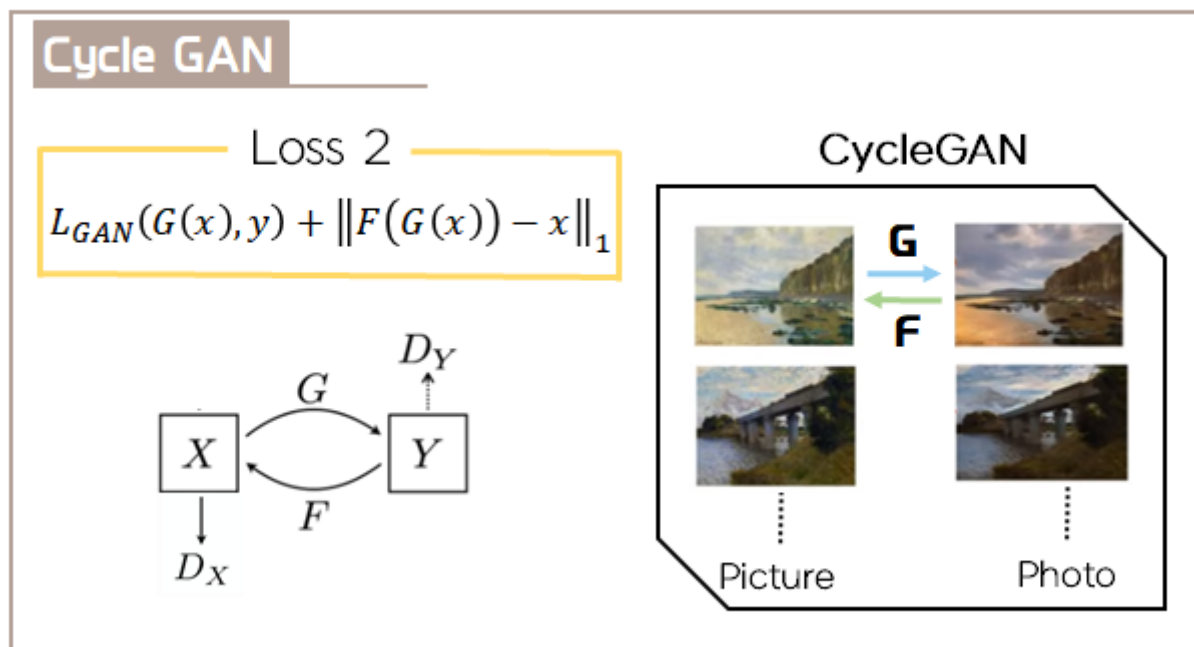


Mode collapse라는 문제는 이렇게 파란색의 실제 데이터의 분포가 주어졌을 때, 우리는 generator가 이 실제 데이터의 분포와 최대한 유사하게 학습하기를 바라게 된다. 그러나 단순히 loss만을 줄이기 위해서 학습을 하기 때문에 G가 이렇게 전체 데이터 분포를 찾지 못하고 오른쪽 그림과 같이 한번에 하나의 mode에만 강하게 몰리게 되는 경우가 발생을 하게 된다. 이렇게 되면 서로 다른 두 그림의 아웃풋이 이렇게 동일한 사진이 나오는 경우가 발생하게 된다.

Cycle LOSS

그래서 추가되는 Loss가 바로 pix2pix에서 L1 loss와 유사한 역할을 하는

manypeople... 구독하기

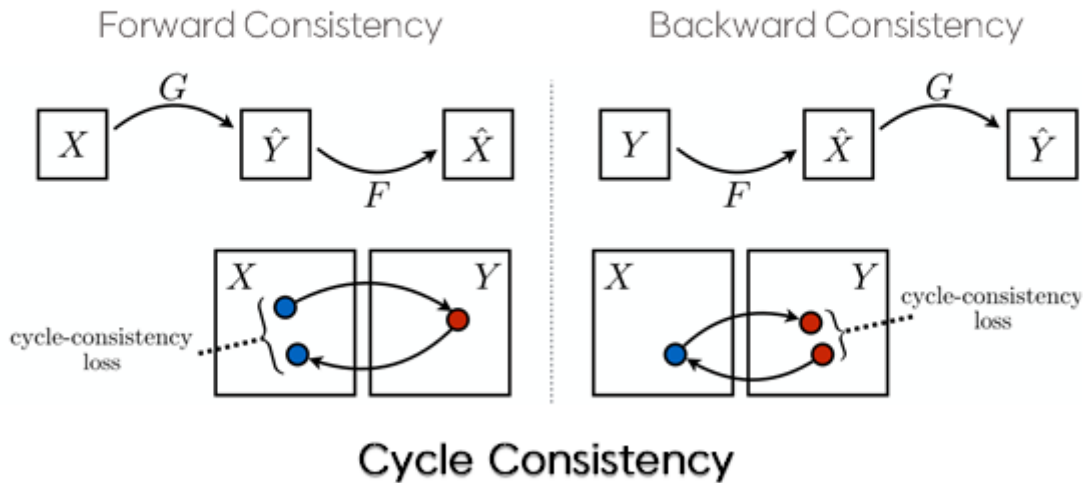


여기서는 기존의 generator G 이외에 **F라는 새로운 generator**가 등장하게 된다. 즉, 이 loss가 하고싶은 것은 이제 그림을 G 라는 generator를 이용해서 사진으로 바꾸고 그 바꾼 사진을 F 라는 generator를 이용해서 다시 그림으로 바꾸었을 때, 처음 주어진 그림과의 차이가 적어야 한다는 내용의 loss가 되는 것이다. 수식을 살펴보면 $G(x)$ 가 바로 generator가 그림을 사진으로 바꾼 이미지이고 그것이 다시 F 의 input으로 들어가서 바꾼 사진을 다시 원래 도메인인 그림으로 복원하면 그 결과와 처음 그림과의 차이를 loss로 이용하는 것이 된다. 그리고 이 loss를 cycle loss라고 부르게 된다. 그런데 이제 이걸 오로지 그림의 관점에서만 해석한 식이라고 볼 수 있다. 즉 두개의 도메인이 존재하는데 한 개의 도메인에서만 이 관계를 해석한 것이 되는 것이다.

Cycle Consistency

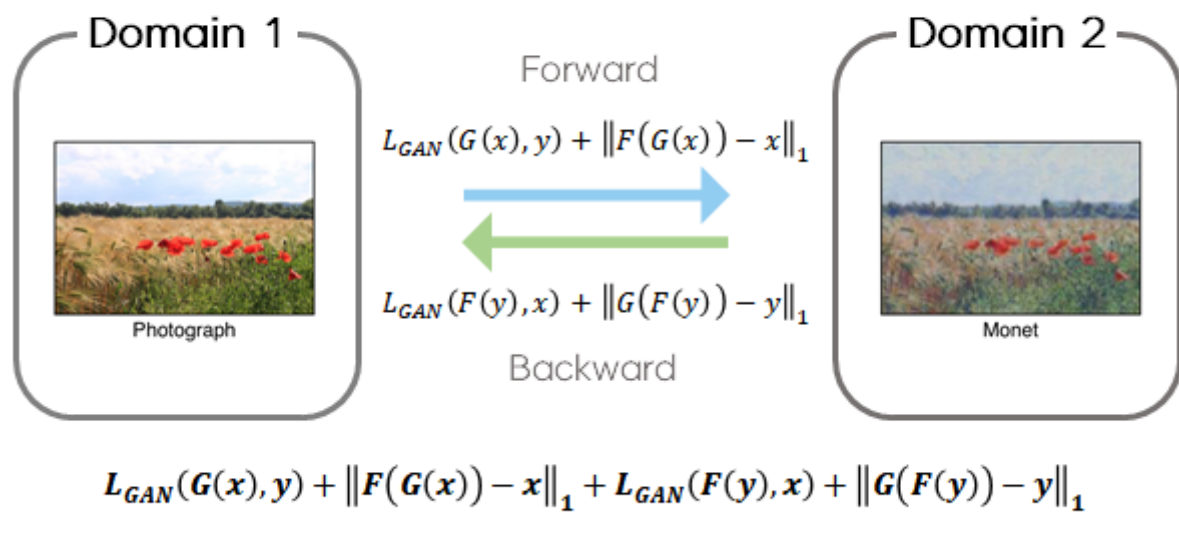
Cycle GAN

manypeople... 구독하기



따라서 cycle GAN은 두 도메인에서 모두 LOSS를 계산하게 된다. 즉 그림에서 사진으로 맵핑하는 동작 과정을 forward consistency라고 하고, 반대의 과정을 backward consistency라고 한다. 이때 Forward와 backward 각각 모두 generator를 거쳐서 한바퀴를 돌아오면 다시 처음 자기자리로 돌아와야 하기 때문에 ‘순환 일관성(Cycle Consistency)’ 라는 이름이 붙었고, cycle GAN에 서는 이러한 원리를 가져와서 손실함수를 만드는 것이다.

Cycle GAN



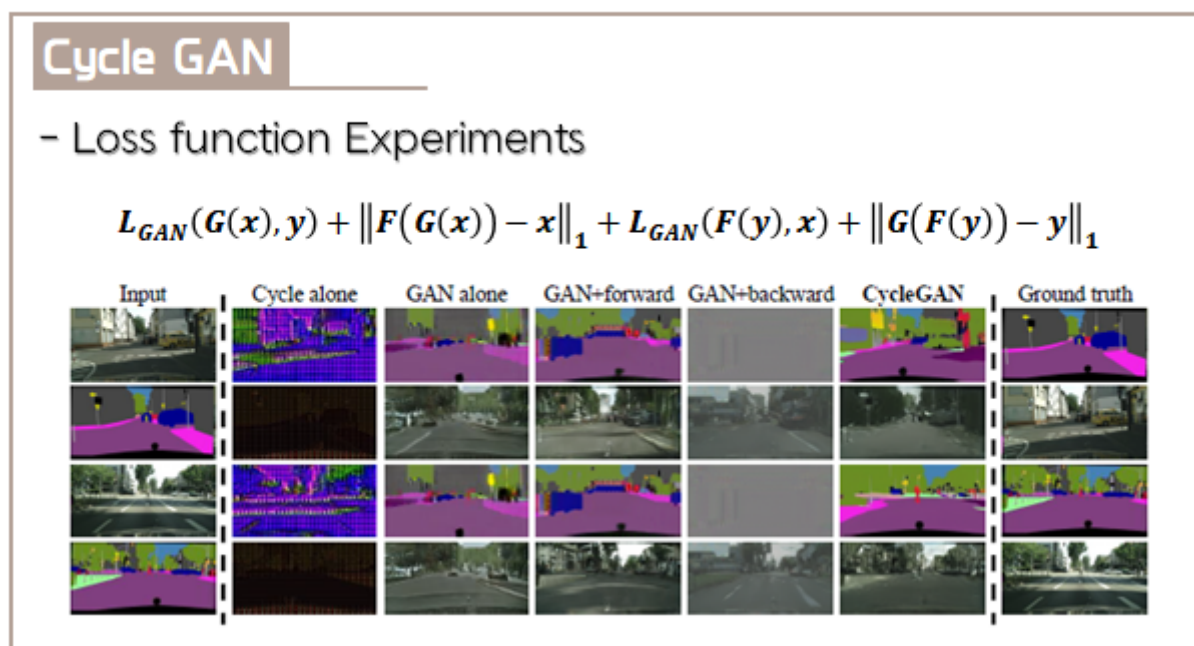
cycle GAN 최종 정리

최종적으로 정리를 해보면 다음과 같다. 계속해서 그림과 사진을 변환하는 예시를 보고있는데 여기서도 첫번째 도메인은 사진이고 두번째 도메인은 그림이 된다. 이때 먼저 사진의 관점에서 해석해보면 이 사진을 이용해 그림을 만들어야하기 때문에 이렇게 GAN LOSS가 이용이되고, 그렇게 해서 생

성된 그림이 다시 사진이 되었을 때 원래 사진과의 오차가 적어야하기 때문에 이렇게 뒤의 LOSS가 따라붙게 된다. 반대로 DOMAIN 2 즉 그림의 관점에서 해석하면 이와 정반대가 된다. manypeople... 구독하기
 그림에서 사진을 생성해야하므로 GAN LOSS가 붙고, 뒤에는 그림이 사진이 되고 그 사진이 다시 그림이 되었을 때 원래의 그림과의 차이가 적도록하는 loss가 오게된다.

이렇게 해서 두가지 domain 관점에서의 모든 loss들을 다 더해지게되면, 맨 아래의 **최종 CycleGAN의 LOSS**가 나오게 됩니다.

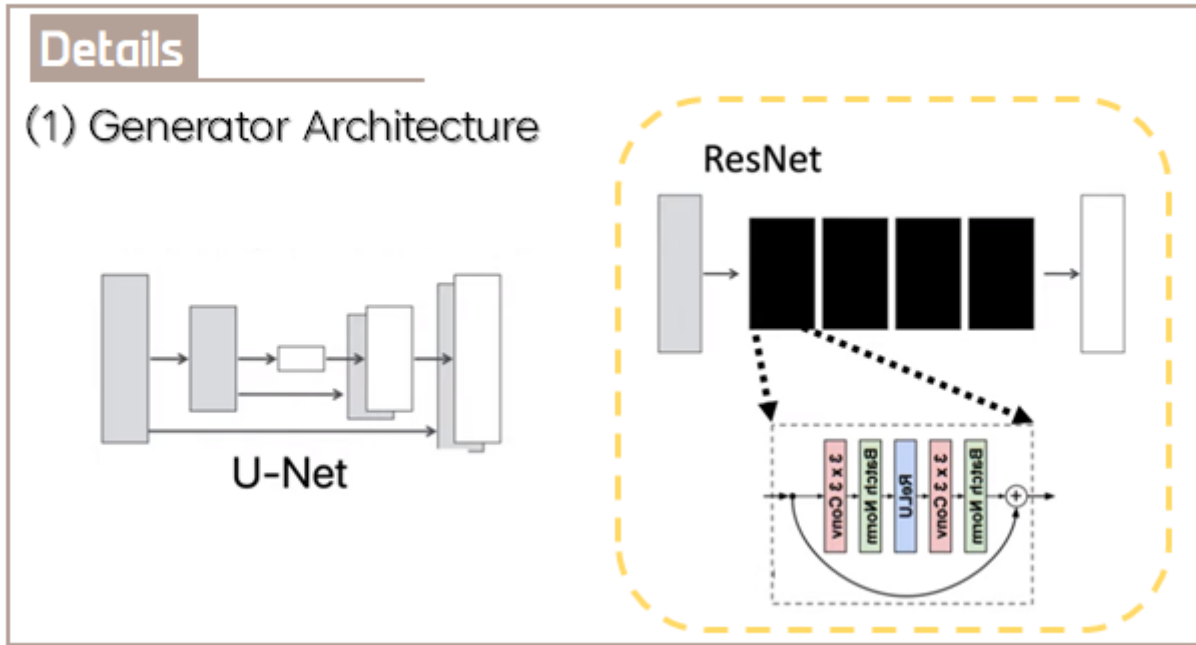
LOSS function Experiments



loss function 비교 실험

다음은 이렇게 만들어진 네 항의 LOSS를 모두 사용하는 것이 진짜 유의미한 결과를 내는지에 대해 실험을 진행한 모습이다. 실험 결과를 살펴보면, 이렇게 GAN LOSS만 사용했을 때, 순환 일관성 LOSS만 사용했을 때, GAN에 FORWARD, GAN에 BACKWARD만 사용했을 때 그리고 최종적으로 CYCLE GAN까지 이렇게 여러경우에 대해 LOSS 함수의 성능을 실험했는데 그림에서도 알 수 있듯이 모두 합한 CYCLE Gan의 경우가 가장 ground truth와 유사한 결과를 내는 것을 알 수 있다.

Generating High Quality Images using Cycle GAN



여기서부터는 Cycle GAN을 이용해서 좀더 고품질의 이미지를 생성하기 위해 논문에서 추가한 디테일들에 대해 간단히 살펴보겠다.

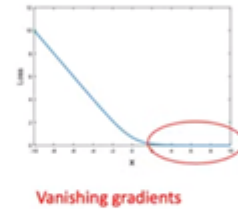
먼저 첫번째는 generator의 아키텍처에 관한 내용이다. Pix2pix generator의 경우에 U-NET을 사용해 Skip connection을 연결해서 처음 detail이 마지막 layer에 바로 전달될 수 있도록해 이미지의 디테일을 살렸지만, CYCLE GAN과같이 어느정도 비슷한 성격을 지닌 두 domain에 대해서는 skip connection이 너무 많이 반영되어 depth가 얇아 생성결과가 좋지 않게된다. 이를 개선하기 위해서 cycleGAN에서는 RESNET 아키텍처를 사용해서 RESIDUAL BLOCK을 통해 detail을 살리면서 모델의 depth또한 유지 할 수 있게 한다.

GAN LOSS FUNCTION

(2) GAN loss function

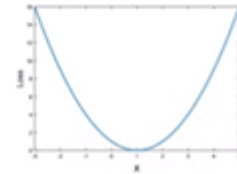
- GANs with cross-entropy loss

$$\mathcal{L}_{GAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{data}(y)} [\log D_Y(y)] + \mathbb{E}_{x \sim p_{data}(x)} [\log(1 - D_Y(G(x)))]$$



- Least square GANs

$$\mathcal{L}_{LSGAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{data}(y)} [(D_Y(y) - 1)^2] + \mathbb{E}_{x \sim p_{data}(x)} [D_Y(G(x))^2]$$



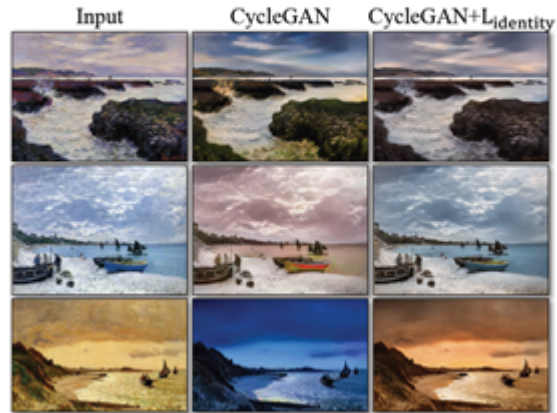
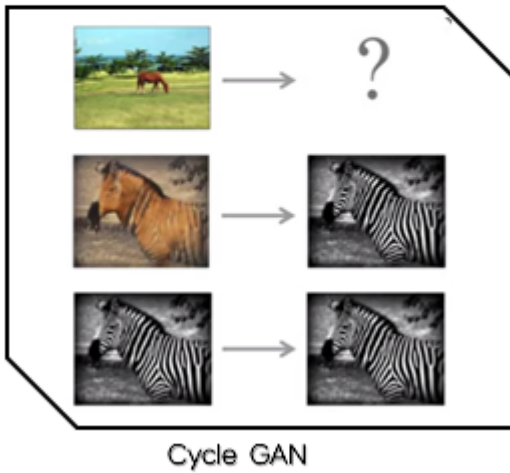
두 번째는 GAN의 LOSS 함수 이다.

기존 GAN에서는 Generator가 discriminator를 잘 속인 데이터 하나를 생성했다고 가정해보면, 가짜 이미지 분포가 실제 데이터 분포와 거리가 멀어도 discriminator를 속이기만 하면 되기 때문에 해당 모델에서는 학습이 잘 일어나지 않는 문제가 발생할 수 있다. 하지만 GAN이라는 모델 자체가 최대한 진짜 이미지 분포와 비슷한 이미지를 생성하는 것이 목표이기 때문에 이렇게 아래 위치한

LSGAN loss를 이용해서 진짜 데이터 분포로부터 거리가 멀면 가까워질수있도록 만들어서 VANISHING GRADIENT 문제를 해결할 수 있게 된다. (LS GAN에 대해서는 추가 내용을 더 공부해야 할 것 같다..!)

Identity LOSS

(3) Identity loss function

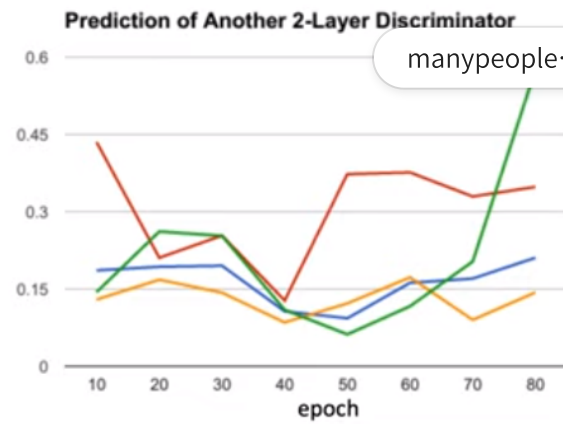
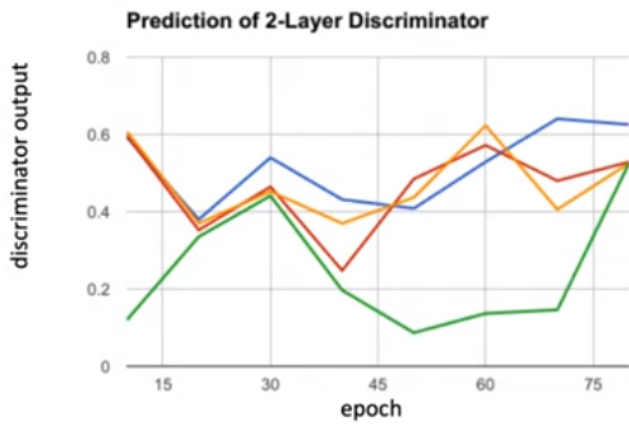


마지막으로는 추가적인 LOSS인 identity loss이다. 논문에 따르면 원래 이미지와 가짜 이미지의 차이인 L1 LOSS가 많으면 많을수록 좋다고 언급을 하고 있다. GAN LOSS로는 진짜 같은 이미지를 생성하는데만 집중시키고 L1 LOSS를 이용해서 DETAIL을 살린다는 아이디어 인 것이다.

말을 얼룩말로 바꾸는 예제를 통해서 identity loss가 무엇인지 살펴보자. 먼저 앞에서 살펴본 cycle loss는 말을 얼룩말로 바꾸고 이 변환한 얼룩말을 다시 말로 복원한 이미지와 진짜 말의 이미지의 차이였다면, identity loss는 input을 얼룩말로 바꾼 generator 여기서는 즉 F에게 input으로 그냥 얼룩말을 주게 되었을 때 얼마나 똑같은 얼룩말이 나올것인지 그 두이미지의 차이를 계산한 것이다.

오른쪽에 보이는 그림들이 identity loss를 추가했을 때와 그렇지 않았을 때의 결과를 비교하는 실험인데, identity loss를 추가해준 것이 보는 것과같이 진짜 이미지와 더욱 비슷한 색감을 가진 output을 냈음을 알 수 있다.

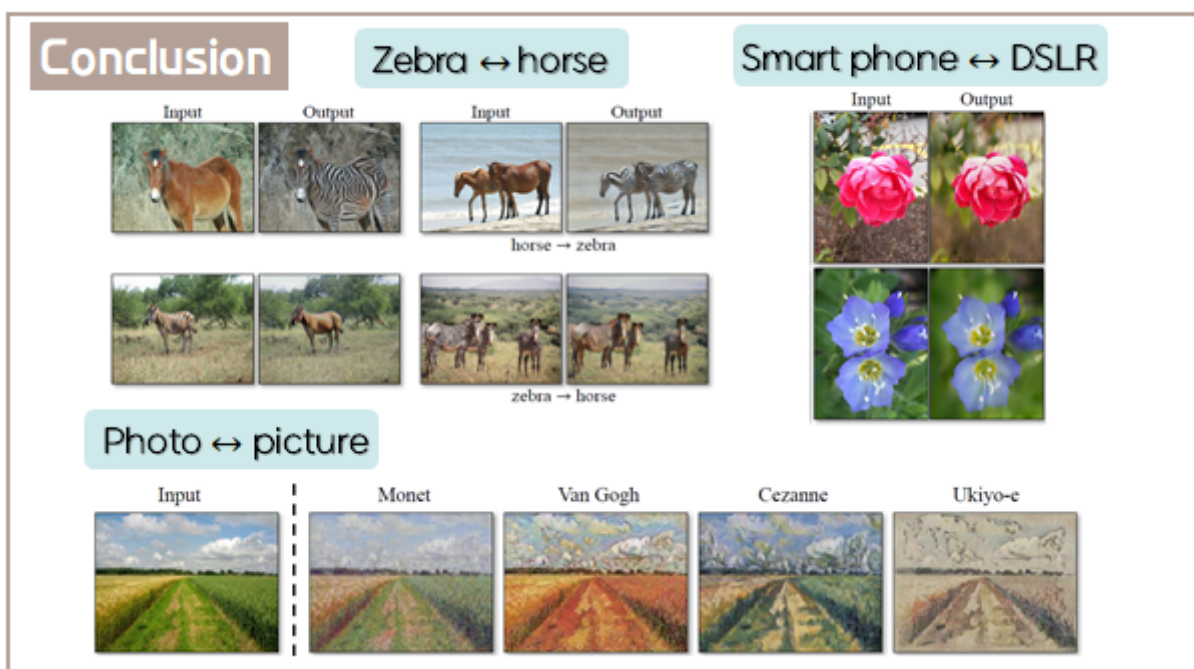
REPLAY BUFFER



manypeople... 구독하기

GAN은 실험을 진행하기 굉장히 어렵다. 여기 보이는 색깔은 서로 다른 샘플들을 의미하는데, 파란색 SAMPLE을 기준으로 그래프를 살펴보자. 이때 왼쪽 그래프에서는 Discriminator가 파란색 샘플에 대해 대체로 진짜라고 판단을 하고 있지만, 똑같은 아키텍처를 가지고 랜덤시드만 다르게 한 오른쪽 그래프에서는 파란색 샘플에 대해 대체적으로 가짜라고 판단을 하고 있다. 이러한 문제를 해결하는 방법으로는 discriminator를 여러개 생성해서 평균을 내는 방법이 있지만, 이러한 방법은 메모리를 너무 많이 잡아먹기 때문에 불가능하다. 그래서 cycle gan에서는 이러한 replay buffer라는 개념을 도입한다. 이전에 generator가 생성했던 이미지 히스토리를 일정개수만큼 저장해놓고 그걸 다시 discriminator에게 보여줌으로써 전에 봤던 것을 어떻게 반영했는지 알게 해주는 그러한 역할을 하는 buffer를 이용하는 것이다.

CONCLUSION



최종적으로 cycle gan을 이용해 생성한 여러가지 결과들이다. 이렇게 얼룩말과 말을 변환하는 이미지, 그림과 사진을 바꾸는 이미지 뿐만 아니라 이렇게 핸드폰으로 촬영한 사진을 D... manypeople... 구독하기 처럼 변환해줄 수도 있는 등 다양한 이미지 변환 문제에 CYCLE GAN을 적용할 수 있다는 것을 확인할 수 있다. 이외에 본 논문을 읽어보면 훨씬 더 다양한 결과들이 많기 때문에 다른 다양한 결과들이 궁금하다면 논문을 확인해보면 좋을 것 같다.

<참고 자료>

원 논문 :

[1703.10593] [Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks \(arxiv.org\)](#)

	Unpaired Image-to-Image Translation using Cycle-Consistent... arxiv.org
--	--

유튜브 Cycle GAN 논문 리뷰 : youtu.be/eCNITST_K-E

공감

구독하기

Tag

Computer Vision

CycleGAN

Deep Learning

GAN

image generator

Pix2Pix

이전글 [pix2pix] Image-to-Image Translation with Conditional Adversarial Networks 논문 정리

현재글 : [CycleGAN] Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks 논문 리뷰

manypeople... 구독하기

다음글 [정보이론] Entropy, Cross Entropy, KL-divergence 이해하기

관련글

[STMC-Transformer] Better Sign Language Translation...

2021.08.12

[LSGAN] Least Squares Generative Adversarial...

2021.04.21

[pix2pix] Image-to-Image Translation with Conditional...

2021.04.12

댓글 2

브라더 훈 2021.11.15 14:20 신고

안녕하세요, 포스팅에 사용하신 이미지와 일부 정리 내용들을 혹시 스터디 활동에서 정리겸 작성하는 위키독스 문서에 사용 가능할까요?(영리목적이 없고, 출처를 기재해 놓겠습니다!)

답글

DI_Lee 2021.11.15 15:23 신고

네!

이름

비밀번호

내용을 입력해주세요.



분류 전체보기 (18)

논문 정리 (10)

Computer Vision (0)

CS231n (0)

MACHINE LEARNING (1)

CS STUDY (4)

하루 3분 네트워크 교실 (2)

운영체제 (2)

SQL (0)

딥러닝 개념 공부 (3)

독일 기록 (0)

Tag

Pix2Pix, CNN, 하루 3분 네트워크 교실,

Cross entropy, CS, 딥러닝,

Generative Adversarial Network, 논문정리, Ai,

네트워크, Vae, 스레드, image generator,

Convolution Neural Network, generative model,

최근글 인기글

[Style Transfer] Image Style
Transfer Using Convolutional
Neural Ne...
2021.11.30 18:28

[RNN/LSTM/GRU] Recurrent
Neural Network 순환신경망
2021.11.10 14:58

[스레드] 스레드의 개념 및 멀티스레드
2021.10.02 19:36

최근댓글

네!
DI_Lee

네!
DI_Lee

안녕하세요, 포스팅에 사용하신 이미지와 일부 정...
브라더 훈

공지사항

Facebook Twitter

Archives

2021/11
2021/10
2021/09
2021/08
2021/04

〈 2022. 01 〉

일	월	화	수	목	금	토
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

manypeople... 구독하기

Total

3,028

Today : 22

Yesterday : 13

검색내용을 입력하세요.