Pix2Pix

(Image-to-Image Translation with Conditional Adversarial Networks)
Univercity of California, Berkeley
In CVPR 2017

2019.05.27

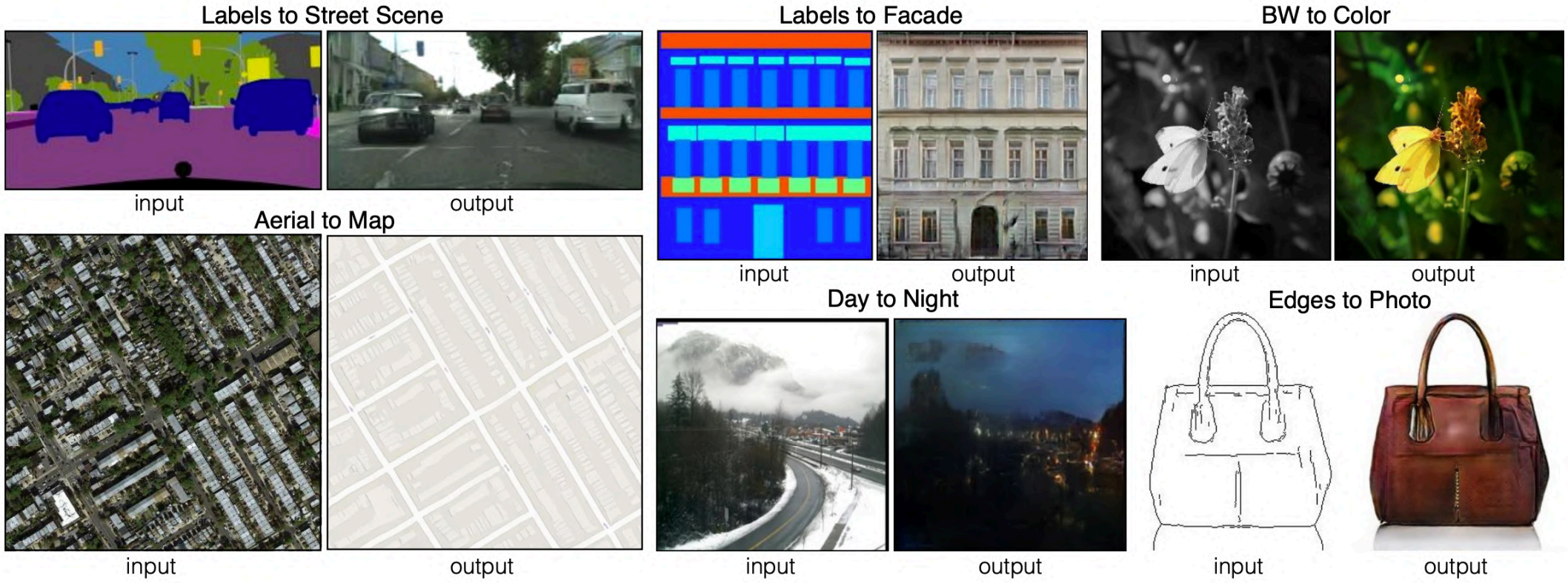
Hanyang univ. AILAB 정지은

Introduction

Introduction : background

- image processing, computer vision등에서 많은 문제들이 입력과 상응되는 아웃풋의 'translating' 문제로 귀결됨
- 기존에 image to image translation 하는 모델로 cGAN(conditional GAN)이 있으나, 범용적으로 활용하지 못하는 한계 존재

=> 여러가지 최적화 솔루션을 통해 만든 보다 범용적으로 사용가능한 모델 pix2pix 를 제안함



Introduction : provided solution

(1) Objective function : cGAN loss + L1 loss 사용

(2) Generator : encoder-decoder 구조로 U-net 사용

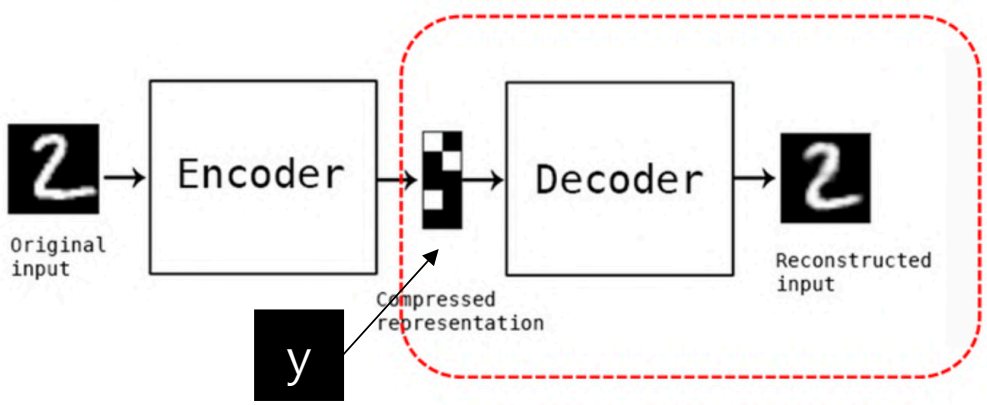
(3) Discriminator : PatchGAN 사용

Related work
(기저모델)

Related work : cGAN (conditional GAN)

cGAN IDEA

generator에서 latent variable에 원하는 condition을 부여해서 이미지 생성을 가이드 할 수 있다면, 원하는 방향으로 이미지를 생성할 수 있고 이를 다양한 분야에 응용 가능하지 않을까?



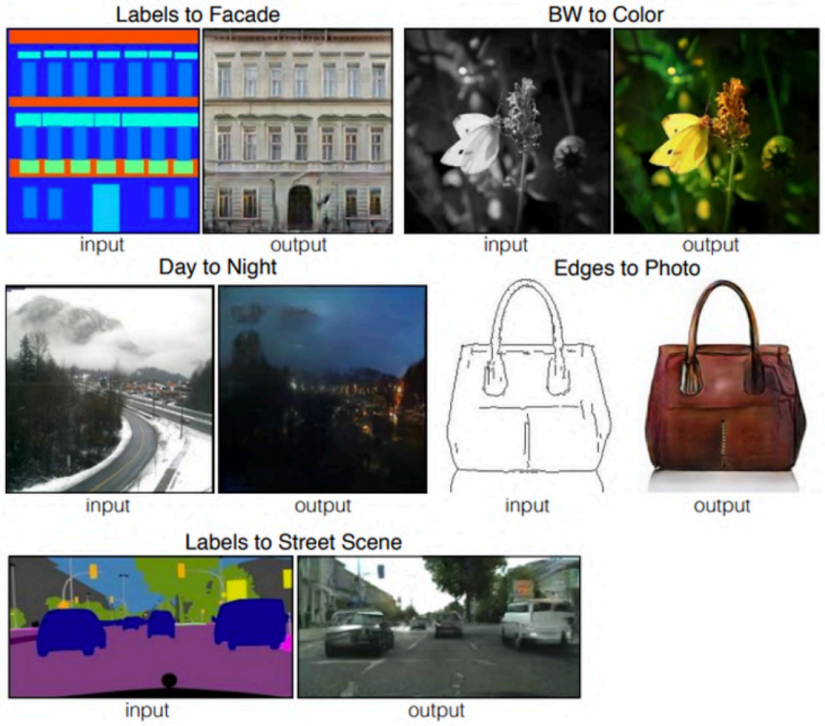
=> GAN에 간단히 condition y 를 추가해서 원하는 조건의 데이터(output)를 생성가능

ex) Pretrain된 text embedding을 조건 y로 추가해주면, 글에서 설명하는 이미지를 생성할 수 있음

Related work : cGAN (conditional GAN)

cGAN 의 한계 : condition y가 보통 one-hot encoding을 사용한 크지 않은 벡터 이므로
소수의 class를 갖는 이미지를 생성에만 가능

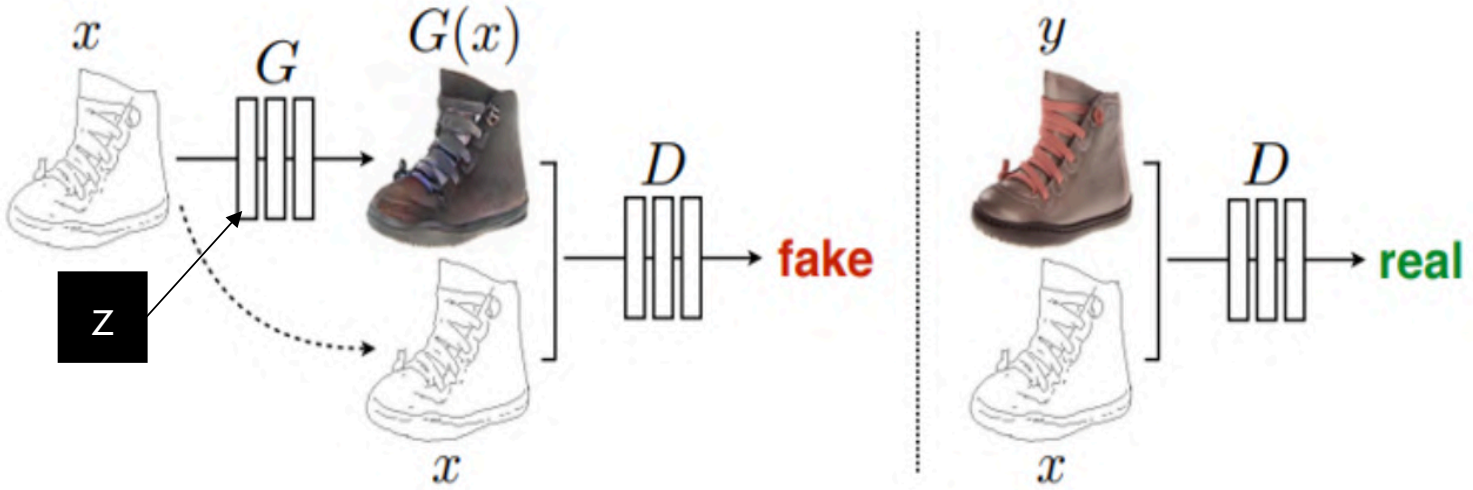
=> 저자는 보다 범용적인 image translation 모델(pix2pix)을 제안 함



Method

Method : Architecture (e.g. edge -> photo)

- 학습하기 위해서는 Pair data {x,y} 가 필요함



x : 외곽선 이미지(edge-map)

y : 목표로 하는 이미지

z : 노이즈 벡터

G(x) : 생성된 가짜 이미지

Method : (1) Objective function

- **일반적인 L1, L2 Loss 사용 시 문제점 : blur 현상**

(a) Input context

(c) Context Encoder
($L2$ loss)

각 픽셀별로 완벽한 답을 찾기보다 전체 픽셀의 관점에서 loss를 줄이려 하므로,

안전한 값으로 최적화하기 때문에 blur 현상 발생

=> **보다 사실적인 이미지(realistic photo)를 얻기 위해 GAN Loss 적용**

참고 : 기본적인 GAN Loss

Sample x from real data distribution

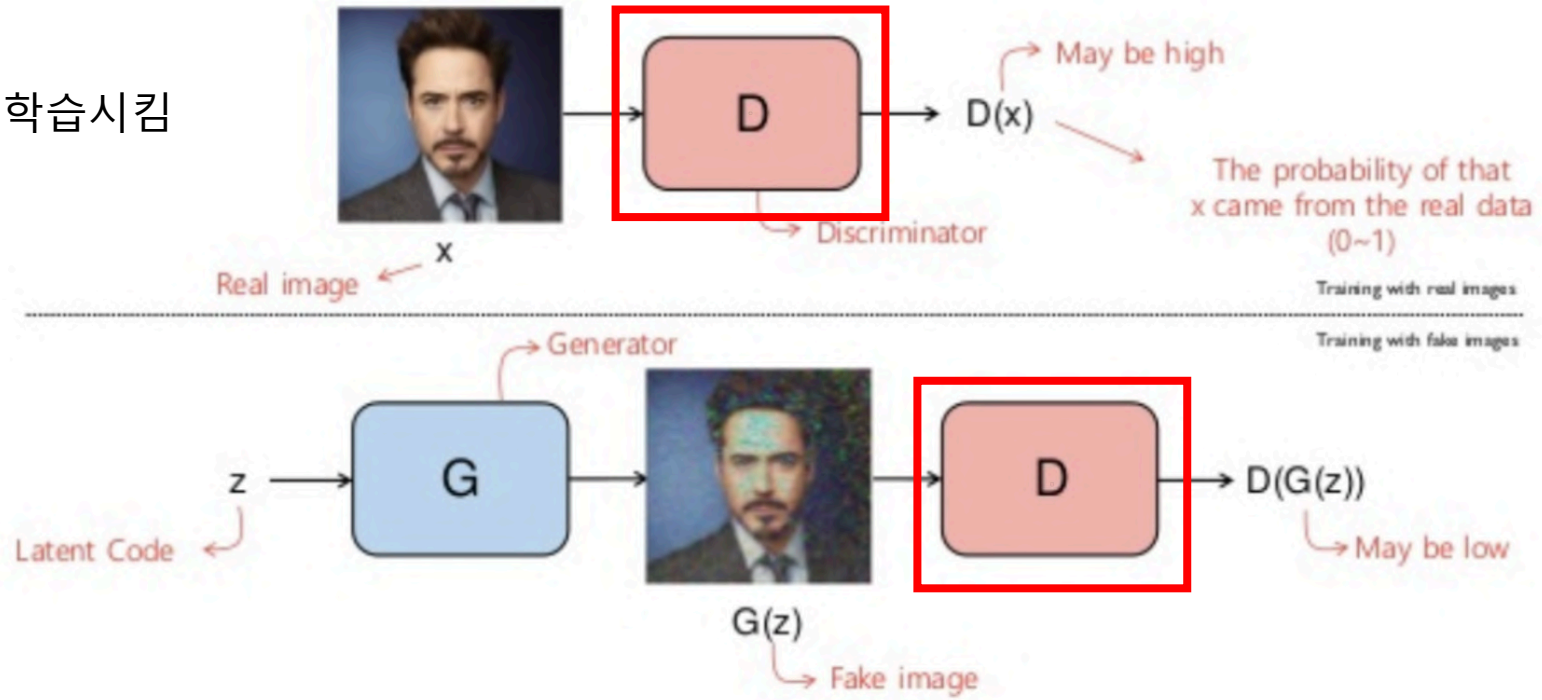
Sample z from Gaussian distribution

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[log\, D(x)] + E_{z\sim p_z(z)}[log(1 - D(G(z))]$$

D(x) = 1일때 Maximum

D(G(z)) = 0일때 Maximum

D를 먼저 학습시킴



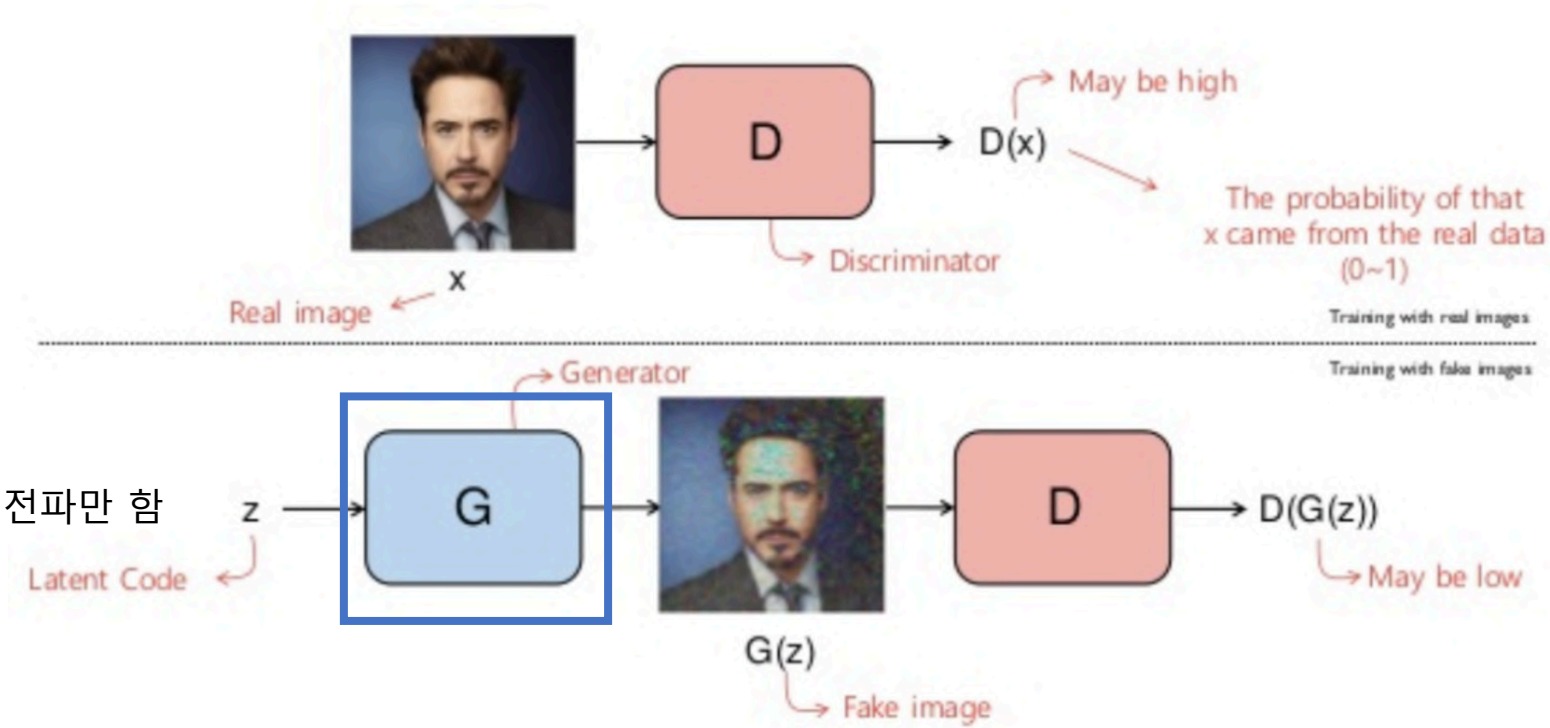
참고 : 기본적인 GAN Loss

G는 D(g(z))가 1에 가깝게 나오게 만드는게 목표

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[log\, D(x)] + E_{z\sim p_z(z)}[log(1 - D(G(z))]$$

G is independent

log(1- D(G(z))) = 0 이 되어야 Minimum



G 학습시에는
D는 학습하지 않고 역전파만 함

Method : (1) Objective function

X : 외곽선 이미지

y : 진짜 이미지

z : 랜덤한 노이즈벡터

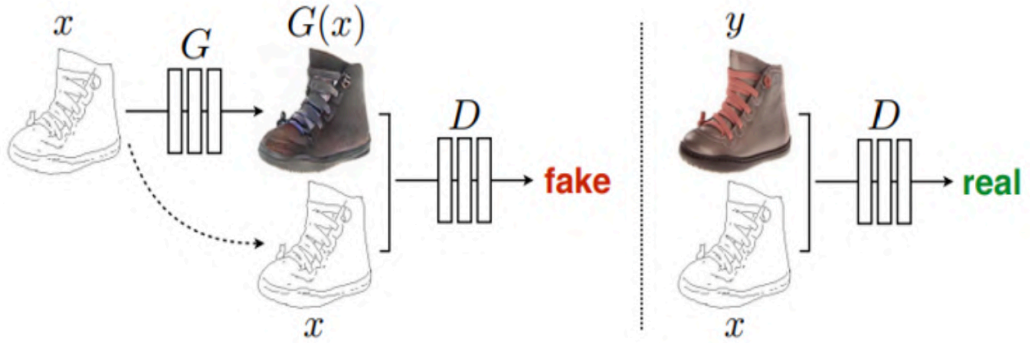
$$\mathcal{L}_{GAN}(G,D) = \mathbb{E}_y[\log D(y)] + \mathbb{E}_{x,z}[\log(1 - D(G(x,z))].$$

$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{x,z}[\log(1 - D(x,G(x,z))],$$

* Discriminator에 항상 pair로 들어감

GAN Generator : z -> y 생성

cGAN Generator : {x, z} -> y 생성



Method : (1) Objective function

진짜 처럼 보여주게 하는 GAN loss뿐 아니라,

Ground truth 이미지와 비슷한 이미지가 나오도록 L1 loss 도 추가

$$\mathcal{L}_{cGAN}(G,D) = \mathbb{E}_{x,y}[\log D(x,y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x,z))],$$

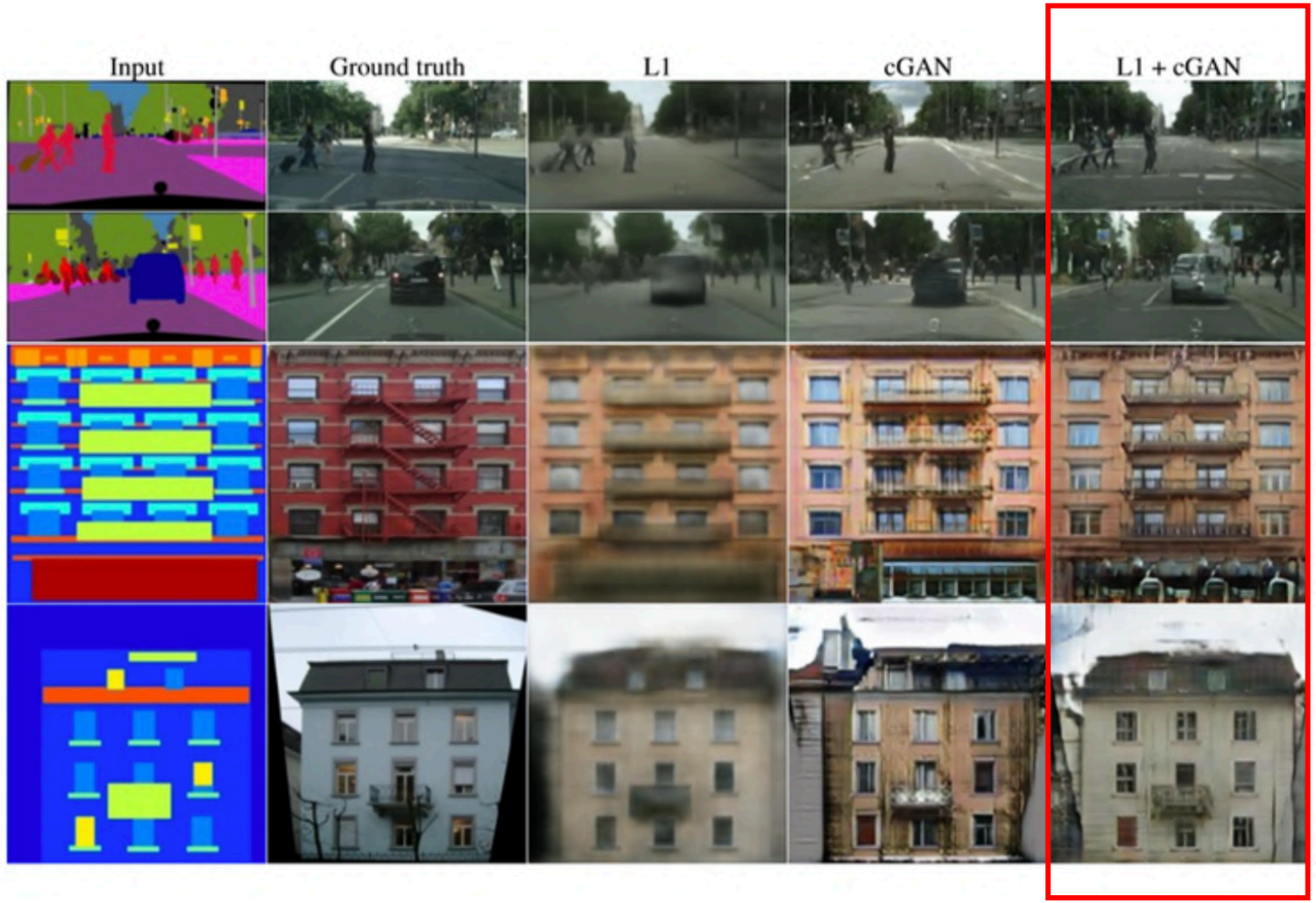
$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\|y - G(x,z)\|_1].$$

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G).$$

cGAN loss

L1

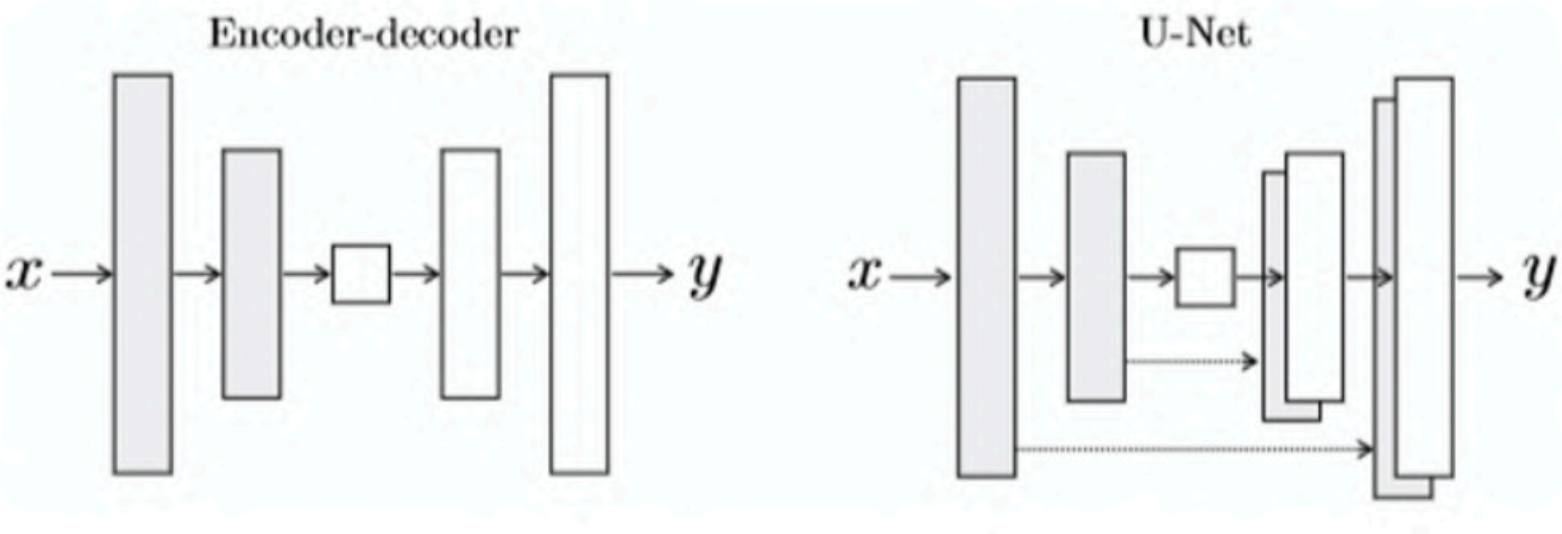
Method : (1) Objective function



Method : (2) Generator

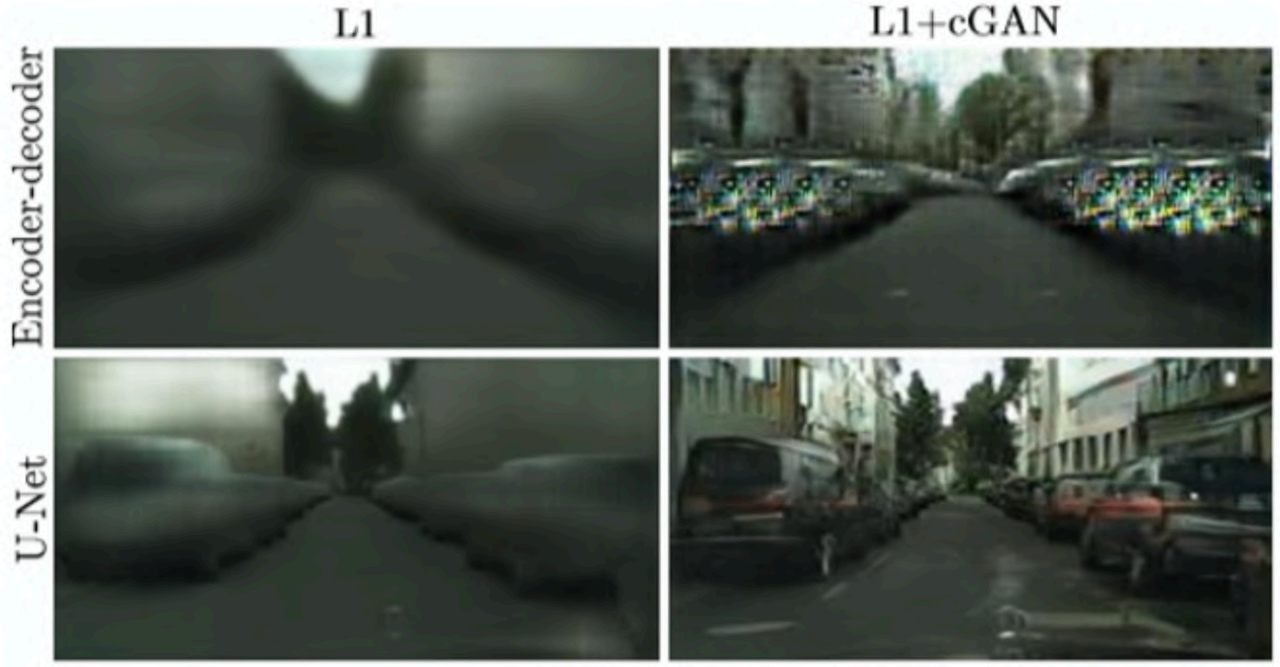
Generator architecture : U-net 구조 사용

Encoder-decoder 구조에서 영상크기를 줄였다가 다시 키우는 과정에서 detail이 사라지면서 영상이 흐려지는 문제를 피하기 위해 skip connection을 갖는 U-net 구조를 사용



- Encoding과정 : 그림의 맥락의 이해를 위해, feature map크기를 줄이면서 핵심 representation만 추출
- Decoding과정 : 원영상 복원 시 encoding에서 사라진 정보를 선명하게 하기 위해 skip connection 사용

Method : (2) Generator

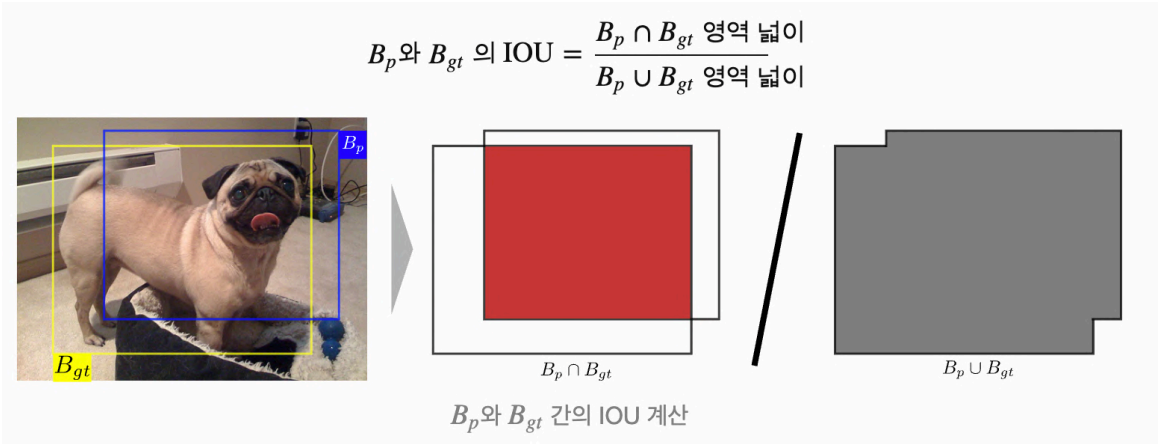


Loss	Per-pixel acc.	Per-class acc.	Class IOU
Encoder-decoder (L1)	0.35	0.12	0.08
Encoder-decoder (L1+cGAN)	0.29	0.09	0.05
U-net (L1)	0.48	0.18	0.13
U-net (L1+cGAN)	**0.55**	**0.20**	**0.14**

참고 : Measure

- 정성평가 : 아마존 Mechanical Turk (AMT) : 컬러가 풍부한가? 선명한가? 사람이 보기에 그럴듯한가?
- 정량평가 : IOU (intersection over union)

(1) Detection

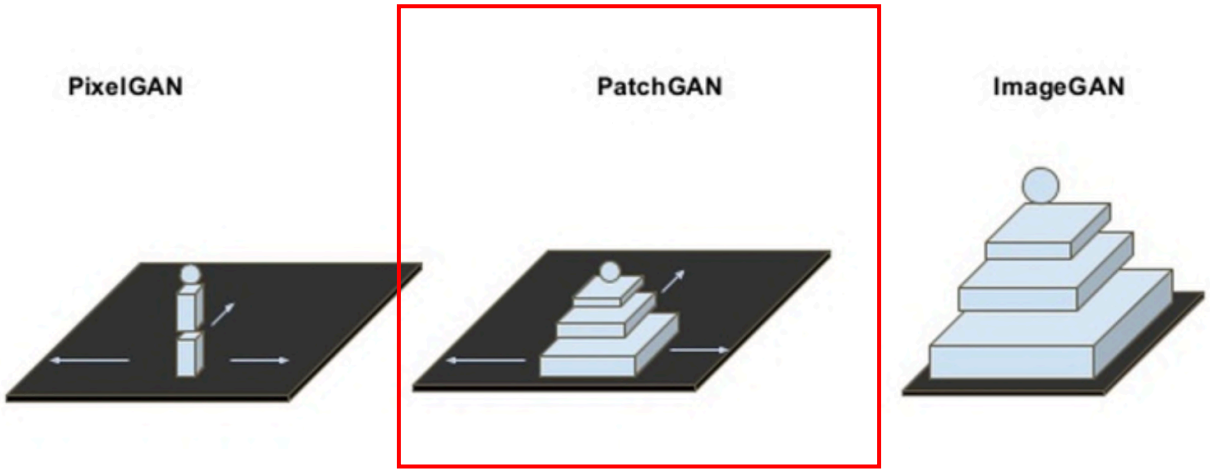


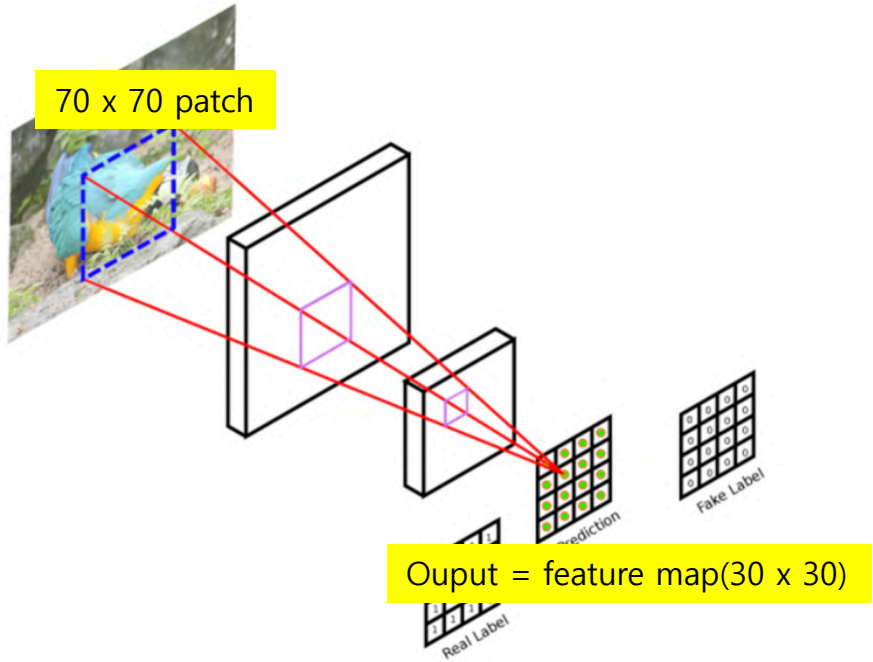
사물의 실제 위치를 나타내는 '*실제(ground truth; GT)*' **바운딩 박스 정보**가 이미지 레이블 상에 포함되어 있음

(2) Semantic segmentation

사물의 실제 위치를 나타내는 '*실제(GT)*' **마스크**가 이미지 레이블 상에 포함되어 있음

Method : (3) Discriminator





- 기존 GAN Discriminator : ImageGAN (이미지 **전체를 보고** 진짜인지 가짜인지 판별)
- Pix2pix Discriminator : patchGAN (이미지의 **Overlap되는 각 patch 별로** 진짜인지 가짜인지 판별한 후 평균)

왜? 픽셀들 간의 연관성(correlation)은 거리에 비례하여 작아지는 경향 존재하며 거리를 넘어서면 별 의미없어짐

=> 진짜같은 이미지를 생성하는 patch들이 많아지는 방향으로 학습을해서 generator의 성능을 더 올릴 수 있음

Method : (3) Discriminator

patchGAN 적용결과

Discriminator receptive field	Per-pixel acc.	Per-class acc.	Class IOU
1×1	0.39	0.15	0.10
16×16	0.65	0.21	**0.17**
70×70	**0.66**	**0.23**	**0.17**
286×286	0.42	0.16	0.11

Experiments

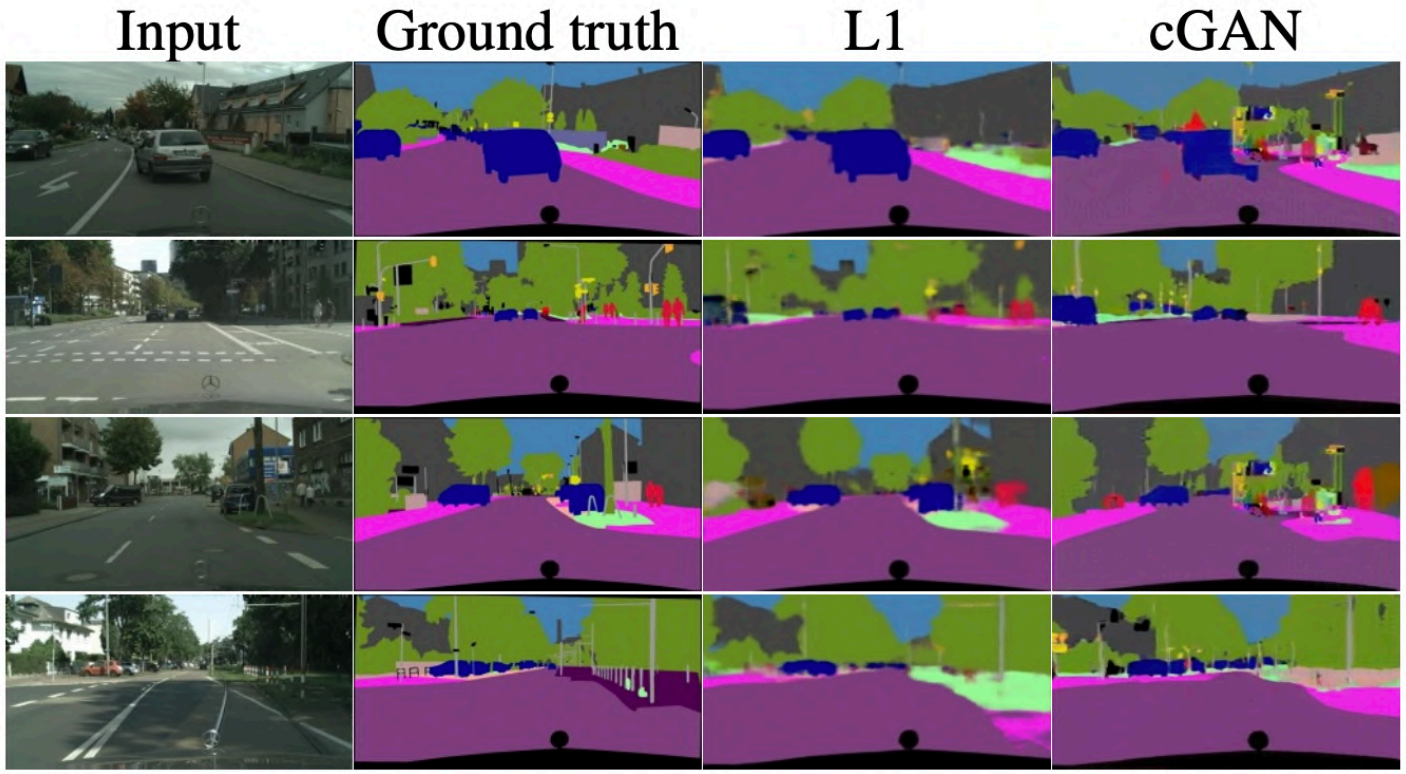
Experiments : perceptual validation

Figure 8: Example results on Google Maps at 512x512 resolution (model was trained on images at 256 × 256 resolution, and run convolutionally on the larger images at test time). Contrast adjusted for clarity.

	Photo → Map	**Map → Photo**
Loss	% Turkers labeled *real*	% Turkers labeled *real*
L1	2.8% ± 1.0%	0.8% ± 0.3%
L1+cGAN	6.1% ± 1.3%	**18.9% ± 2.5%**

Table 4: AMT “real vs fake” test on maps↔aerial photos.

Experiments : semantic segmentation



Loss	Per-pixel acc.	Per-class acc.	Class IOU
L1	**0.86**	**0.42**	**0.35**
cGAN	0.74	0.28	0.22
L1+cGAN	0.83	0.36	0.29

Table 6: Performance of photo→labels on cityscapes.

Experiments

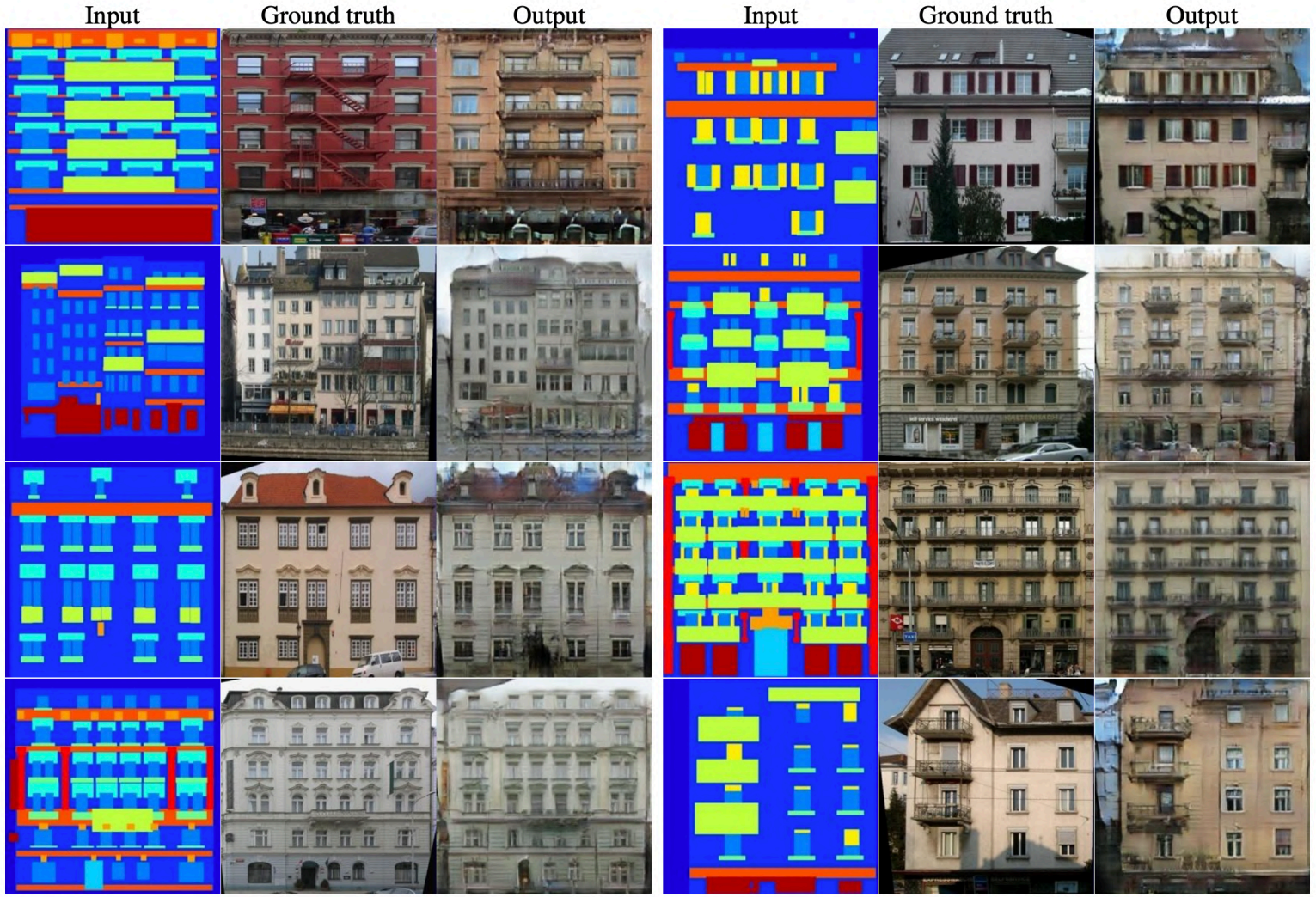


Figure 14: Example results of our method on facades labels→photo, compared to ground truth.

Experiments

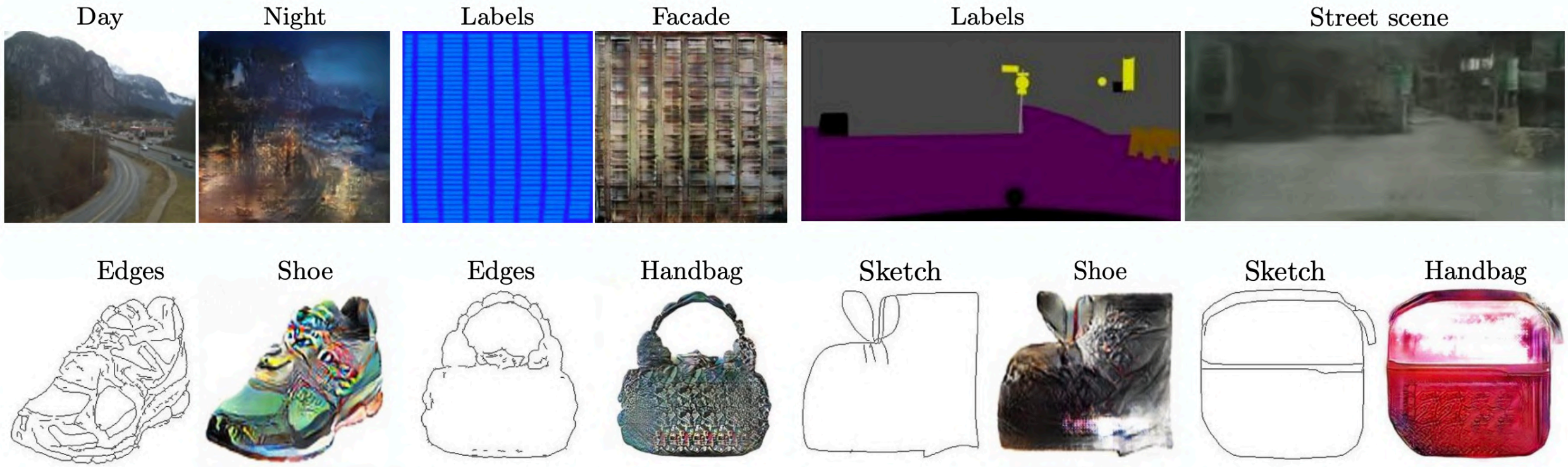


Figure 21: Example failure cases. Each pair of images shows input on the left and output on the right. These examples are selected as some of the worst results on our tasks. Common failures include artifacts in regions where the input image is sparse, and difficulty in handling unusual inputs. Please see `https://phillipi.github.io/pix2pix/` for more comprehensive results.

Conclusion

Conclusion

- cGAN(conditional GAN)이 있으나, 범용적으로 활용하지 못하는 한계 존재
- 여러가지 최적화 솔루션을 통해 만든 보다 범용적으로 사용가능한 모델인 pix2pix 를 제안함

Conclusion

- cGAN(conditional GAN)이 있으나, 범용적으로 활용하지 못하는 한계 존재
- 여러가지 최적화 솔루션을 통해 만든 보다 범용적으로 사용가능한 모델인 pix2pix 를 제안함

어떻게?

Conclusion

1. **Objective function : cGAN loss + L1 loss**

   => realistic photo를 얻기 위함

2. **Generator : U-net 구조 취함**

   => 전체 맥락 이해 및 복원을 위한 encoder-decoder 형태에서 이미지를 보다 선명하게 하기 위함

3. **Discriminator : PatchGAN 구조 사용**

   =>디테일한 부분에 집중해서 선명한 화질 얻도록함 + 연산속도 up

Conclusion

1. **Objective function : cGAN loss + L1 loss**

   => realistic photo를 얻기 위함

2. **Generator : U-net 구조 취함**

   => 전체 맥락 이해 및 복원을 위한 encoder-decoder 형태에서 이미지를 보다 선명하게 하기 위함

3. **Discriminator : PatchGAN 구조 사용**

   =>디테일한 부분에 집중해서 선명한 화질 얻도록함 + 연산속도 up

=> 범용적으로 사용 가능한 모델인 pix2pix 를 제안함

pix2pix paper summary

- 어떤 condition을 추가하여 원하는 output을 생성하는 방식의 cGAN에서 영감을 얻음
- Image to Image Mapping Network에서 Photo-realistic 을 추구하고자 GAN loss 도입
- U-Net, PatchGAN을 통해 제안한 모델(pix2pix)의 성능 최적화
- 기존 모델보다 범용적인 Image to Image translation net을 만듬
- 한계로는 Training Data가 Pair 로 존재해야 함

Thank You

Reference

Pixpix paper : https://arxiv.org/pdf/1611.07004.pdf

cGAN Paper : https://arxiv.org/abs/1411.1784

Semantic segmentation(FCN score) : https://people.eecs.berkeley.edu/~jonlong/long_shelhamer_fcn.pdf

IOU : http://research.sualab.com/introduction/2017/11/29/image-recognition-overview-2.html

http://blog.naver.com/PostView.nhn?blogId=laonple&logNo=221356582945&parentCategoryNo=&categoryNo=22&viewDate=&isShowPopularPosts=false&from=postView

http://blog.naver.com/PostView.nhn?blogId=laonple&logNo=221366130381&parentCategoryNo=&categoryNo=22&viewDate=&isShowPopularPosts=false&from=postView