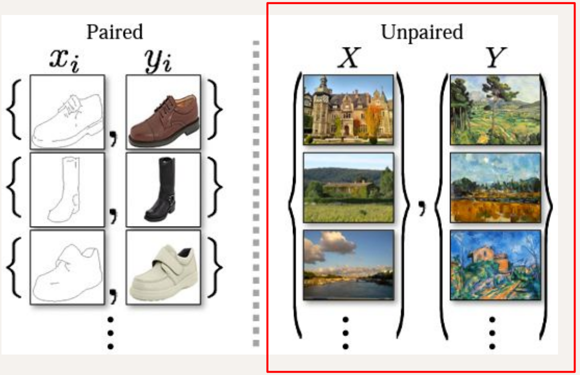
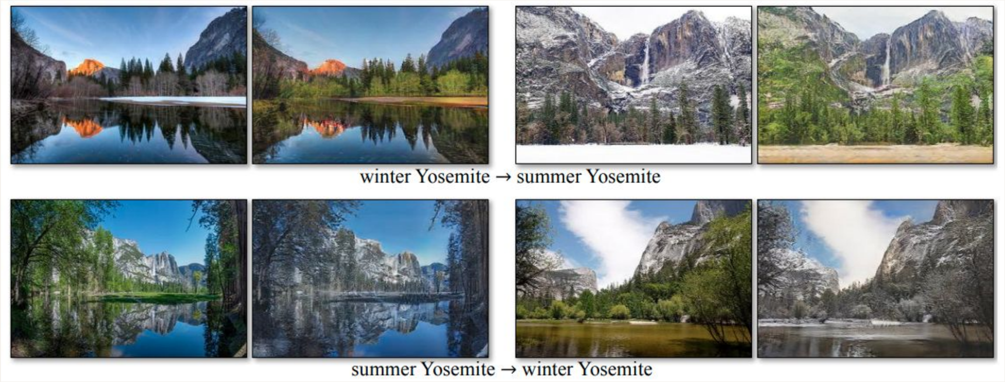
Cycle GAN

발표자 : 문경환

Abstract

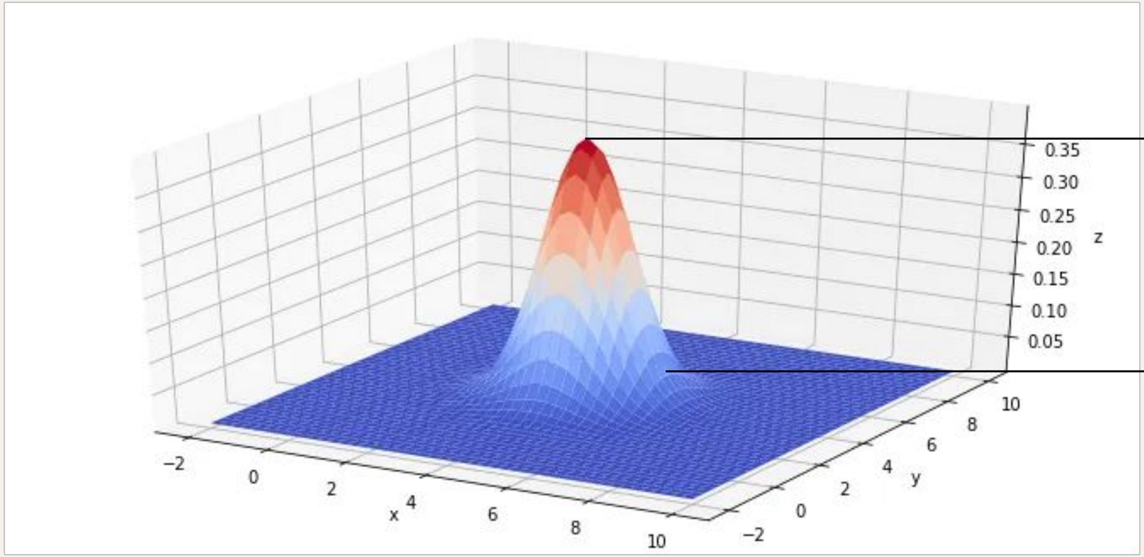
- Unpaired 데이터셋으로 학습이 가능한 image-to-image translation 메서드.
- 기존의 Image-to-Image 모델에서 도메인 간의 학습이 이루어지지 않을 수 있었던 문제 해결.
- Cycle-consistency loss를 통해 다양한 Task에서 좋은 결과를 도출.





Generative model

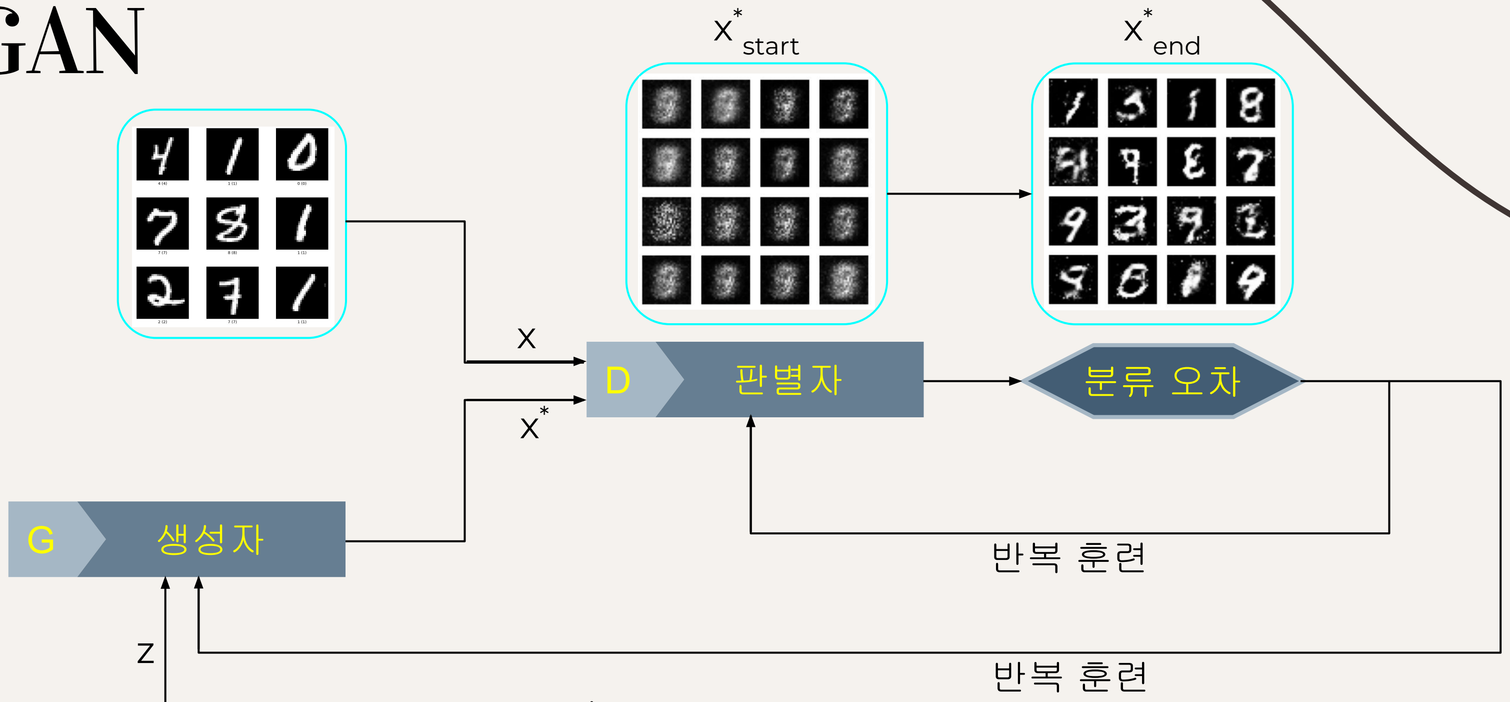
다변수 확률분포



출처 : Aly Shmahell, medium

출처 : analyticsvidhya

GAN



Loss Function

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

CGAN → Pix2Pix → CycleGAN

CGAN → Pix2Pix → CycleGAN

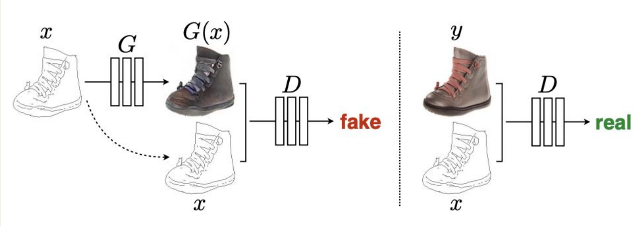
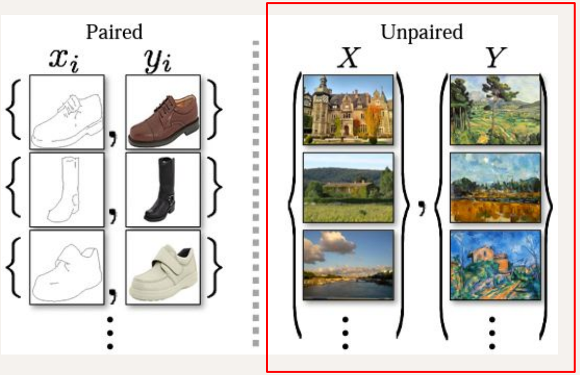


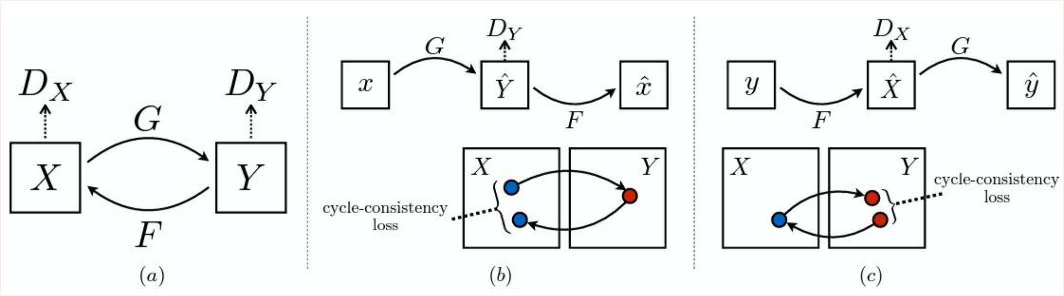
Figure 4: Different losses induce different quality of results. Each column shows results trained under a different loss. Please see `https://phillipi.github.io/pix2pix/` for additional examples.

CGAN → Pix2Pix → CycleGAN

CycleGAN

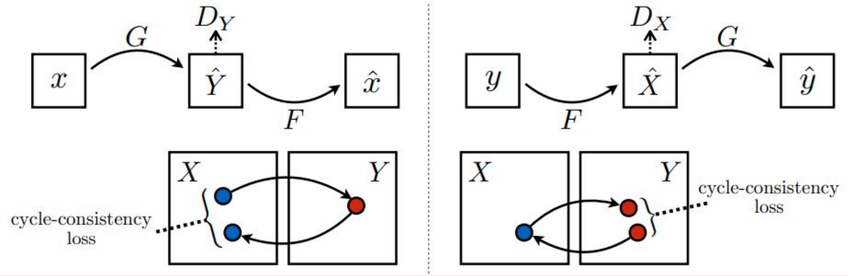
- 도메인 X를 도메인 Y로 변환하였다가 다시 도메인 X로 변환시키며 이미지가 다시 복원될 수 있도록 학습함.
- cycle-consistency loss를 사용한 학습으로 원본을 복원하는 능력을 향상.





Cycle-consistent loss

- 한 사이클을 돌아 이미지가 생성 되었을 때에, 원본 이미지와 얼마나 달라졌는지를 평가.
- Loss의 값이 작을수록 원본 이미지와의 차이가 적다고 볼 수 있음.



$$
\begin{aligned}
\mathcal{L}_{\text{cyc}}(G, F) = \; & \mathbb{E}_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] \\
& + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1].
\end{aligned}
$$

cycle-consistency loss

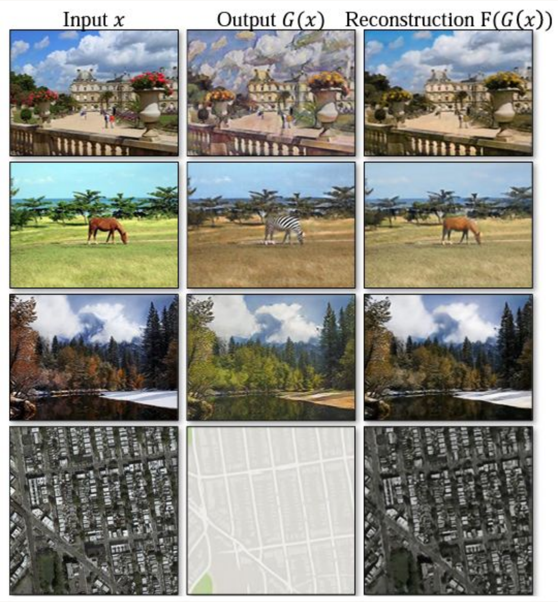
Full Objective

- 각각의 GAN loss와 cycle-consistency loss의 합을 최소화하는 것이 generator의 목표.

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

$$\begin{aligned}\mathcal{L}(G, F, D_X, D_Y) =& \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) \\ &+ \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) \\ &+ \lambda \mathcal{L}_{\text{cyc}}(G, F),\end{aligned}$$

$$G^*, F^* = \arg\min_{G,F} \max_{D_x, D_Y} \mathcal{L}(G, F, D_X, D_Y).$$



Network Architecture & Training

- 여러 residual block을 사용
- Instance Normalization 사용
- Discriminator은 이미지를 여러 패치로 나누어서 진짜/가짜 구분을 수행
- Cross-entropy 기반의 loss에서 mse 기반의 loss를 사용함
    - Mse 기반의 loss가 더욱 학습이 잘되고, 퀄리티가 좋은 이미지를 만들어냄
- 50개의 이전에 생성된 이미지를 저장해두고, Discriminator을 업데이트함
    - 모델 oscillation을 줄임

Performance

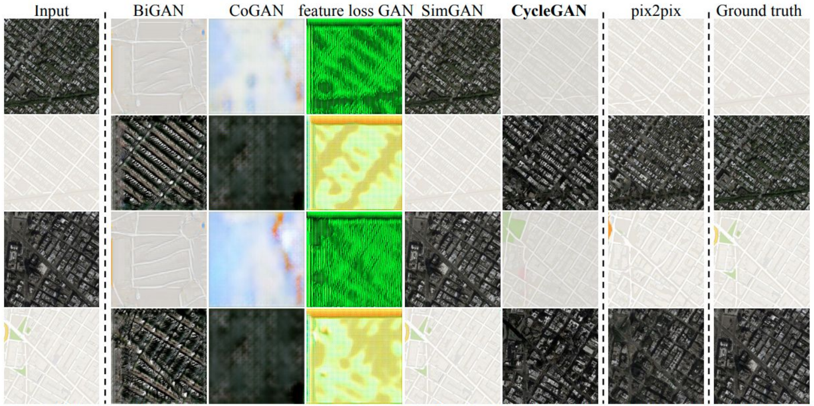


Figure 6: Different methods for mapping aerial photos↔maps on Google Maps. From left to right: input, BiGAN/ALI [7, 9], CoGAN [32], feature loss + GAN, SimGAN [46], CycleGAN (ours), pix2pix [22] trained on paired data, and ground truth.

Loss	**Map → Photo** % Turkers labeled *real*	**Photo → Map** % Turkers labeled *real*
CoGAN [32]	0.6% ± 0.5%	0.9% ± 0.5%
BiGAN/ALI [9, 7]	2.1% ± 1.0%	1.9% ± 0.9%
SimGAN [46]	0.7% ± 0.5%	2.6% ± 1.1%
Feature loss + GAN	1.2% ± 0.6%	0.3% ± 0.2%
CycleGAN (ours)	**26.8% ± 2.8%**	**23.2% ± 3.4%**

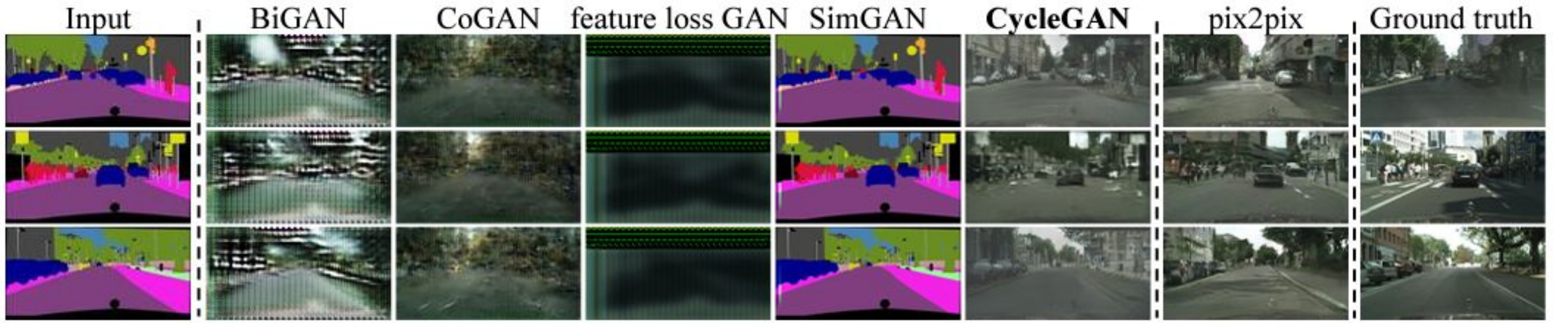
Table 1: AMT "real vs fake" test on maps↔aerial photos at 256 × 256 resolution.

Loss	Per-pixel acc.	Per-class acc.	Class IOU
CoGAN [32]	0.40	0.10	0.06
BiGAN/ALI [9, 7]	0.19	0.06	0.02
SimGAN [46]	0.20	0.10	0.04
Feature loss + GAN	0.06	0.04	0.01
CycleGAN (ours)	**0.52**	**0.17**	**0.11**
pix2pix [22]	0.71	0.25	0.18

Table 2: FCN-scores for different methods, evaluated on Cityscapes labels→photo.

Loss	Per-pixel acc.	Per-class acc.	Class IOU
CoGAN [32]	0.45	0.11	0.08
BiGAN/ALI [9, 7]	0.41	0.13	0.07
SimGAN [46]	0.47	0.11	0.07
Feature loss + GAN	0.50	0.10	0.06
CycleGAN (ours)	**0.58**	**0.22**	**0.16**
pix2pix [22]	0.85	0.40	0.32

Table 3: Classification performance of photo→labels for different methods on cityscapes.



Q&A