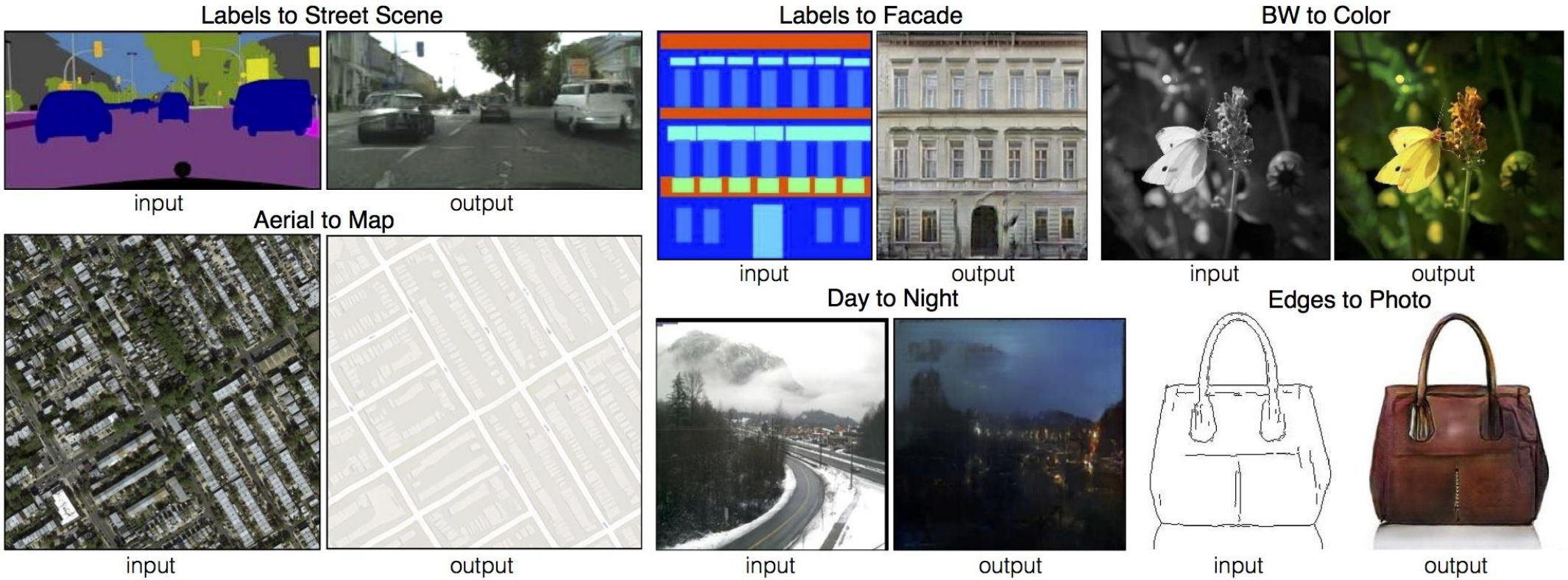
Chapter 06. 무엇이든 진짜처럼 생성하는 생성 모델(Generative Networks)

이미지 변환 모델

Pix2Pix



흥미로운 영상 변형 모델 Pix2Pix의 구조를 알아보자.

Conditional GAN

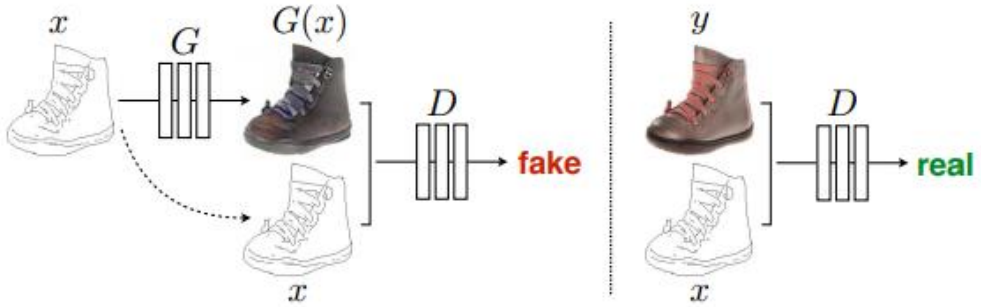


Figure 2: Training a conditional GAN to map edges→photo. The discriminator, D, learns to classify between fake (synthesized by the generator) and real {edge, photo} tuples. The generator, G, learns to fool the discriminator. Unlike an unconditional GAN, both the generator and discriminator observe the input edge map.

Pix2Pix에서는 Conditional GAN 구조를 채용하고 있다.

Discriminator의 입력으로 Generator 출력과 입력을 Pair로 함께 넣는 특징이 있다.

Loss Function

3.1. Objective

The objective of a conditional GAN can be expressed as

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

where G tries to minimize this objective against an adversarial D that tries to maximize it, i.e. $G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D)$.

To test the importance of conditioning the discriminator, we also compare to an unconditional variant in which the discriminator does not observe x:

$$\mathcal{L}_{GAN}(G,D) = \mathbb{E}_{y}[\log D(y)] + \mathbb{E}_{x,z}[\log(1 - D(G(x,z))]. \quad (2)$$

Previous approaches have found it beneficial to mix the GAN objective with a more traditional loss, such as L2 distance [43]. The discriminator's job remains unchanged, but the generator is tasked to not only fool the discriminator but also to be near the ground truth output in an L2 sense. We also explore this option, using L1 distance rather than L2 as L1 encourages less blurring:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\|y - G(x,z)\|_1]. \quad (3)$$

Our final objective is

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G). \quad (4)$$

GAN Loss에 L-1 Loss를 추가하여 Content Loss를 추가하였다.

U-Net Architecture

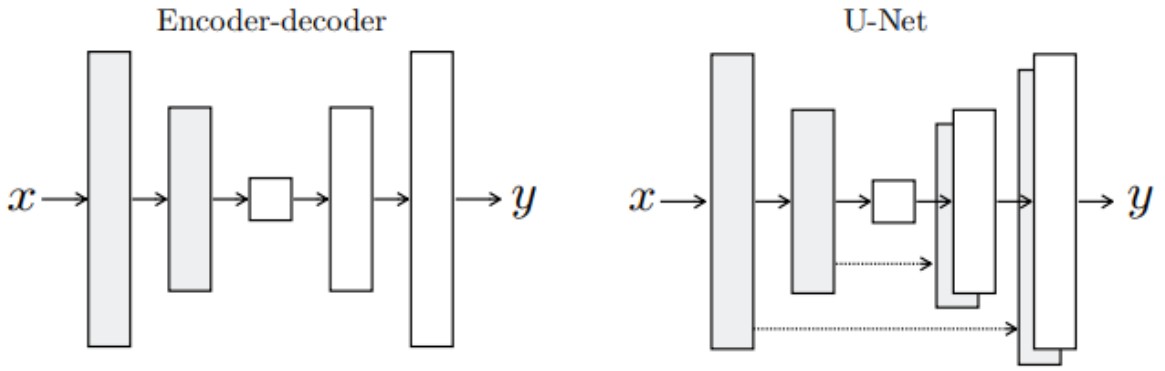


Figure 3: Two choices for the architecture of the generator. The "U-Net" [50] is an encoder-decoder with skip connections between mirrored layers in the encoder and decoder stacks.

U-Net 구조를 채용하여 Encoder-Decoder 구조에 해상력을 더했다.

Results w/ Different Losses

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.

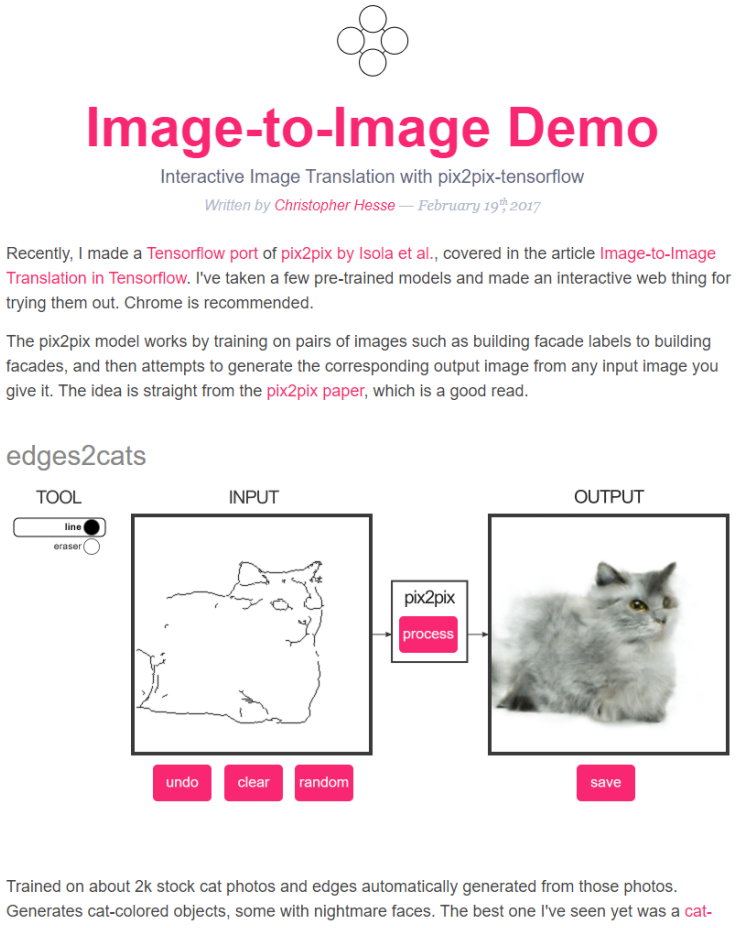
Loss별로 다른 결과가 나타난다. GAN의 단점을 L1 Loss가 커버하는걸 볼 수 있다.

Results w/ Different Parameters

Figure 6: Patch size variations. Uncertainty in the output manifests itself differently for different loss functions. Uncertain regions become blurry and desaturated under L1. The 1x1 PixelGAN encourages greater color diversity but has no effect on spatial statistics. The 16x16 PatchGAN creates locally sharp results, but also leads to tiling artifacts beyond the scale it can observe. The 70×70 PatchGAN forces outputs that are sharp, even if incorrect, in both the spatial and spectral (colorfulness) dimensions. The full 286×286 ImageGAN produces results that are visually similar to the 70×70 PatchGAN, but somewhat lower quality according to our FCN-score metric (Table 3). Please see `https://phillipi.github.io/pix2pix/` for additional examples.

Discriminator의 Receptive Field에 따른 해상도 차이를 볼 수 있다. 70x70 수준이면 적절하다고 보았다.

Interactive Demos



다양한 재미있는 데모가 마련되어 있다. 직접 시도해 보자!

https://affinelayer.com/pixsrv/