

Image – to – Image Translation

2016. 11. 28

**Tensorflow study group
Hyungjoo Cho**

Image-to-Image translation with Conditional Adversarial Networks

Image-to-Image Translation with Conditional Adversarial Networks

Phillip Isola

Jun-Yan Zhu

Tinghui Zhou

Alexei A. Efros

Berkeley AI Research (BAIR) Laboratory
University of California, Berkeley

{isola, junyanz, tinghuiz, efros}@eecs.berkeley.edu

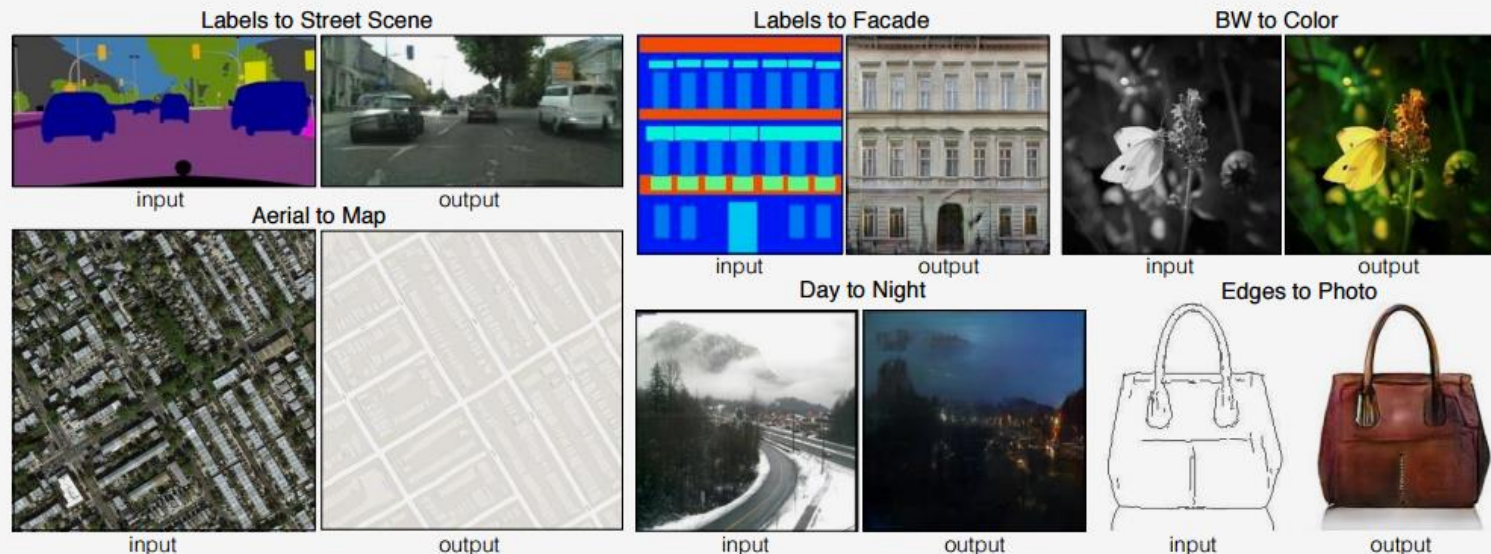


Figure 1: Many problems in image processing, graphics, and vision involve translating an input image into a corresponding output image. These problems are often treated with application-specific algorithms, even though the setting is always the same: map pixels to pixels. Conditional adversarial nets are a general-purpose solution that appears to work well on a wide variety of these problems. Here we show results of the method on several. In each case we use the same architecture and objective, and simply train on different data.

Image-to-Image translation with Conditional Adversarial Networks

Image-to-Image Translation with Conditional Adversarial Networks

Phillip Isola

Jun-Yan Zhu

Tinghui Zhou

Alexei A. Efros

Berkeley AI Research (BAIR) Laboratory
University of California, Berkeley

{isola, junyanz, tinghuiz, efros}@eecs.berkeley.edu

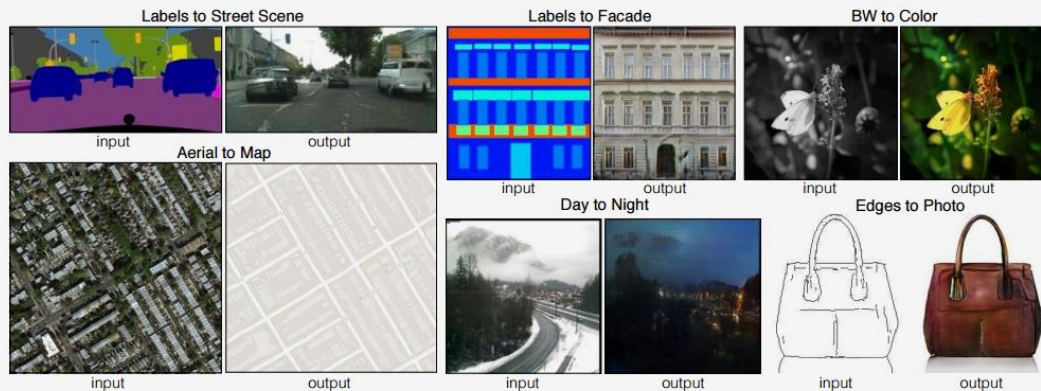


Figure 1: Many problems in image processing, graphics, and vision involve translating an input image into a corresponding output image. These problems are often treated with application-specific algorithms, even though the setting is always the same: map pixels to pixels. Conditional adversarial nets are a general-purpose solution that appears to work well on a wide variety of these problems. Here we show results of the method on several. In each case we use the same architecture and objective, and simply train on different data.

- Mapping photographs to edges, segments, or semantic labels
- Mapping labels or sparse user input to realistic images
- Mapping grayscale to color
- Bidirection mapping day to night
- Image inpainting
- Etc
- → **Predict pixels from pixels**

Generative Adversarial Nets

**Ian J. Goodfellow, Jean Pouget-Abadie*, Mehdi Mirza, Bing Xu, David Warde-Farley,
Sherjil Ozair,[†] Aaron Courville, Yoshua Bengio[‡]**

Département d'informatique et de recherche opérationnelle
Université de Montréal
Montréal, QC H3C 3J7

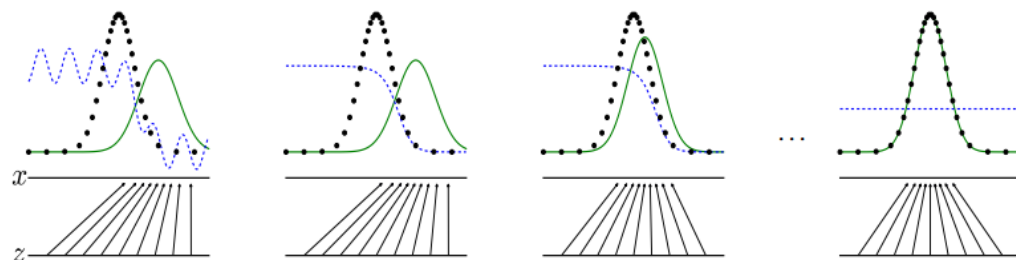
Abstract

We propose a new framework for estimating generative models via an adversarial process, in which we simultaneously train two models: a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample came from the training data rather than G . The training procedure for G is to maximize the probability of D making a mistake. This framework corresponds to a minimax two-player game. In the space of arbitrary functions G and D , a unique solution exists, with G recovering the training data distribution and D equal to $\frac{1}{2}$ everywhere. In the case where G and D are defined by multilayer perceptrons, the entire system can be trained with backpropagation. There is no need for any Markov chains or unrolled approximate inference networks during either training or generation of samples. Experiments demonstrate the potential of the framework through qualitative and quantitative evaluation of the generated samples.

Generative Adversarial Nets

D and G play the following two-player minimax game with value function $V(G, D)$

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



Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k , is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ from noise prior $p_g(\mathbf{z})$.
- Sample minibatch of m examples $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\mathbf{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m \left[\log D(\mathbf{x}^{(i)}) + \log(1 - D(G(\mathbf{z}^{(i)}))) \right].$$

end for

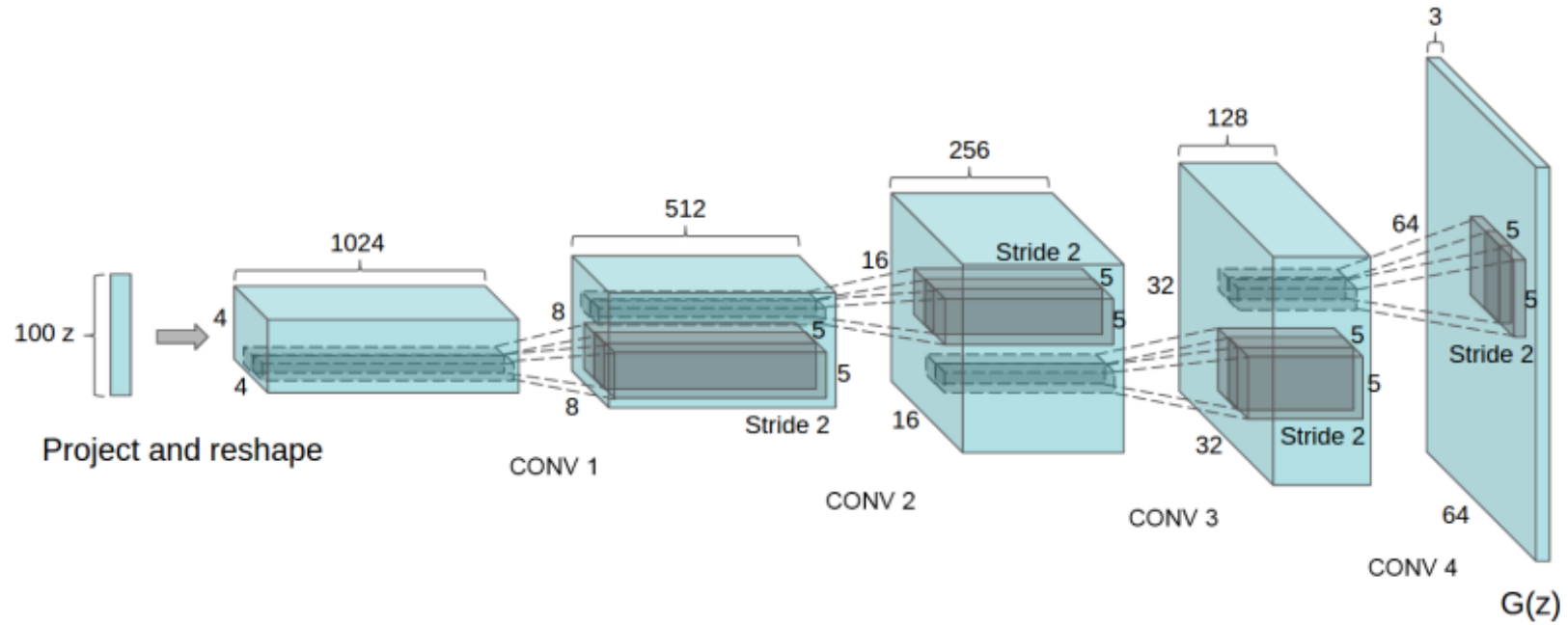
- Sample minibatch of m noise samples $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ from noise prior $p_g(\mathbf{z})$.
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^m \log(1 - D(G(\mathbf{z}^{(i)}))).$$

end for

The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.

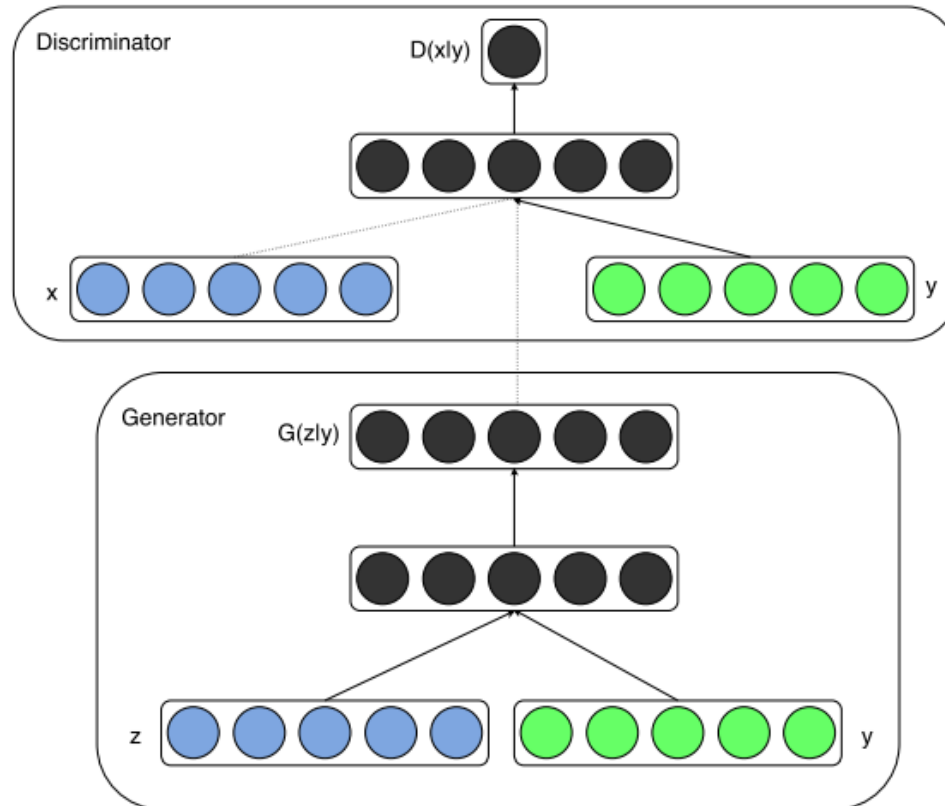
Deep Convolutional GAN



Conditional GAN

- Form

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



Generative Adversarial Nets

Image-to-Image Translation with Conditional Adversarial Networks

- Form

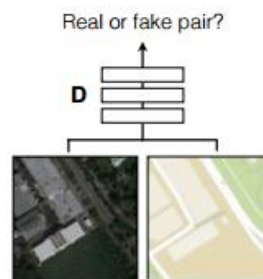
$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x, y \sim p_{data}(x, y)} [\log D(x, y)] + \mathbb{E}_{x \sim p_{data}(x), z \sim p_z(z)} [\log(1 - D(x, G(x, z)))]$$

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

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

- Structure : U-Net(G) + PatchGan(D)

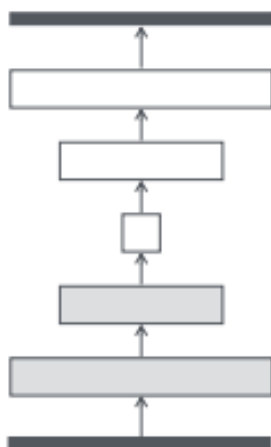
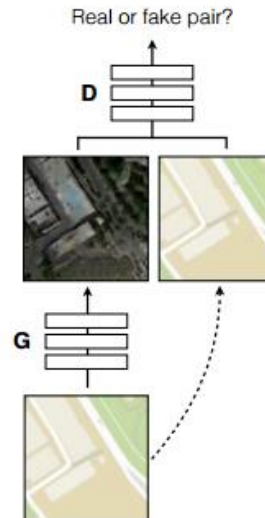
Positive examples



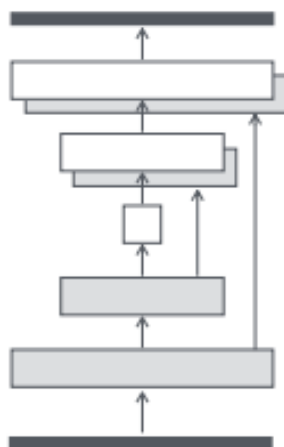
G tries to synthesize fake images that fool **D**

D tries to identify the fakes

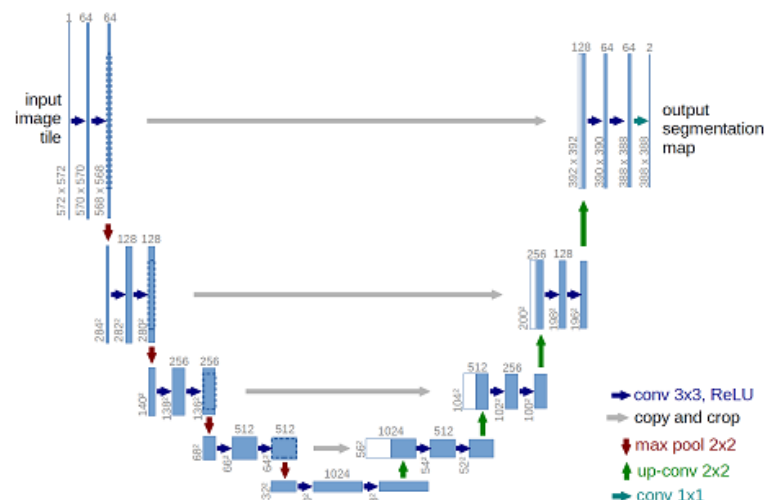
Negative examples



Encoder-decoder



U-Net



Generative Adversarial Nets

Image-to-Image Translation with Conditional Adversarial Networks

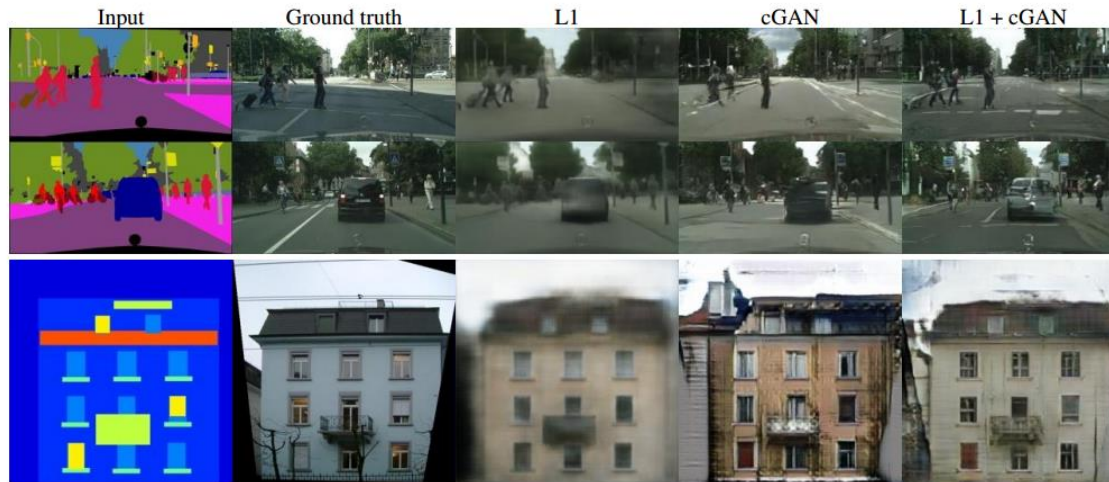
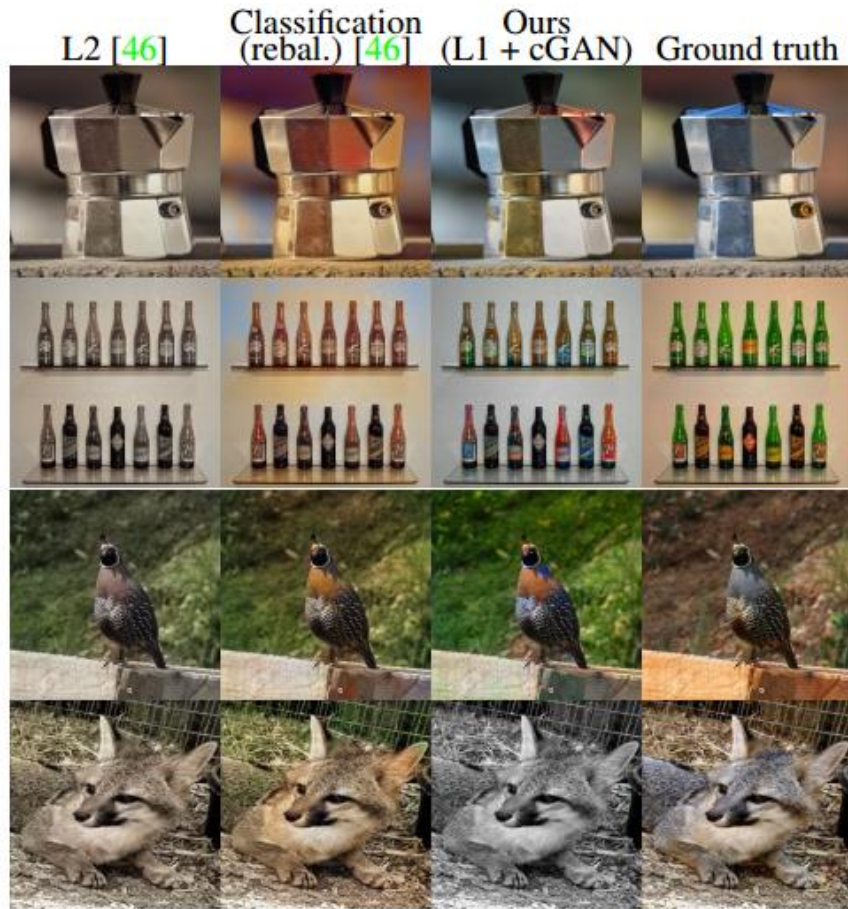


Image-to-Image Translation with Conditional Adversarial Networks



Aerial photo to map



Map to aerial photo



Image-to-Image Translation with Conditional Adversarial Networks

