Image-to-Image Translation with Conditional Adversarial Networks

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros

[GitHub] [Arxiv]

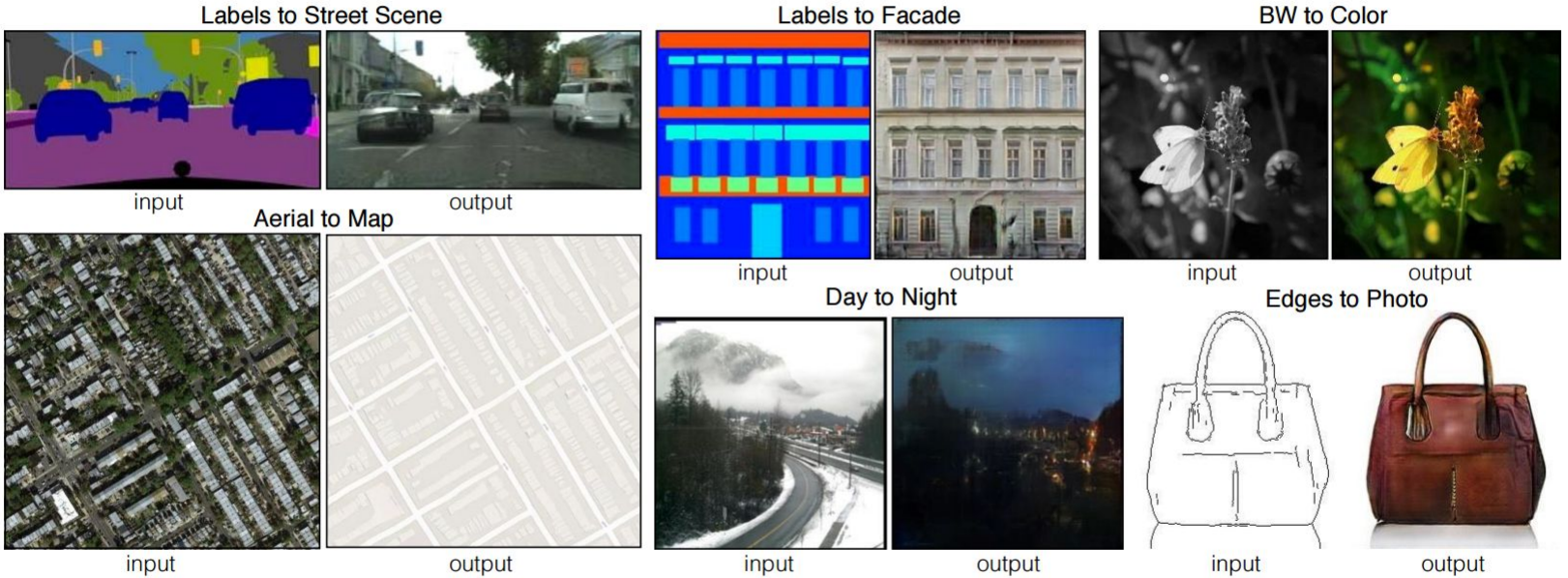
Image Processing Group
Signal Theory and Communications Department
Universitat Politècnica de Catalunya. BARCELONATECH

Slides by Víctor Garcia [GDoc]
UPC Computer Vision Reading Group (25/11/2016)

Index

Introduction

Image → Image



Index

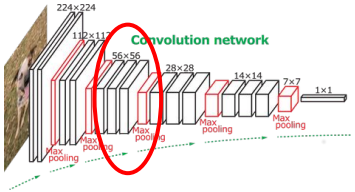
- Introduction
- **State of the Art**
  - Image to Image
- Method
- Experiments
- Conclusions

State of the Art - Image to Image

Super-Resolution

CNN

Loss = MSE($\Phi(I_{in})$, $\Phi(I_{out})$)



State of the Art - Image to Image

Super-Resolution

CNN

Loss = MSE($\Phi(I_{in})$, $\Phi(I_{out})$)

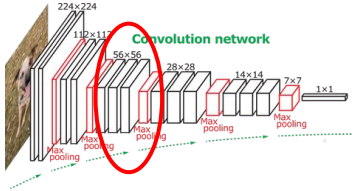
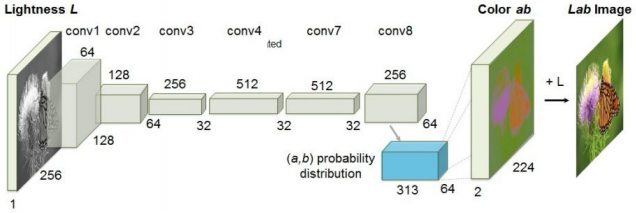


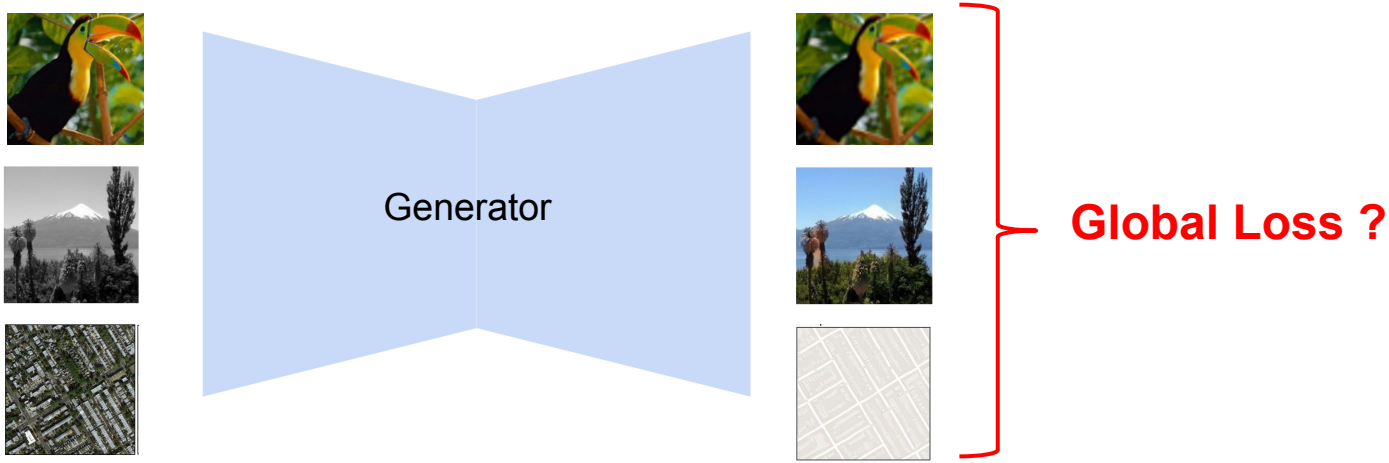
Image colorization

CNN

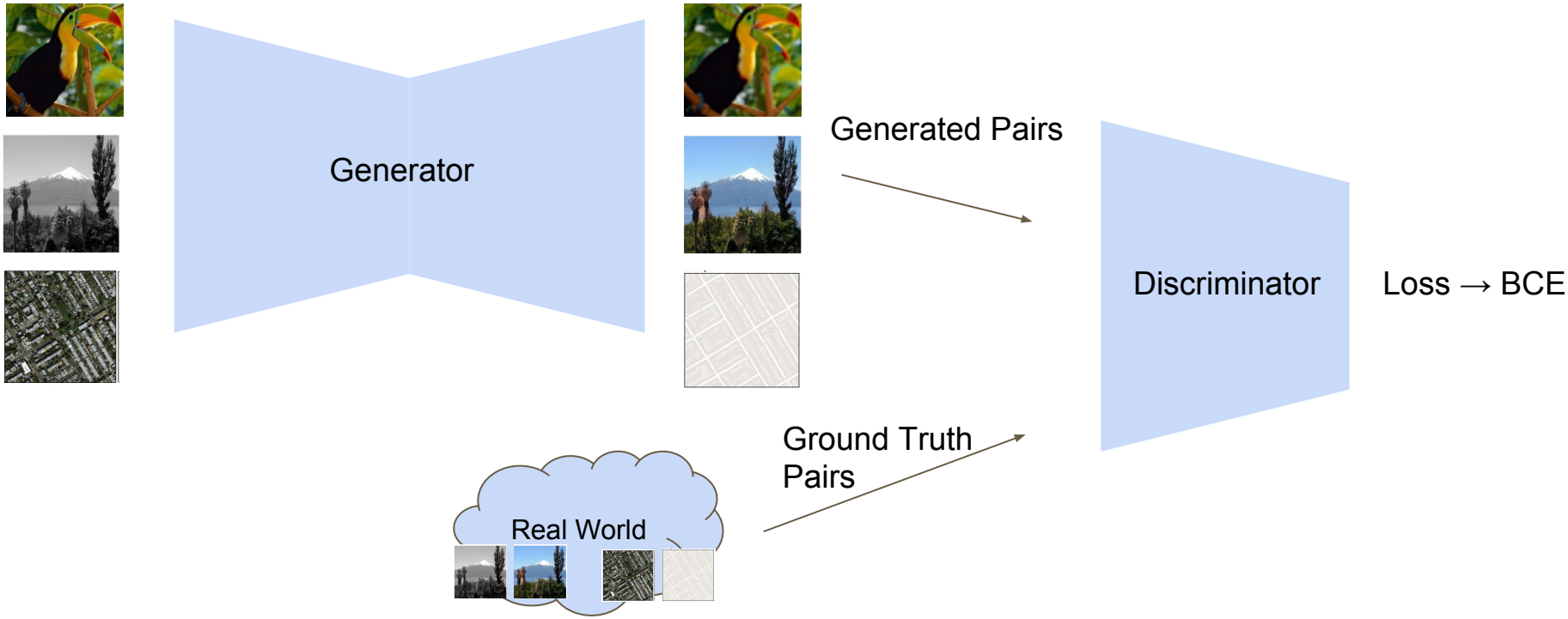
Loss = CE($\Phi(I_{in})$, $\Phi(I_{out})$) weighted



State of the Art - Image to Image



State of the Art - Image to Image



State of the Art - Image to Image

Some works already use conditional GANs for Image to Image translation

Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. (2 Weeks ago)

CNN

Loss1 = MSE_VGG($\Phi(I_{in})$, $\Phi(I_{out})$)

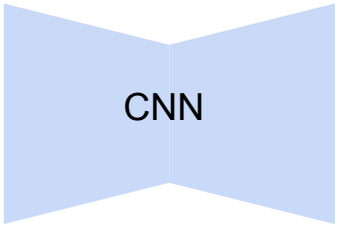
Loss2 = Regularization

Loss3 = GAN Loss

State of the Art - Image to Image

Some works already use conditional GANs for Image to Image translation

Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. (2 Weeks ago)

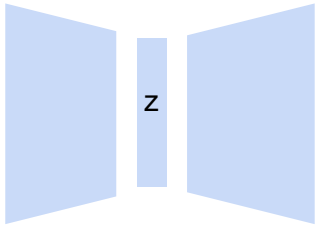


Loss1 = MSE_VGG($\Phi(I_{in})$, $\Phi(I_{out})$)

Loss2 = Regularization

Loss3 = GAN Loss

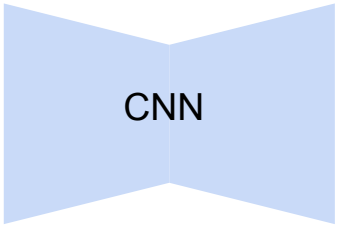
Unsupervised cross-domain Image Generation (2 Weeks ago)



State of the Art - Image to Image

Some works already use conditional GANs for Image to Image translation

Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. (2 Weeks ago)

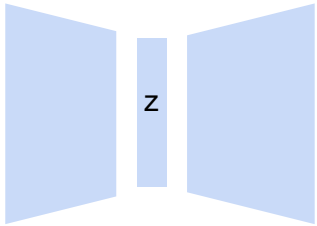


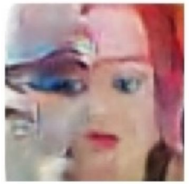
Loss1 = MSE_VGG($\Phi(I_{in})$, $\Phi(I_{out})$)

Loss2 = Regularization

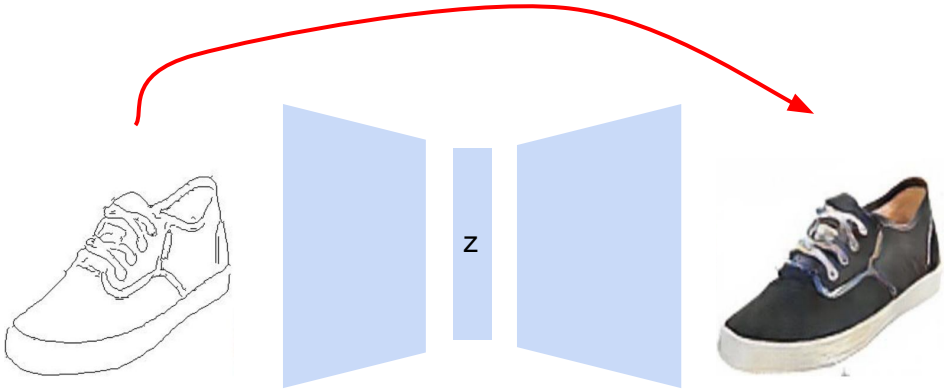
Loss3 = GAN Loss

Unsupervised cross-domain Image Generation (2 Weeks ago)





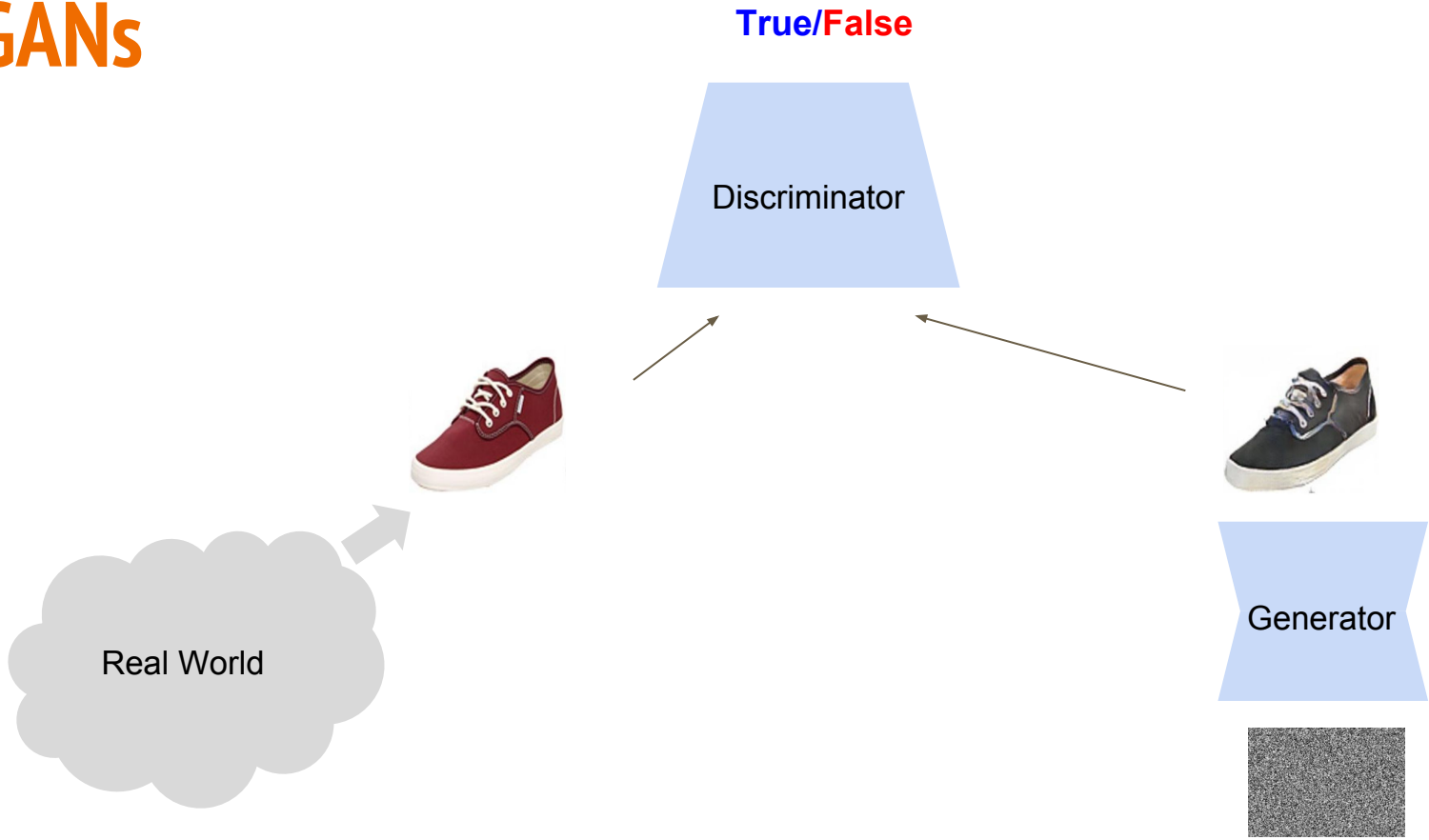
State of the Art - Image to Image



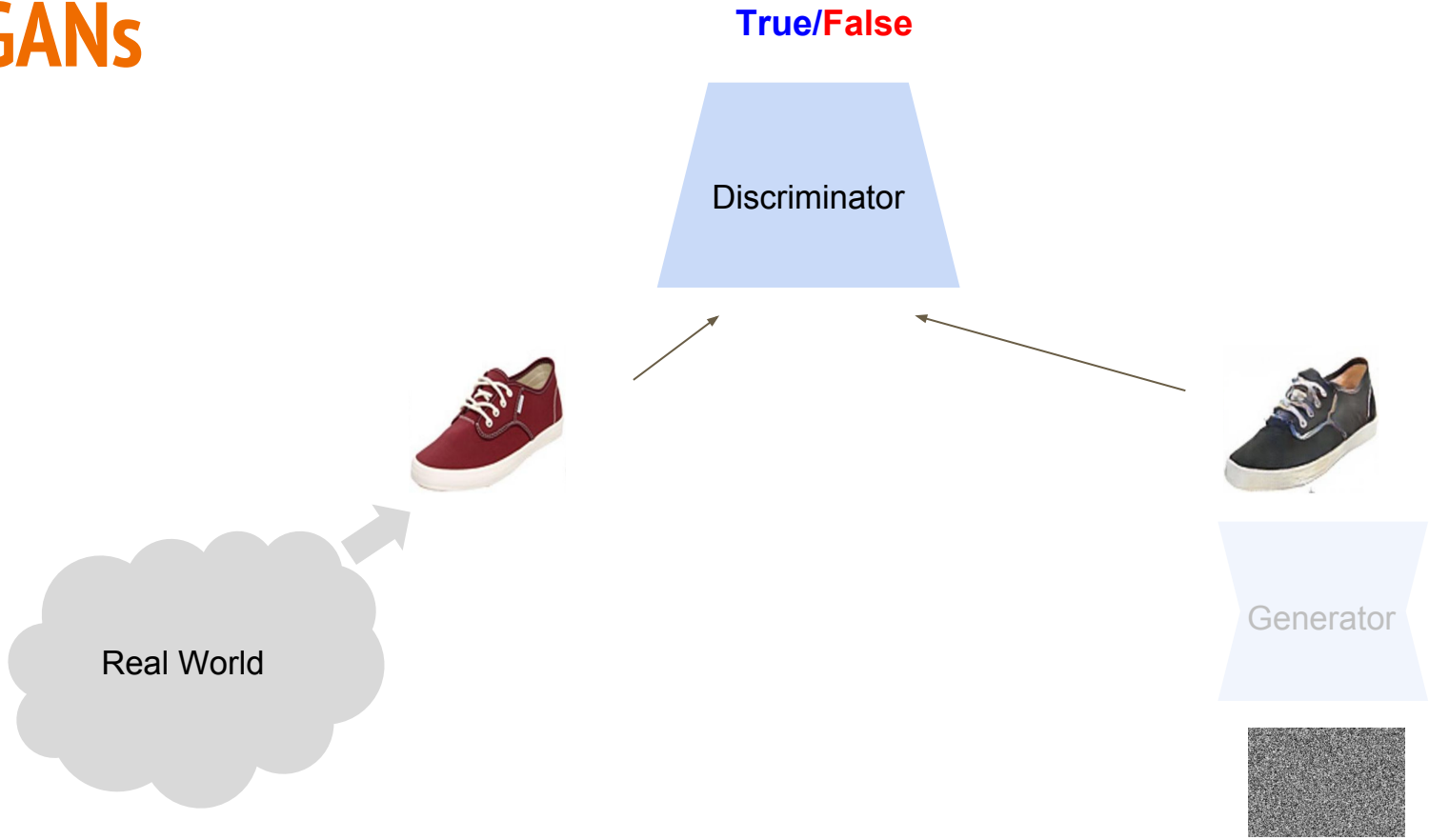
This paper solves the problem of Image to Image Translation using the same architecture for any task without need of handcrafting any Loss function.

Index

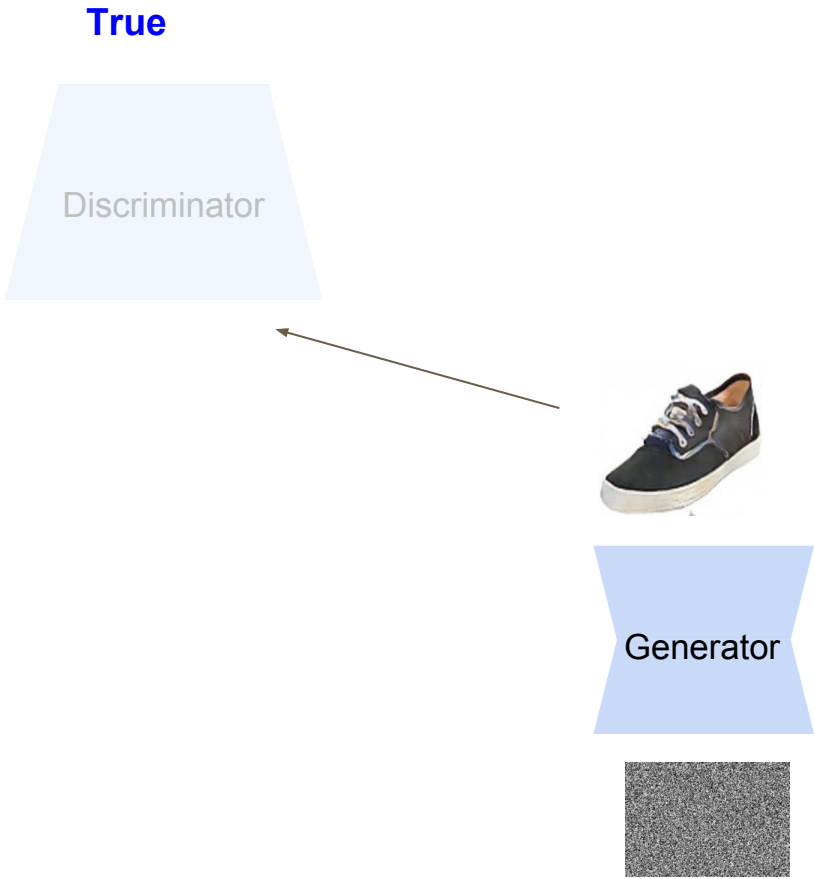
GANs



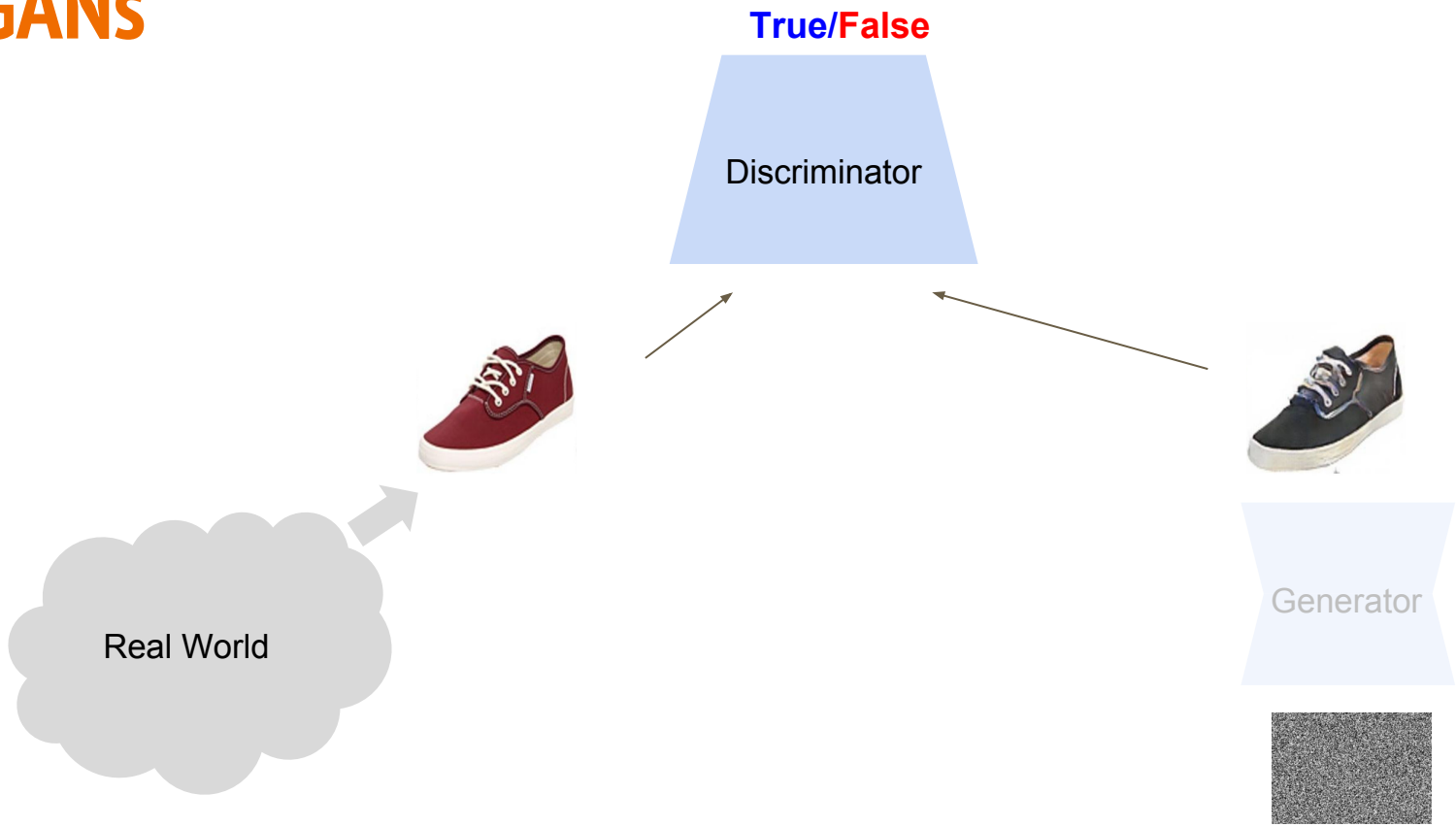
GANs



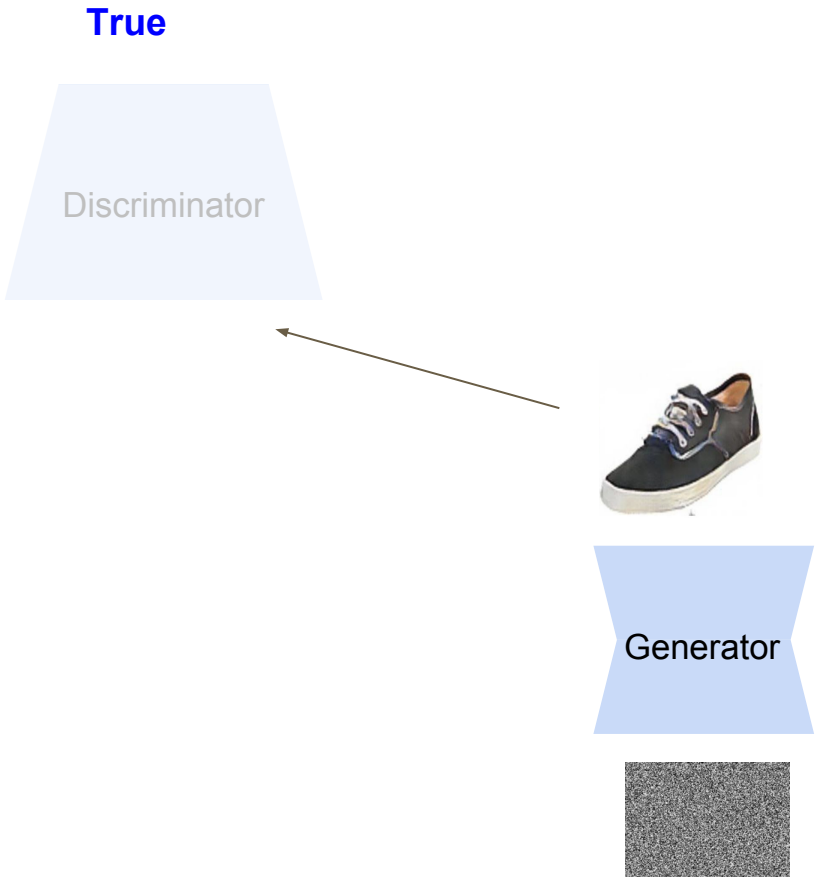
GANs



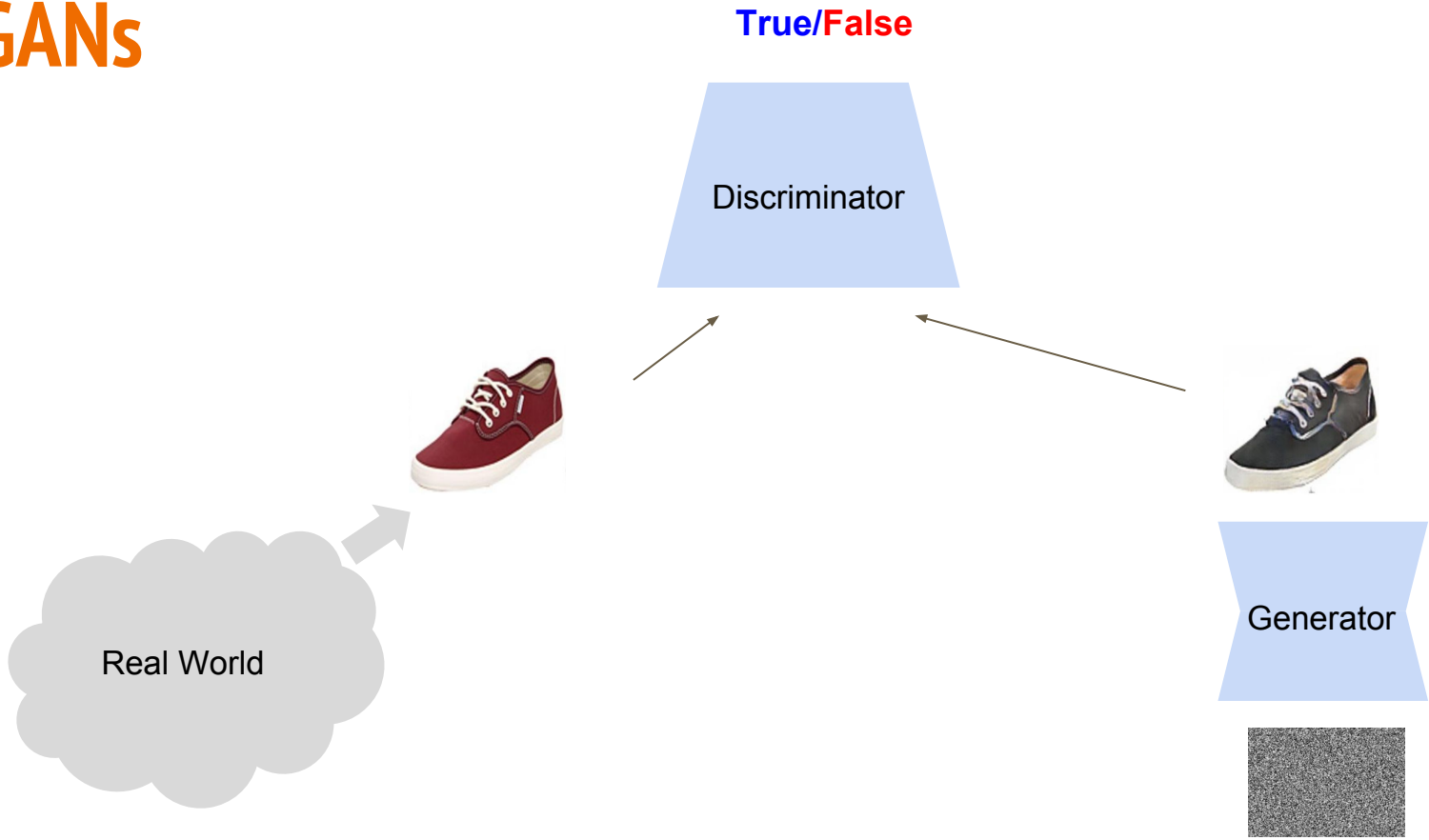
GANs



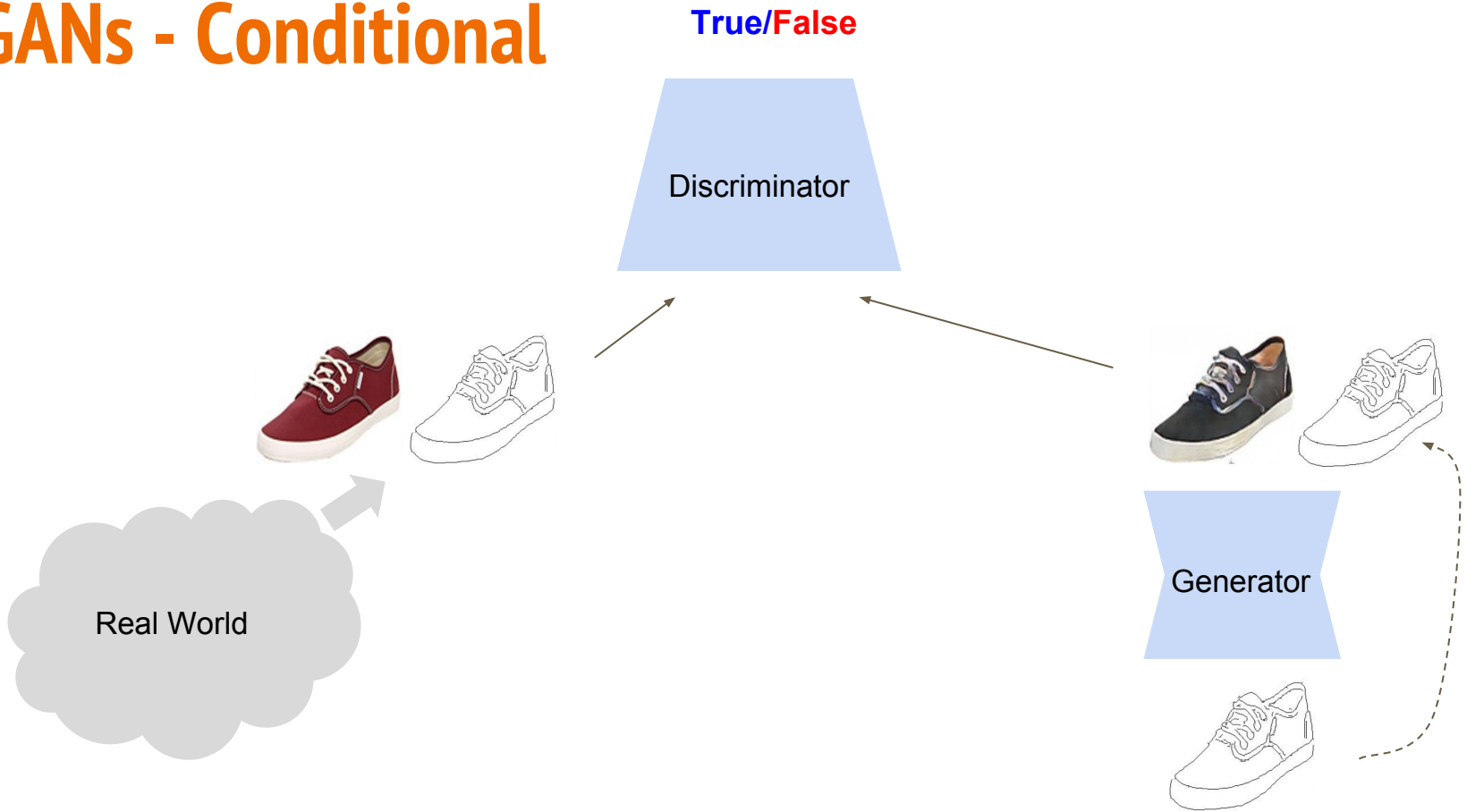
GANs



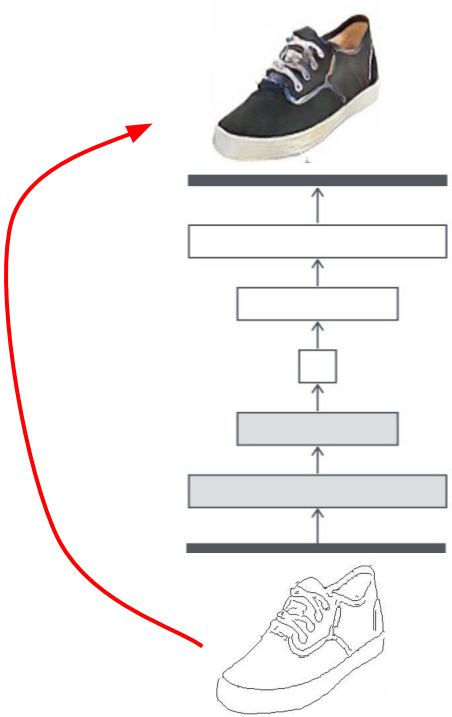
GANs



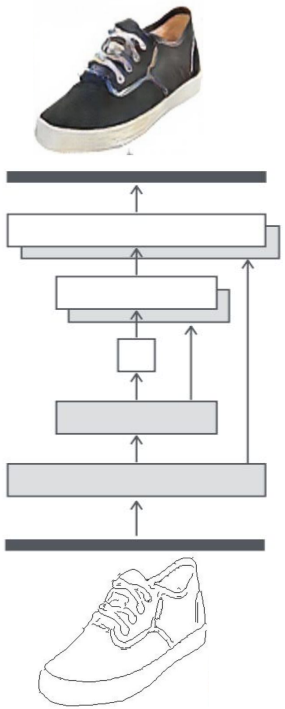
GANs - Conditional



Generator - Unet



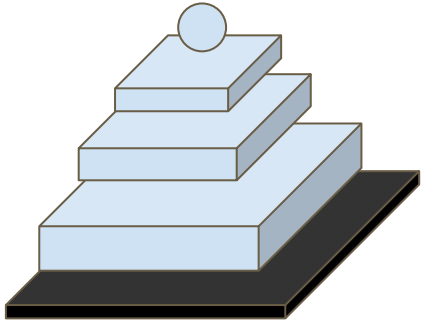
Generator - Unet



Skip Connections

Discriminator - Patch GAN

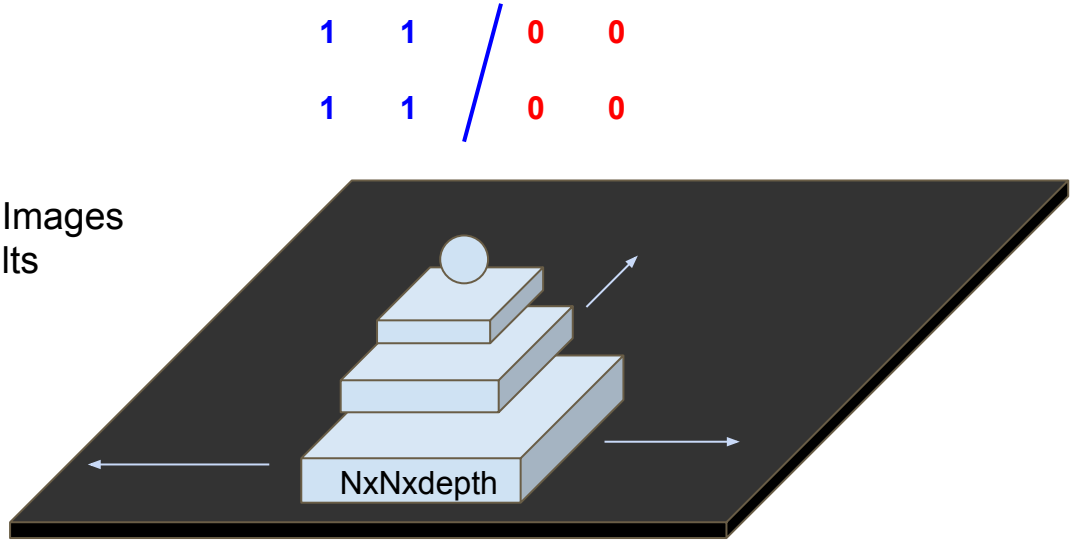
1/0



256x256x3

Discriminator - Patch GAN

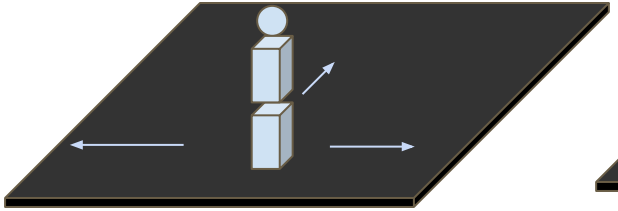
- Faster
- Training with larger Images
- Equal or better results



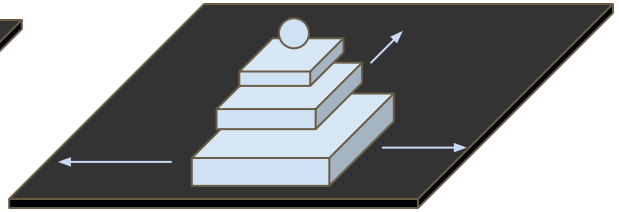
512x512x3

Discriminator - Patch GAN

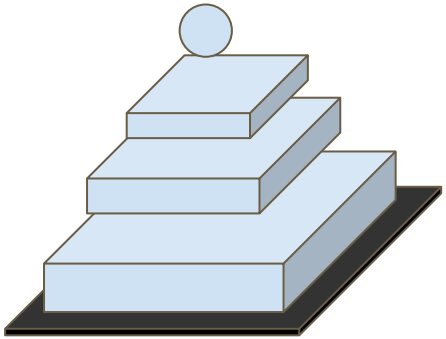
PixelGAN



PatchGAN



ImageGAN



Optimization Losses

For training they are only using two Losses:

- GAN Loss:

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

Optimization Losses

For training they are only using two Losses:

- GAN Loss:

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

- L1 Loss (Enforce correctness at Low Frequencies):

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

Optimization Losses

For training they are only using two Losses:

- GAN Loss:

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

- L1 Loss (Enforce correctness at Low Frequencies):

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

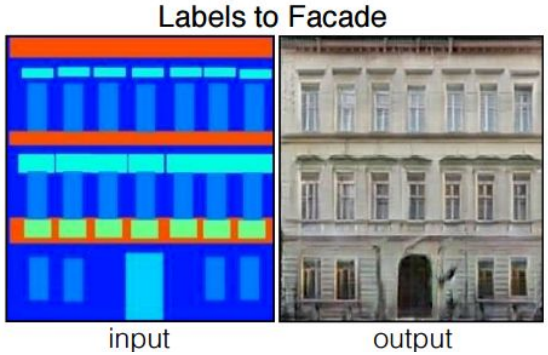
$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda \mathcal{L}_{L1}(G)$$

Index

- Introduction
- State of the Art
- Method
- **Experiments**
  - Experiment types
  - Evaluation Metrics
  - Cityscapes
  - Colorization
  - Map <-> Aerial
- Conclusions

Experiments

- ***Archirectural labels → photo,* trained on Facades**
- *Semantic labels <-> photo*, on Cityscapes
- *Map <-> Aerial photo,* from Google Maps
- *BW → Color photos,* trained on Imagenet
- *Edges → Photo,* trained on Handbags and Shoes
- *Sketch → Photo,* human drawn sketches
- *Day → Night*

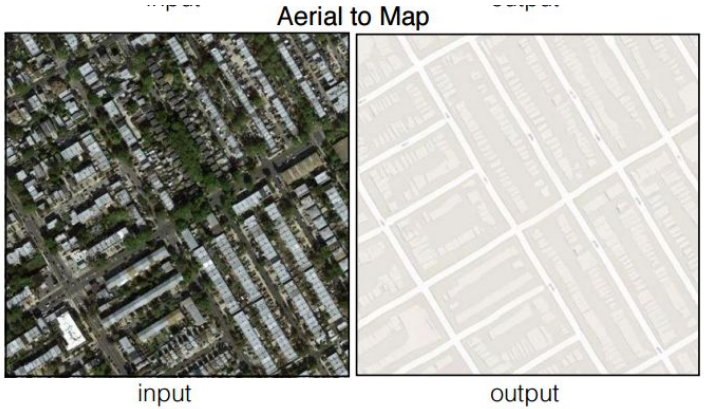


Experiments

- *Archirectural labels → photo,* trained on Facades
- ***Semantic labels <-> photo*, on Cityscapes**
- *Map <-> Aerial photo,* from Google Maps
- *BW → Color photos,* trained on Imagenet
- *Edges → Photo,* trained on Handbags and Shoes
- *Sketch → Photo,* human drawn sketches
- *Day → Night*

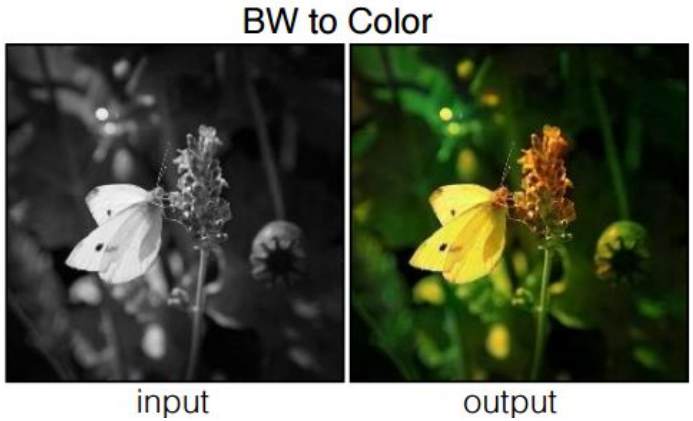
Experiments

- *Archirectural labels → photo,* trained on Facades
- *Semantic labels <-> photo*, on Cityscapes
- ***Map <-> Aerial photo,* from Google Maps**
- *BW → Color photos,* trained on Imagenet
- *Edges → Photo,* trained on Handbags and Shoes
- *Sketch → Photo,* human drawn sketches
- *Day → Night*



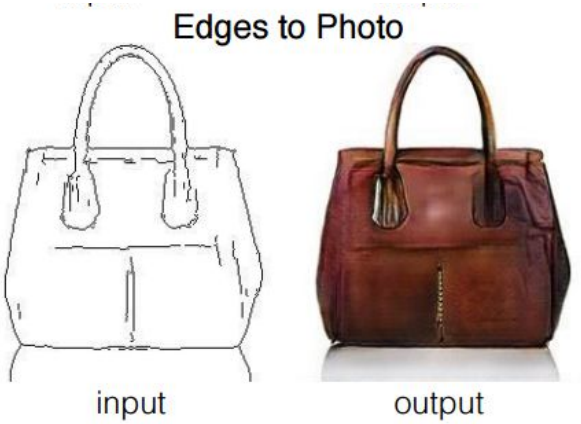
Experiments

- *Archirectural labels → photo,* trained on Facades
- *Semantic labels <-> photo*, on Cityscapes
- *Map <-> Aerial photo,* from Google Maps
- ***BW → Color photos,* trained on Imagenet**
- *Edges → Photo,* trained on Handbags and Shoes
- *Sketch → Photo,* human drawn sketches
- *Day → Night*



Experiments

- *Archirectural labels → photo,* trained on Facades
- *Semantic labels <-> photo*, on Cityscapes
- *Map <-> Aerial photo,* from Google Maps
- *BW → Color photos,* trained on Imagenet
- ***Edges → Photo,* trained on Handbags and Shoes**
- *Sketch → Photo,* human drawn sketches
- *Day → Night*

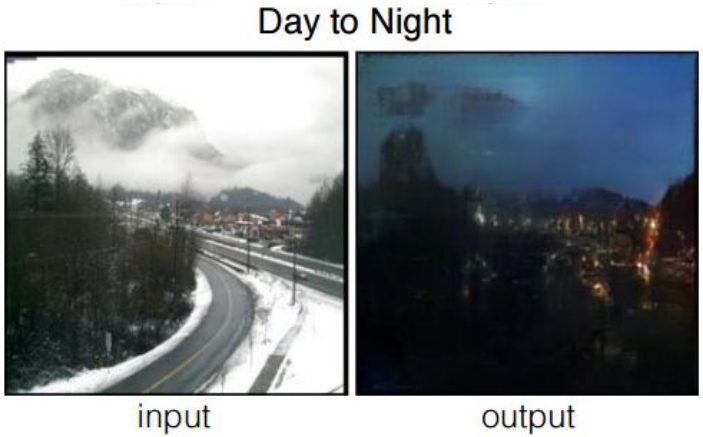


Experiments

- *Archirectural labels → photo,* trained on Facades
- *Semantic labels <-> photo*, on Cityscapes
- *Map <-> Aerial photo,* from Google Maps
- *BW → Color photos,* trained on Imagenet
- *Edges → Photo,* trained on Handbags and Shoes
- ***Sketch → Photo,* human drawn sketches**
- *Day → Night*

Experiments

- *Archirectural labels → photo,* trained on Facades
- *Semantic labels <-> photo*, on Cityscapes
- *Map <-> Aerial photo,* from Google Maps
- *BW → Color photos,* trained on Imagenet
- *Edges → Photo,* trained on Handbags and Shoes
- *Sketch → Photo,* human drawn sketches
- ***Day → Night***



Experiments

- *Archirectural labels → photo,* trained on Facades
- ***Semantic labels <-> photo*, on Cityscapes**
- ***Map <-> Aerial photo,* from Google Maps**
- ***BW → Color photos,* trained on Imagenet**
- *Edges → Photo,* trained on Handbags and Shoes
- *Sketch → Photo,* human drawn sketches
- *Day → Night*

Evaluation Metrics - Cityscapes

Evaluation for qualitative images is an open and difficult problem

For semantic labels <-> Photo in Cityscapes we are using:

FCN-Score

Evaluation Metrics - Colorization and Maps

Amazon Mekanical Turks

Is this picture Real ? Yes/No

Cityscapes - FCN Score

- FCN-score

Loss	Per-pixel acc.	Per-class acc.	Class IOU
L1	0.44	0.14	0.10
GAN	0.22	0.05	0.01
cGAN	0.61	**0.21**	**0.16**
L1+GAN	**0.64**	0.19	0.15
L1+cGAN	0.63	**0.21**	**0.16**
Ground truth	0.80	0.26	0.21

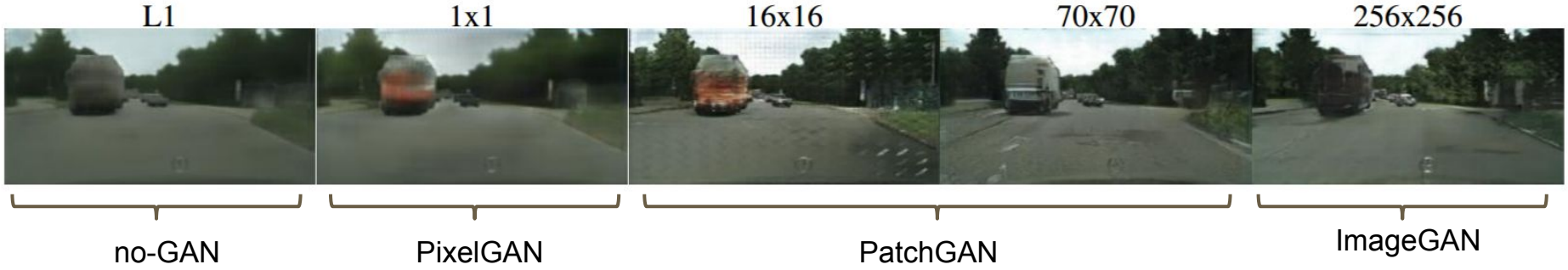
Cityscapes - FCN Score

- FCN-score

Loss	Per-pixel acc.	Per-class acc.	Class IOU
L1	0.44	0.14	0.10
GAN	0.22	0.05	0.01
cGAN	0.61	**0.21**	**0.16**
L1+GAN	**0.64**	0.19	0.15
L1+cGAN	0.63	**0.21**	**0.16**
Ground truth	0.80	0.26	0.21

Cityscapes - PatchGAN

Discriminator receptive field	Per-pixel acc.	Per-class acc.	Class IOU
1×1	0.44	0.14	0.10
16×16	0.62	0.20	**0.16**
70×70	**0.63**	**0.21**	**0.16**
256×256	0.47	0.18	0.13

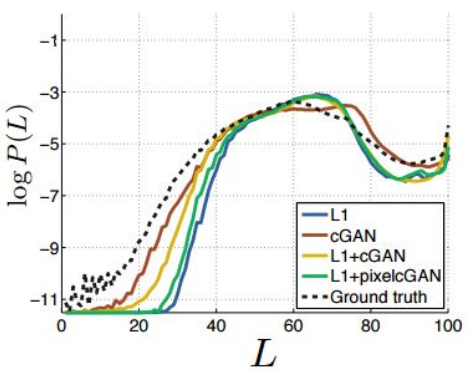


Cityscapes - Color Distribution

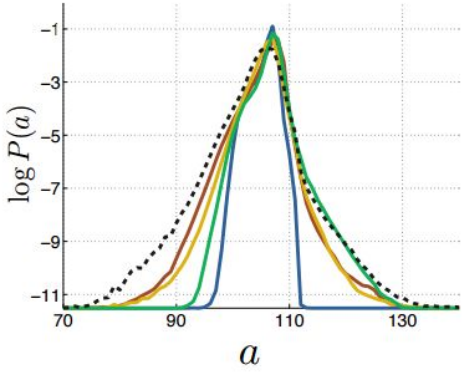
L1

L1 + pixelcGAN

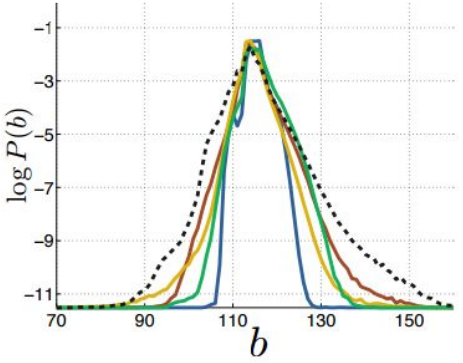
L1 + cGAN



(a)

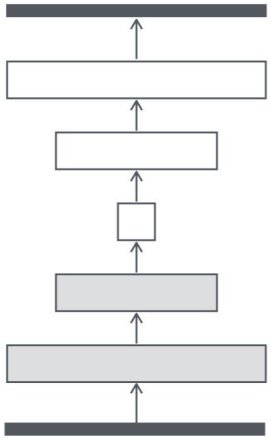


(b)

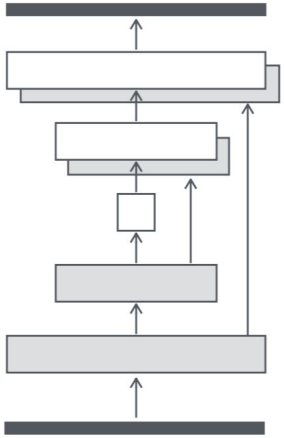


(c)

Cityscapes - Autoencoder vs U-net



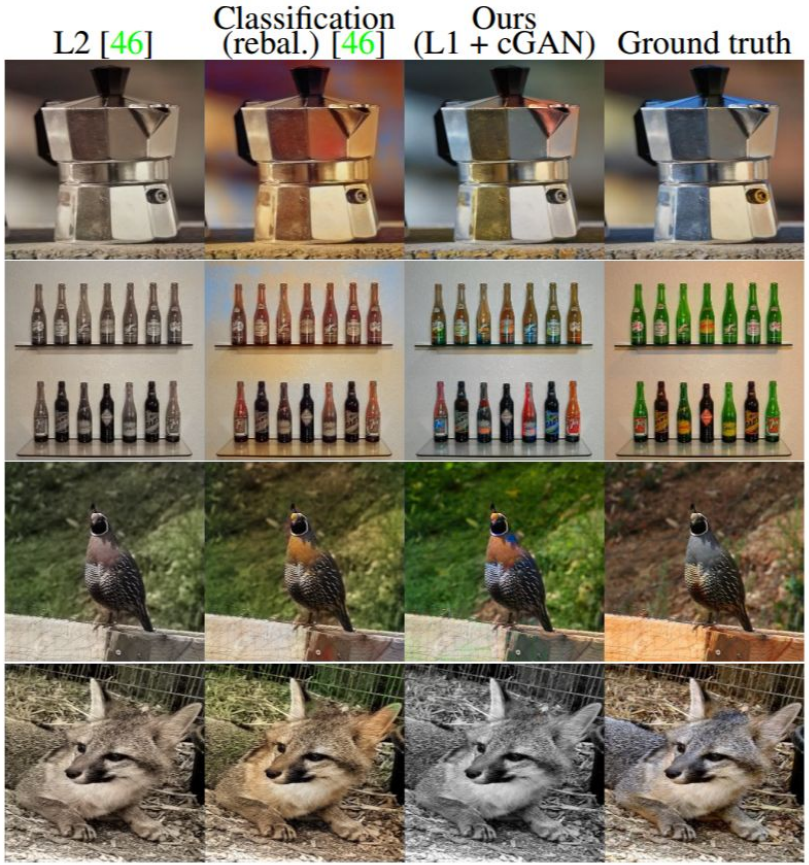
Encoder-decoder



U-Net

Image Colorization

	L2	Classification (rebal.)	L1+cGAN
Labeled as real	16.3%	**27.8%**	22.5%



Map to Aerial

	L1	L1+cGAN
Labeled as real	0.8%	**18.9%**

Map to Aerial

512x512

Aerial to Map

	L1	L1+cGAN
Labeled as real	2.8%	**6.1%**

Aerial to Map

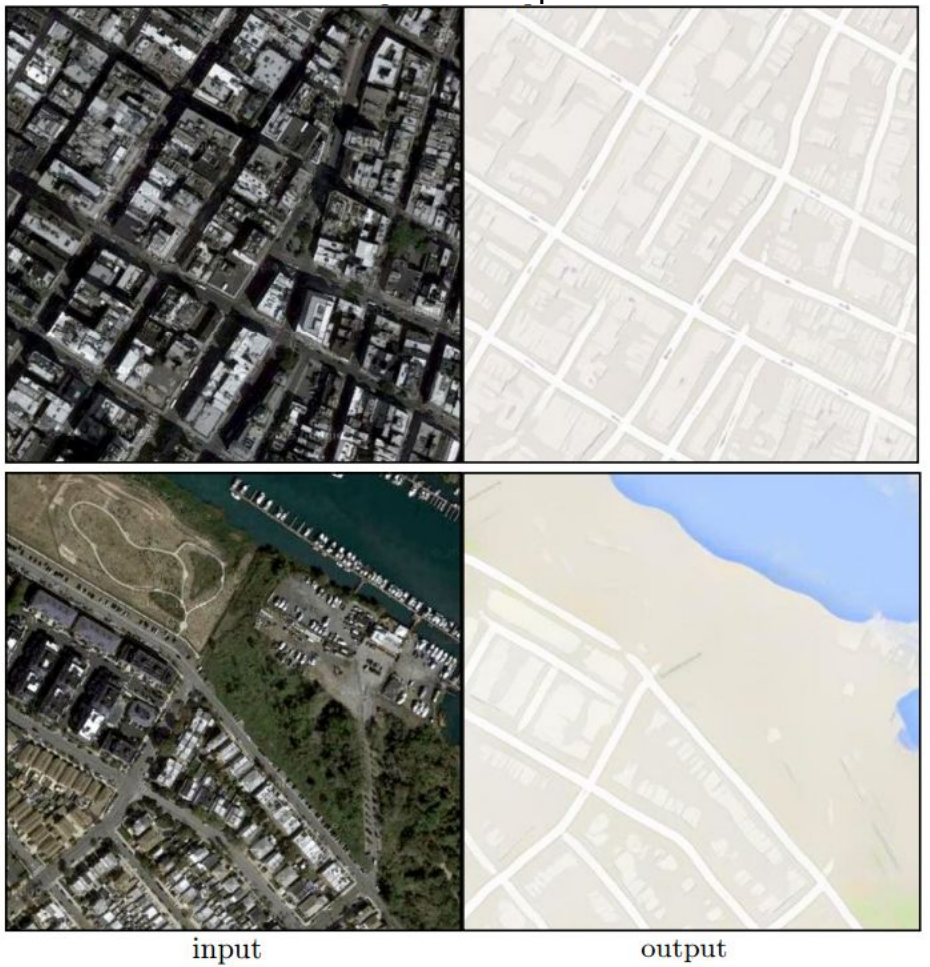
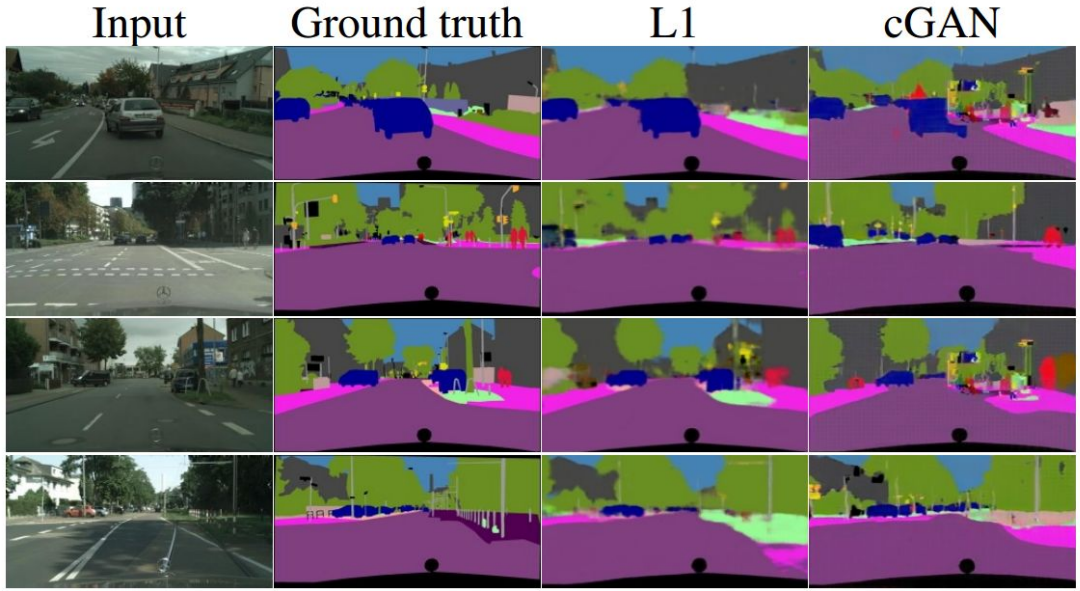
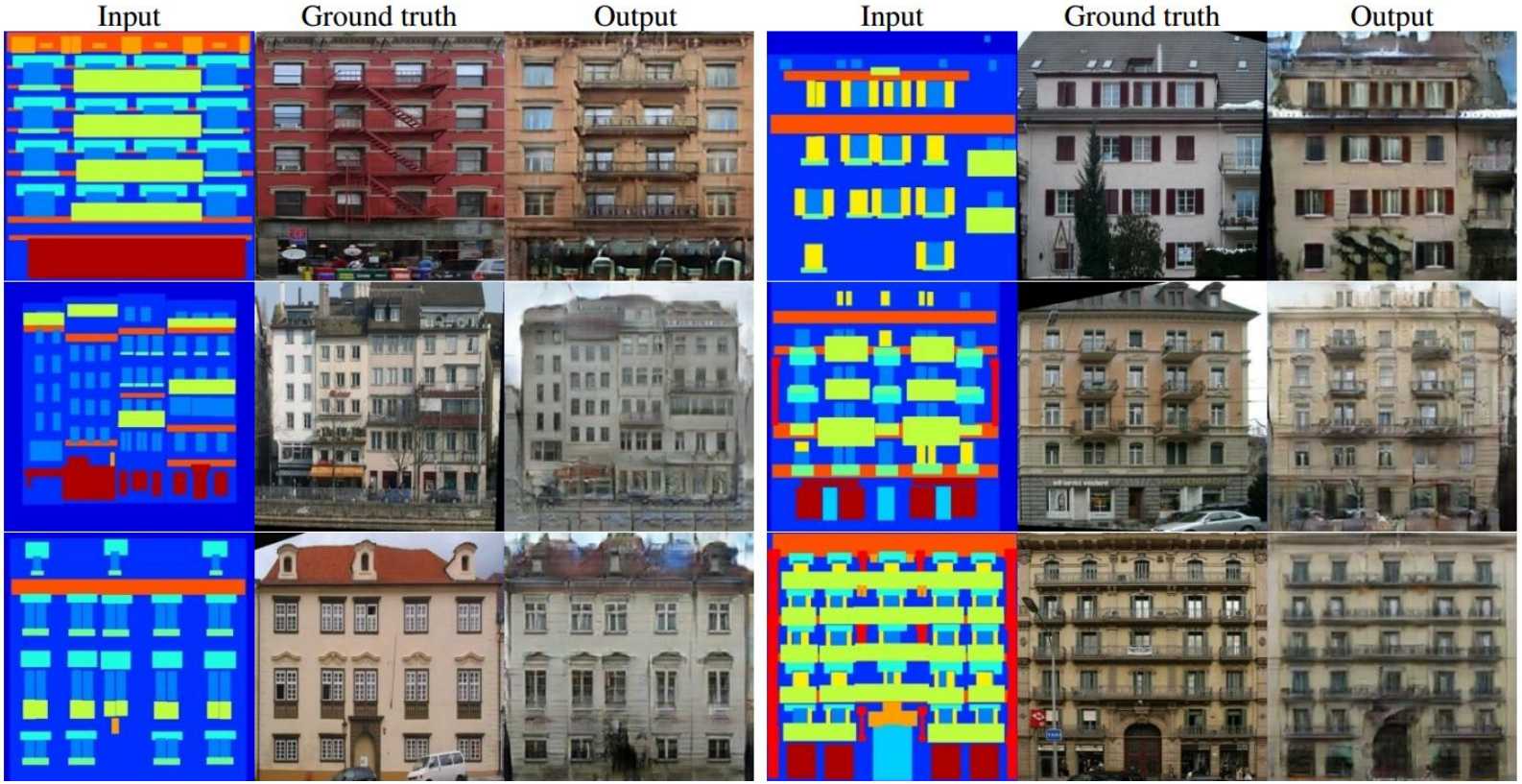


Image Segmentation

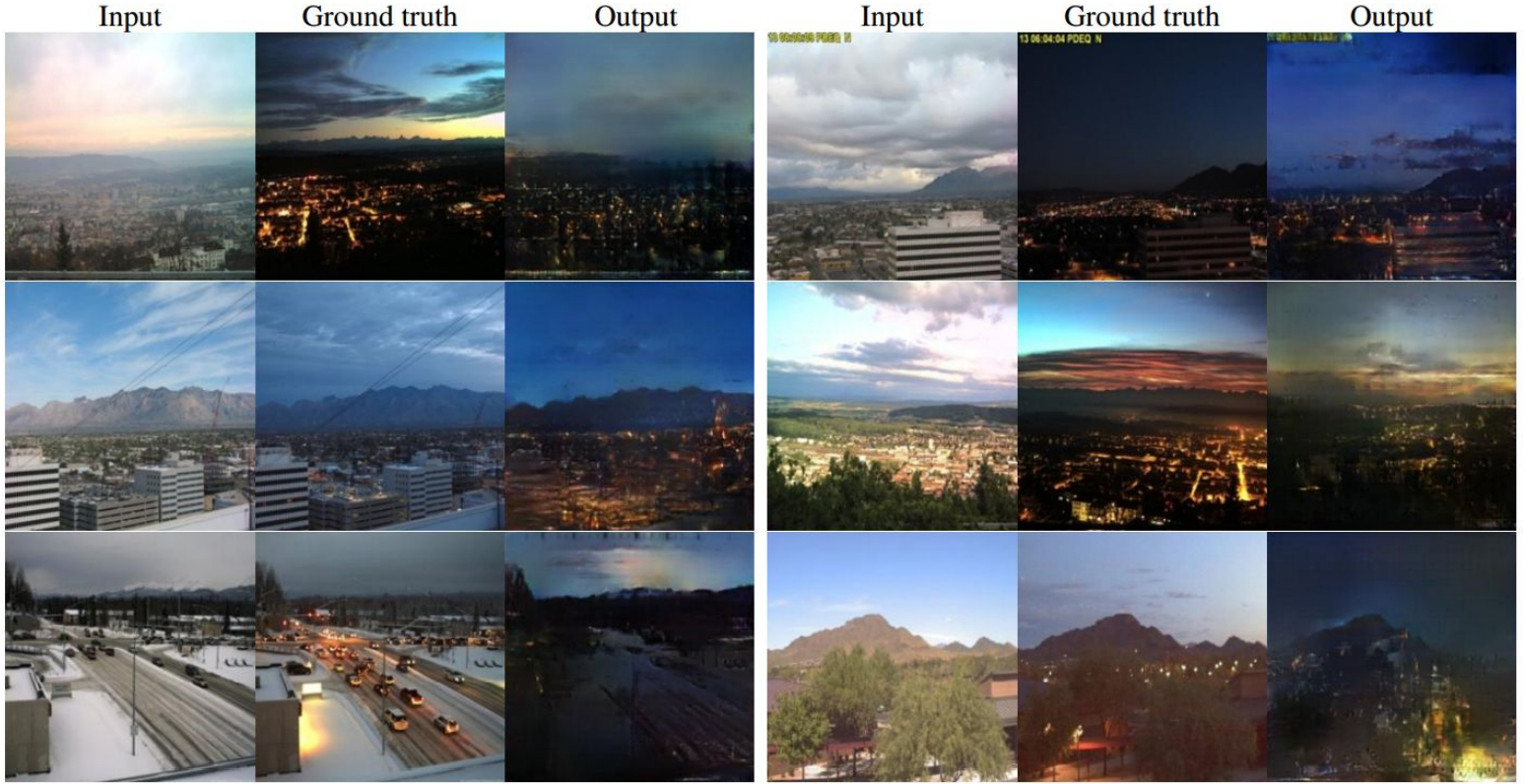
	L1	cGAN	L1+cGAN
Per-pixel acc.	**0.86**	0.74	0.83
Per-class acc.	**0.42**	0.28	0.36
Class IOU	**0.35**	0.22	0.29



Other Experiments - Labels → Facades



Other Experiments - Day → Night



Other Experiments - Edges → Handbags

Other Experiments - Edges → Shoes

Other Experiments - Edges → Shoes

Conclusions

- Conditional Adversarial Networks are a promising approach for many image to image translation tasks.
- Using U-net as a generator has been a big improvement for forwarding low level features through the network and partially reconstructing it at the output.
- Using the Patch GAN Approach we can train and generate high resolution images


THANK YOU