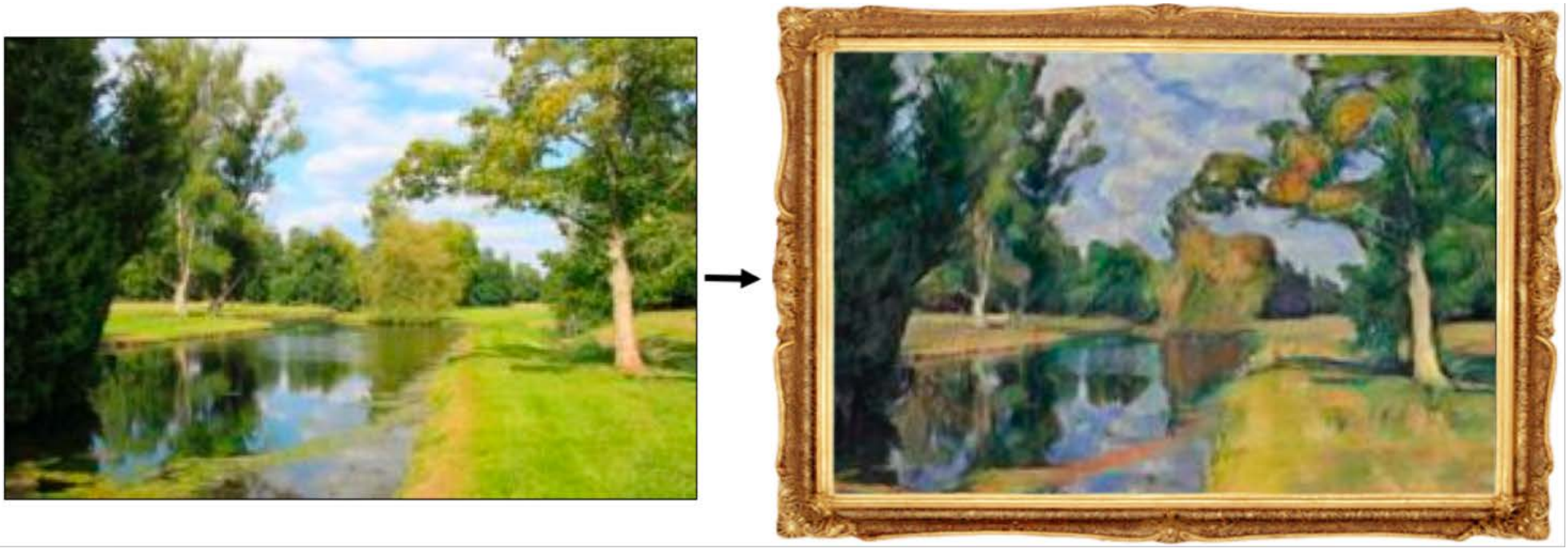
Conditional GANs



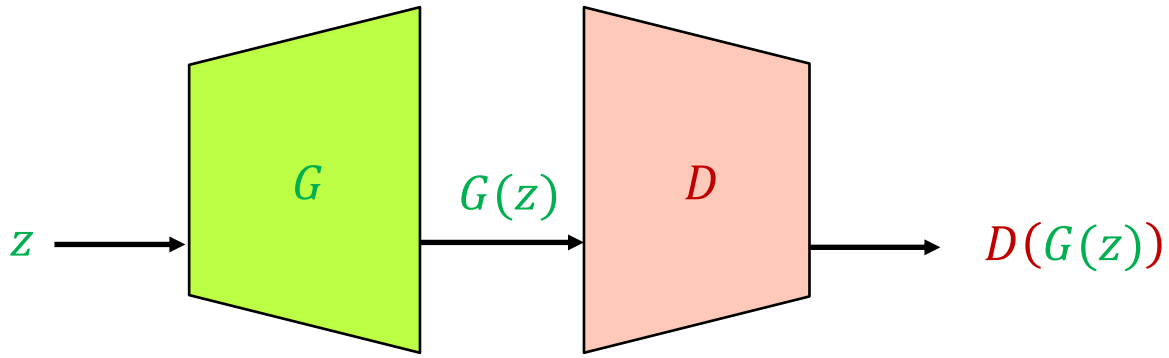
Source

Outline

- Introduction
- Generation conditioned on class
  - Self-attention GAN
  - BigGAN
- Generation conditioned on image
  - Paired image-to-image translation: pix2pix
  - Unpaired image-to-image translation: CycleGAN
- Recent trends

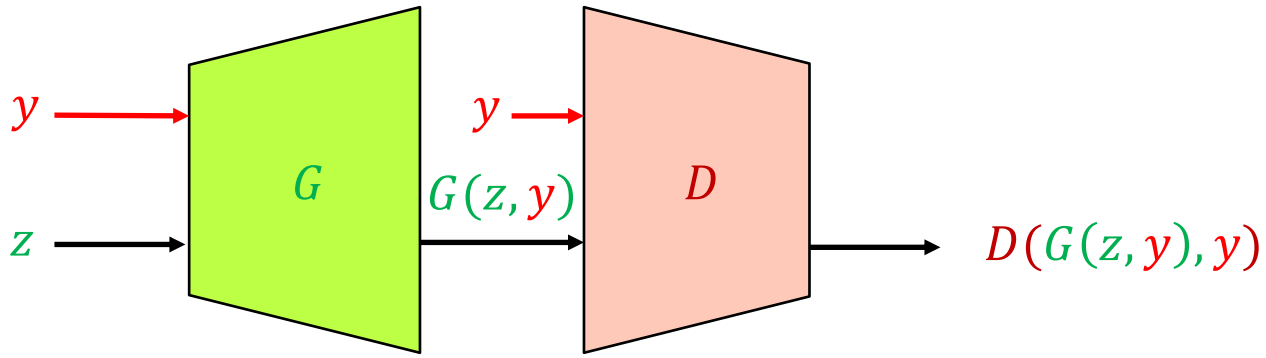
Conditional generation

- Suppose we want to condition the generation of samples on discrete side information (label) y
  - How do we add y to the basic GAN framework?



Conditional generation

- Suppose we want to condition the generation of samples on discrete side information (label) y
  - How do we add y to the basic GAN framework?



Conditional generation

- Example: simple network for generating 28 x 28 MNIST digits

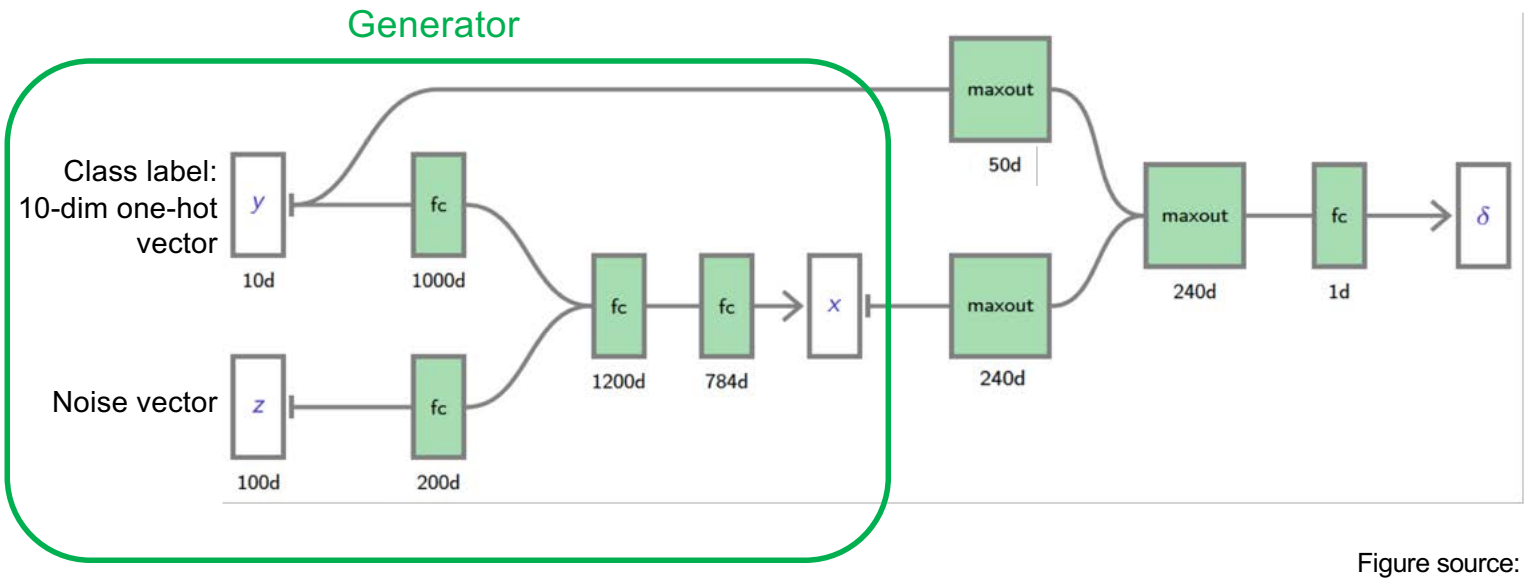


Figure source:
F. Fleuret

M. Mirza and S. Osindero, Conditional Generative Adversarial Nets, arXiv 2014

Conditional generation

- Example: simple network for generating 28 x 28 MNIST digits

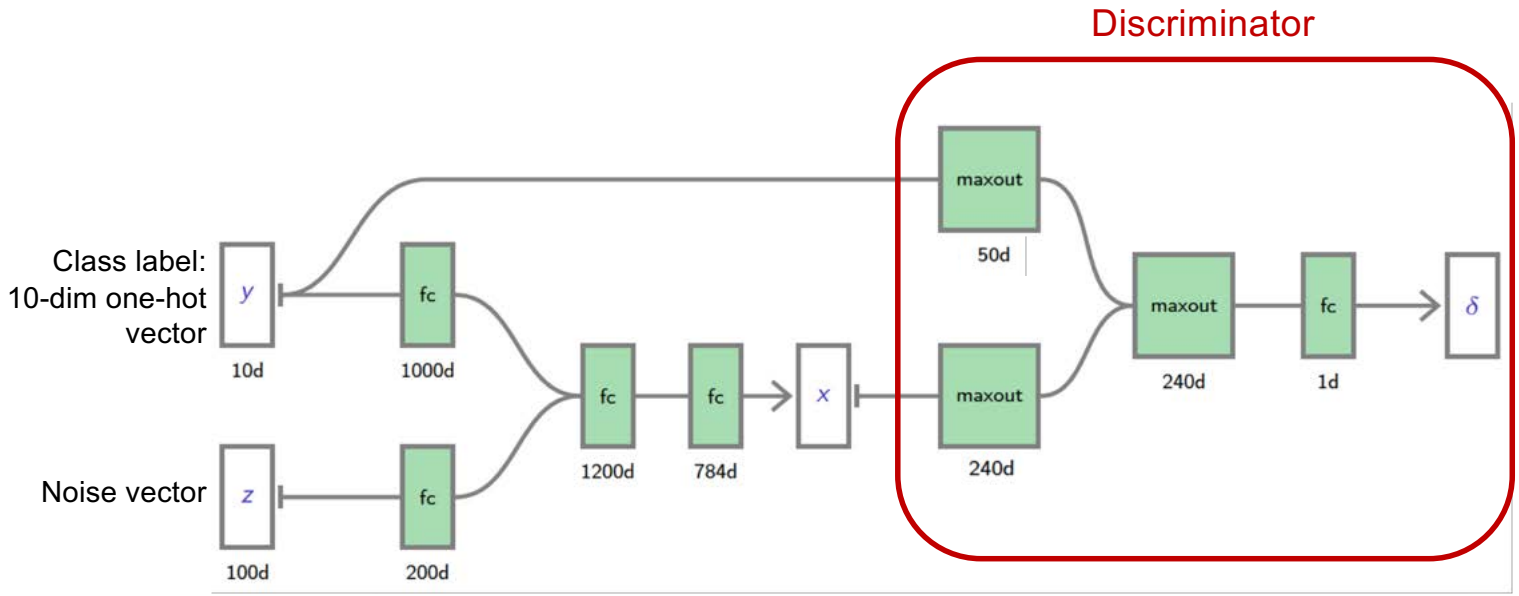


Figure source:
F. Fleuret

M. Mirza and S. Osindero, Conditional Generative Adversarial Nets, arXiv 2014

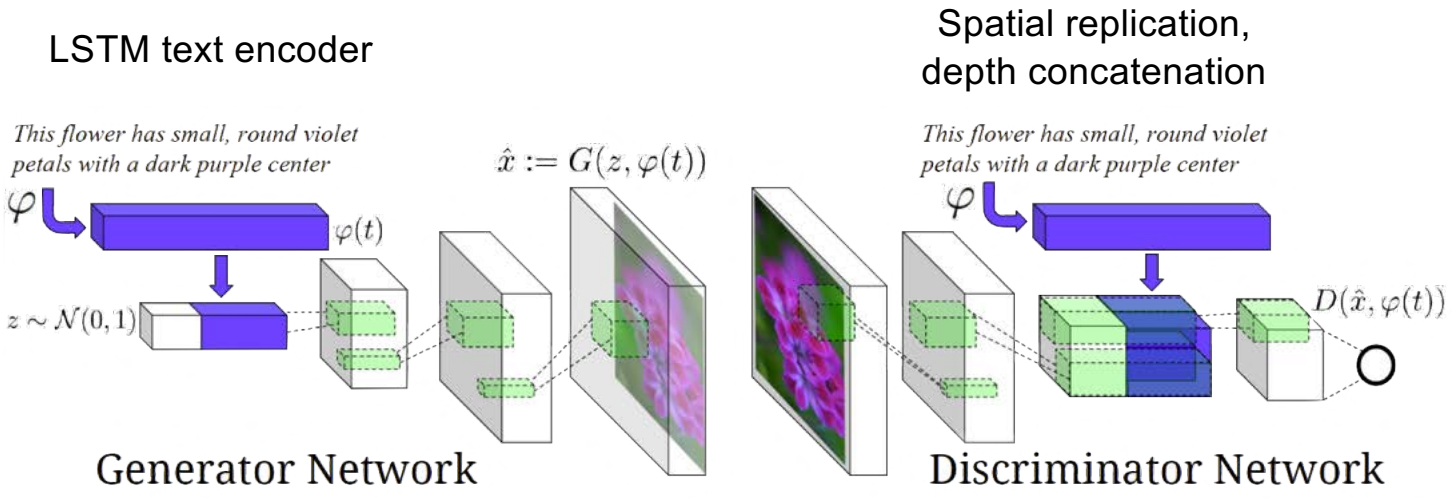
Conditional generation

- Example: simple network for generating 28 x 28 MNIST digits

M. Mirza and S. Osindero, Conditional Generative Adversarial Nets, arXiv 2014

Conditional generation

- Another example: text-to-image synthesis

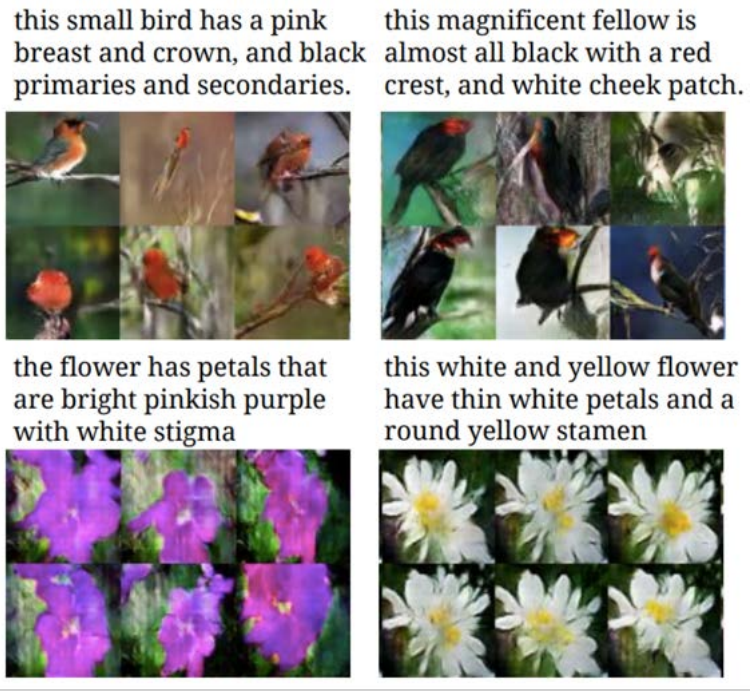


S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, H. Lee, Generative adversarial text to image synthesis, ICML 2016

Conditional generation

- Another example: text-to-image synthesis

Previously unseen captions (*zero-shot* setting)



Captions seen in the training set

S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, H. Lee, Generative adversarial text to image synthesis, ICML 2016

Outline

- Introduction
- Generation conditioned on class
  - Self-attention GAN
  - BigGAN

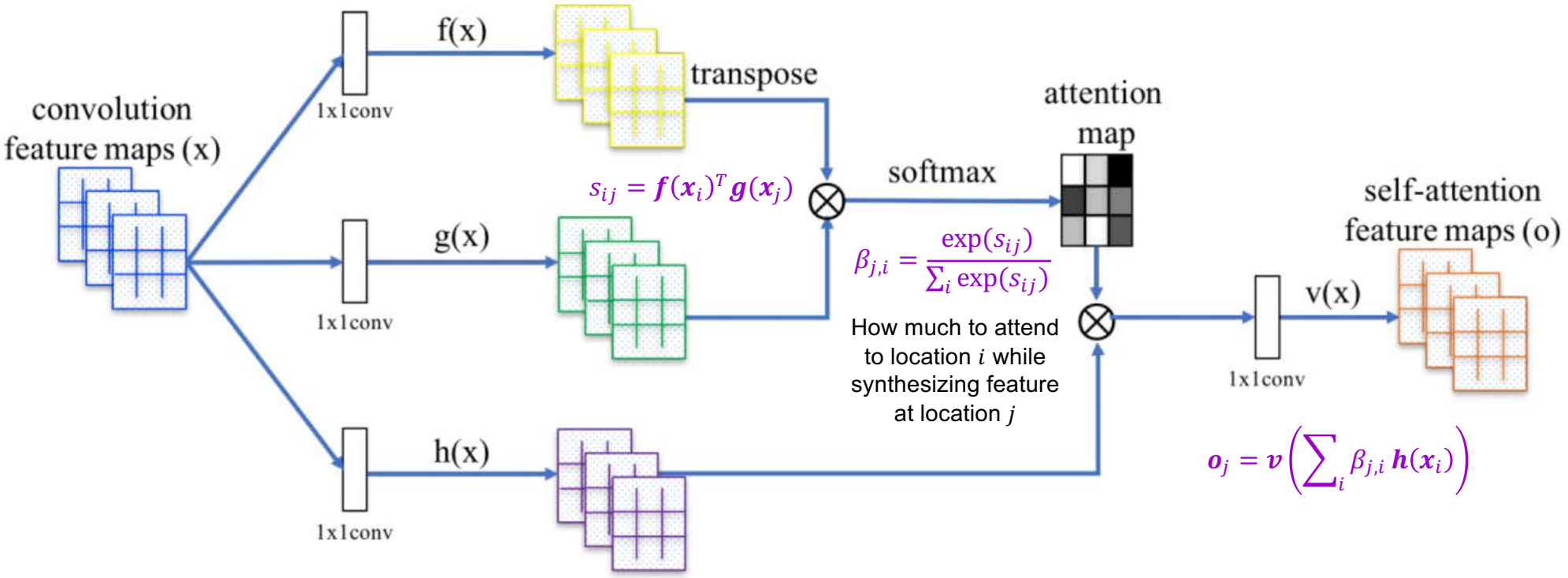
Self-attention GAN

- Adaptive receptive fields to capture non-local structure

H. Zhang, I. Goodfellow, D. Metaxas, A. Odena. Self-Attention Generative Adversarial Networks. ICML 2019

Self-attention GAN

- Adaptive receptive fields to capture non-local structure (based on Wang et al., 2018)



Self-attention GAN: Implementation details

- Hinge loss formulation:

$$L_D = -\mathbb{E}_{(x,y)\sim p_{\text{data}}}[\min(0, D(x,y) - 1)] \\ -\mathbb{E}_{z\sim p_z, y\sim p_{\text{data}}}[\min(0, -D(G(z,y),y) - 1)]$$

$$L_G = -\mathbb{E}_{z\sim p_z, y\sim p_{\text{data}}} D(G(z,y),y)$$

Self-attention GAN: Implementation details

- Hinge loss formulation
- Conditioning the discriminator: *projection* (Miyato & Koyama, 2018)
- Conditioning the generator: *conditional batch norm*

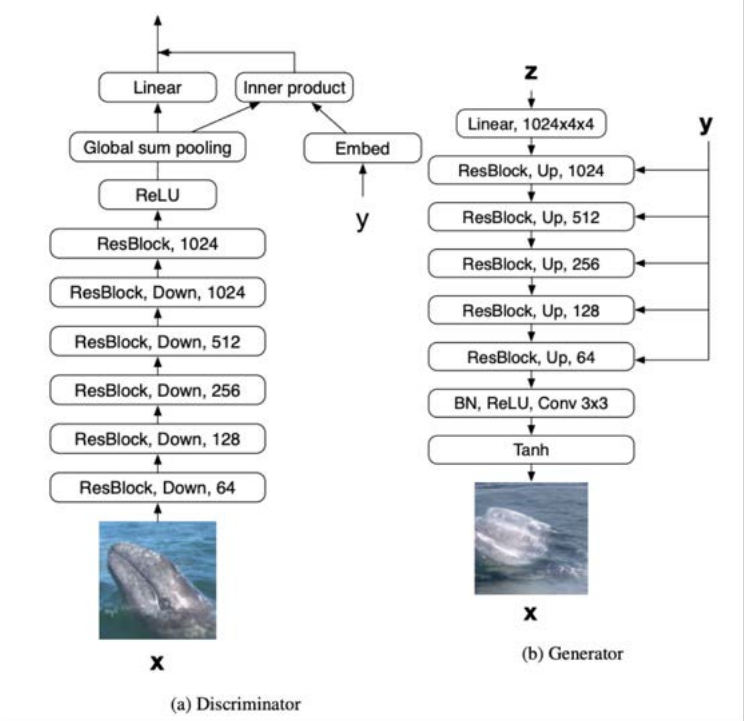


Figure source

Self-attention GAN: Implementation details

- Hinge loss formulation
- Conditioning the discriminator: *projection* (Miyato & Koyama, 2018)
- Conditioning the generator: *conditional batch norm*
- *Spectral normalization* for generator and discriminator (Miyato et al., 2018) – divide weight matrices by largest singular value (estimated)
- Different learning rates for generator and discriminator (TTUR – Heusel et al., 2017)

Self-attention GAN: Results

- 128 x 128 ImageNet

goldfish

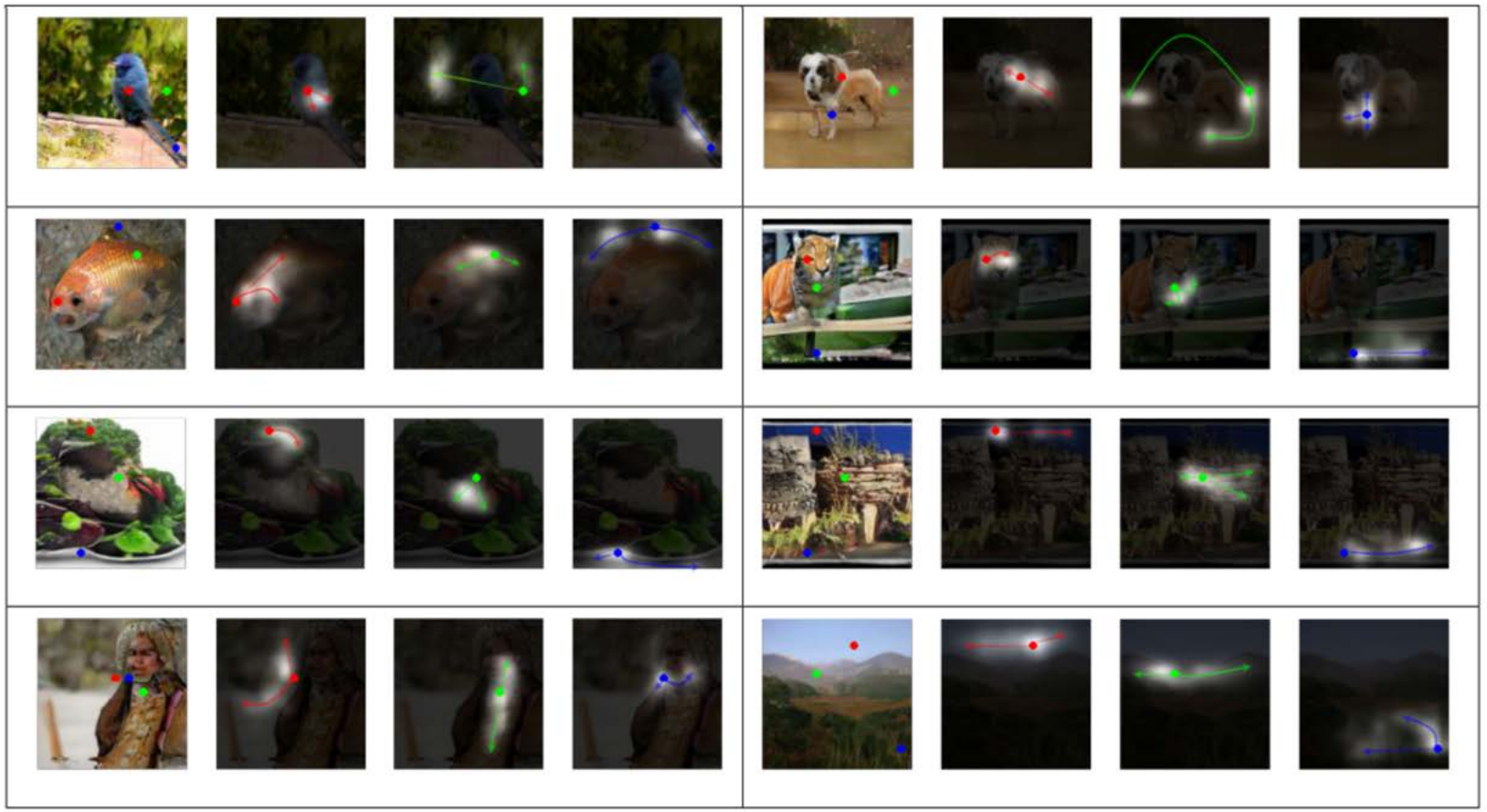
indigo bunting

redshank

Saint Bernard

Self-attention GAN: Results

- Attention map visualization



BigGAN

- Scale up SA-GAN to generate ImageNet images up to 512 x 512 resolution

A. Brock, J. Donahue, K. Simonyan, Large scale GAN training for high fidelity natural image synthesis, ICLR 2019

BigGAN: Implementation details

- 8x larger batch size, 50% more channels (2x more parameters) than baseline SA-GAN
- Hierarchical latent space: feed (transformations of) z vector into multiple layers of the generator

BigGAN: Implementation details

- 8x larger batch size, 50% more channels (2x more parameters) than baseline SA-GAN
- Hierarchical latent space: feed (transformations of) z vector into multiple layers of the generator
- Truncation trick: at test time, resample the values of the z vector with magnitude above a chosen threshold
  - Trade off diversity for image quality

"The effects of increasing truncation. From left to right, the threshold is set to 2, 1, 0.5, 0.04."

BigGAN: Implementation details

- 8x larger batch size, 50% more channels (2x more parameters) than baseline SA-GAN
- Hierarchical latent space: feed (transformations of) z vector into multiple layers of the generator
- Truncation trick: at test time, resample the values of the z vector with magnitude above a chosen threshold
- Lots of other tricks (initialization, training, etc.)
- Training observed to be unstable, but good results are achieved “just before collapse”
- Evidence that discriminator memorizes the training data, but the generator doesn’t

BigGAN: Implementation details

https://xkcd.com/1838/

BigGAN: Results

- Samples at 256 x 256 resolution:

BigGAN: Results

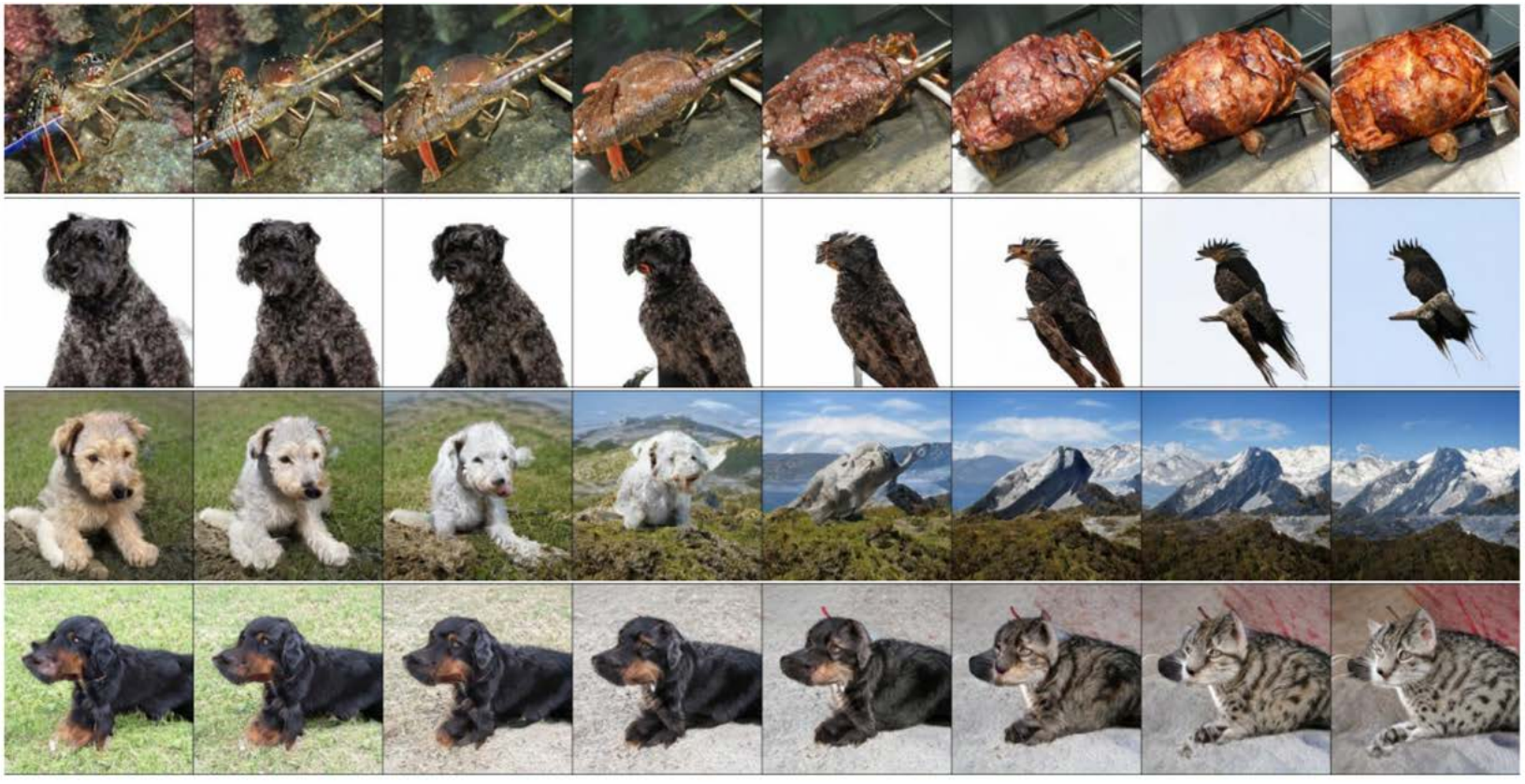
- Samples at 512 x 512 resolution:

BigGAN: Results

- Interpolation between c, z pairs:

BigGAN: Results

- Interpolation between c with z held constant:



BigGAN: Results

- Difficult classes:

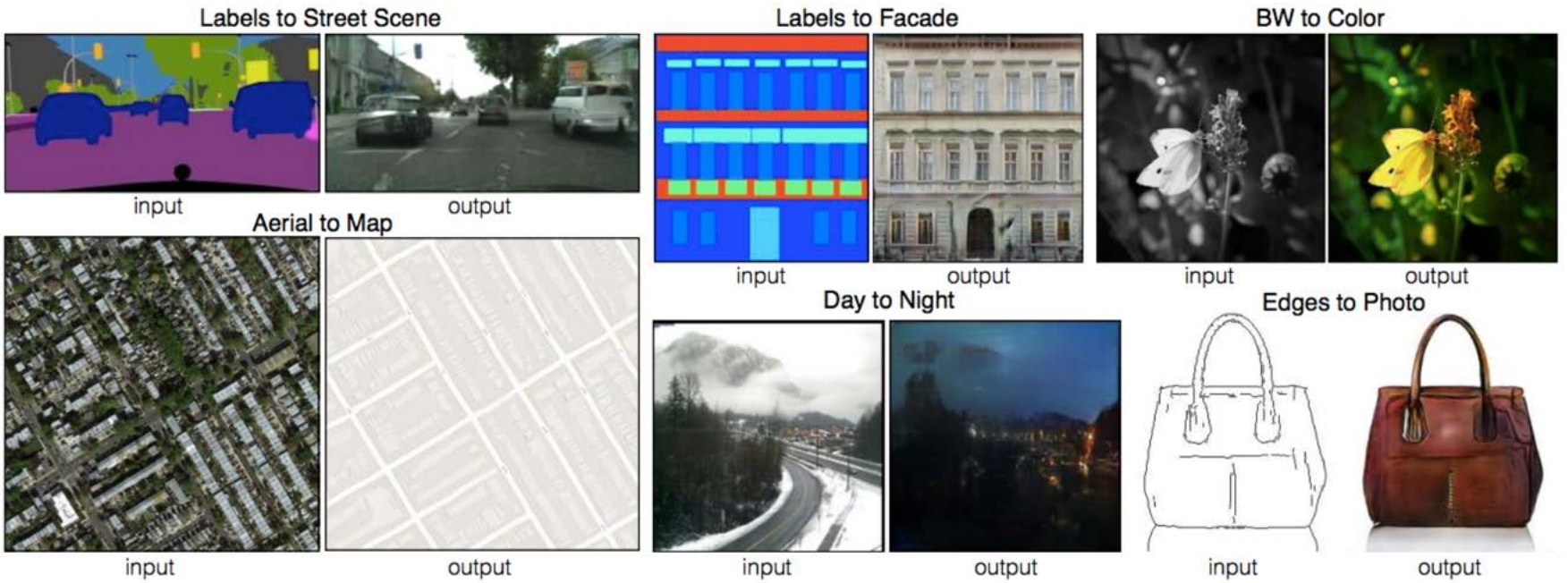
Announcements and reminders

- Assignment 3 deadline extended until the end of tomorrow, November 4
- Assignment 4 is out, due November 20 (right before Thanksgiving break)
  - Two parts – one on GANs, one on RNNs
  - Get started on the GAN part *now!*
- Project progress reports due Monday, November 16
  - Target length ~3 pages

Conditional GANs: Outline

- Introduction
- Generation conditioned on class
  - Self-attention GAN
  - BigGAN
- Generation conditioned on image
  - Paired image-to-image translation: pix2pix
  - Unpaired image-to-image translation: CycleGAN

Image-to-image translation



P. Isola, J.-Y. Zhu, T. Zhou, A. Efros, Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017

Image-to-image translation

- Produce modified image y conditioned on input image x (note change of notation)
  - Generator receives x as input
  - Discriminator receives an x, y pair and has to decide whether it is real or fake

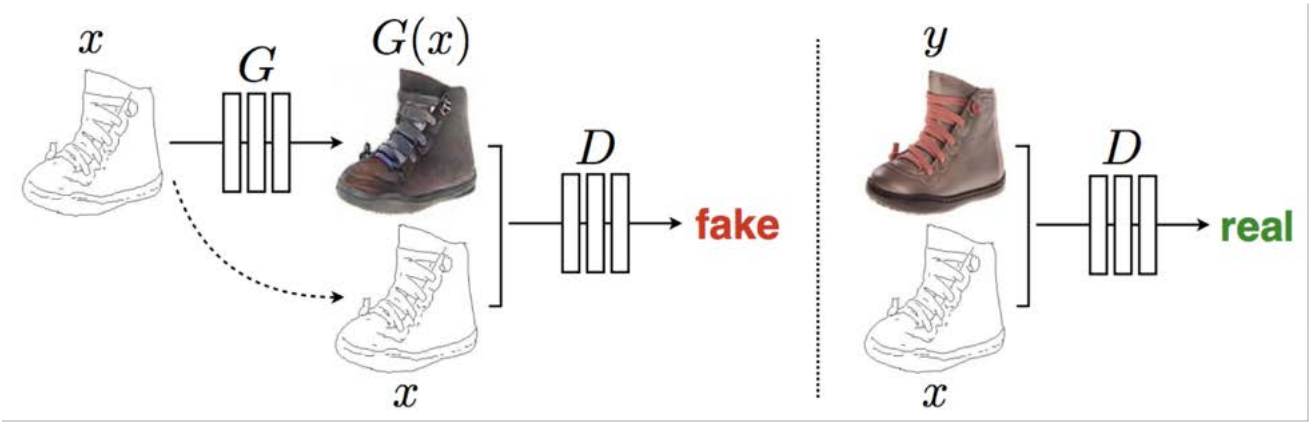
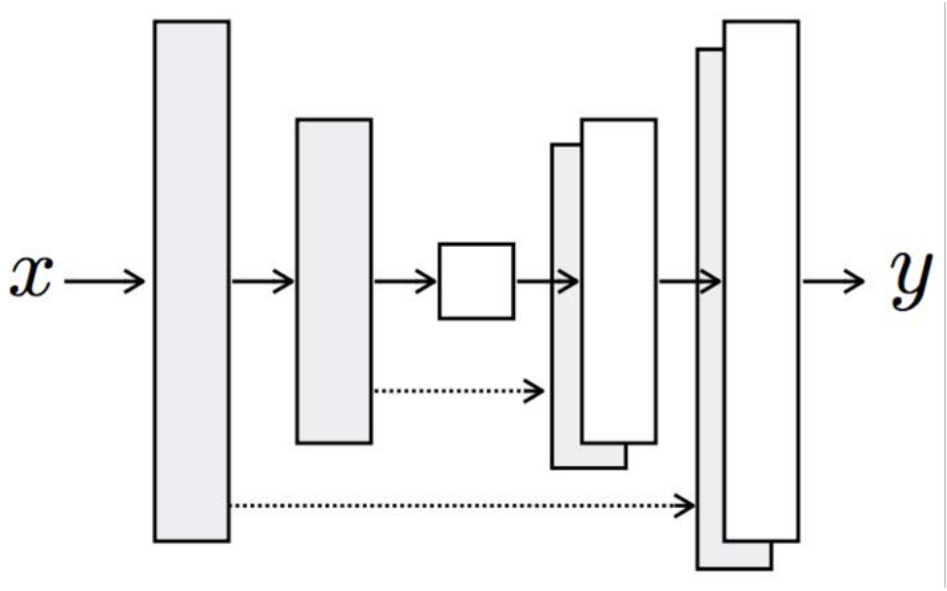


Image-to-image translation

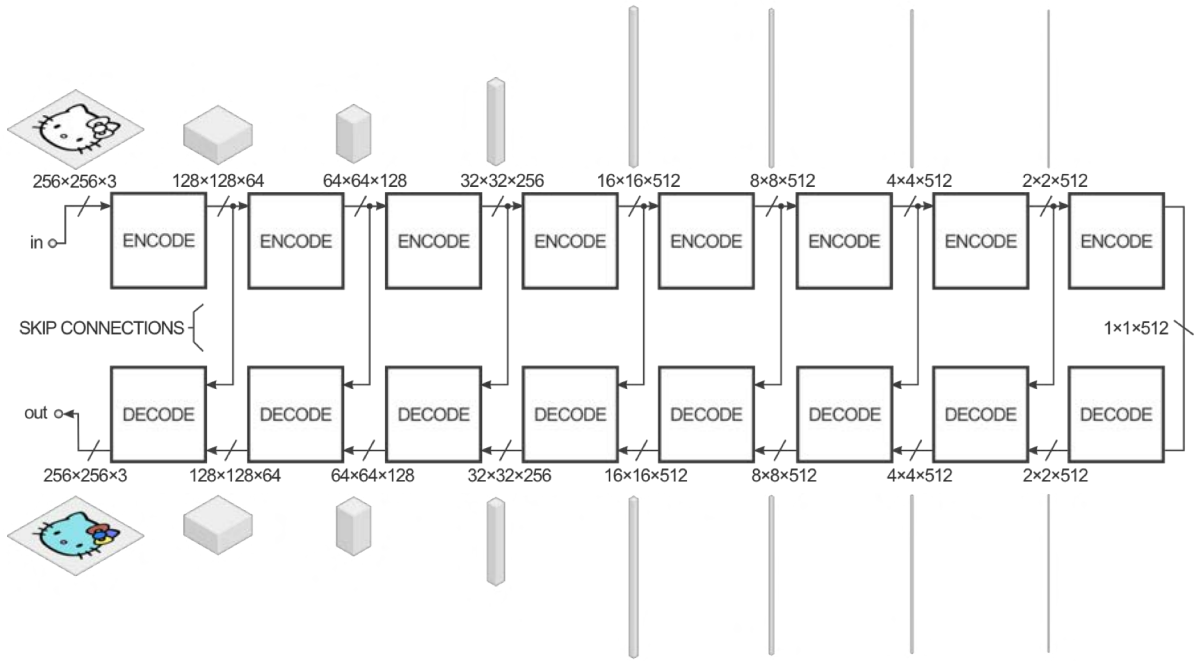
- Generator architecture: U-Net



- Note: no z used as input, transformation is basically deterministic

Image-to-image translation

- Generator architecture: U-Net



Encode: convolution → BatchNorm → ReLU

Decode: transposed convolution → BatchNorm → ReLU

Figure source

Image-to-image translation

- Generator architecture: U-Net

Effect of adding skip connections to the generator

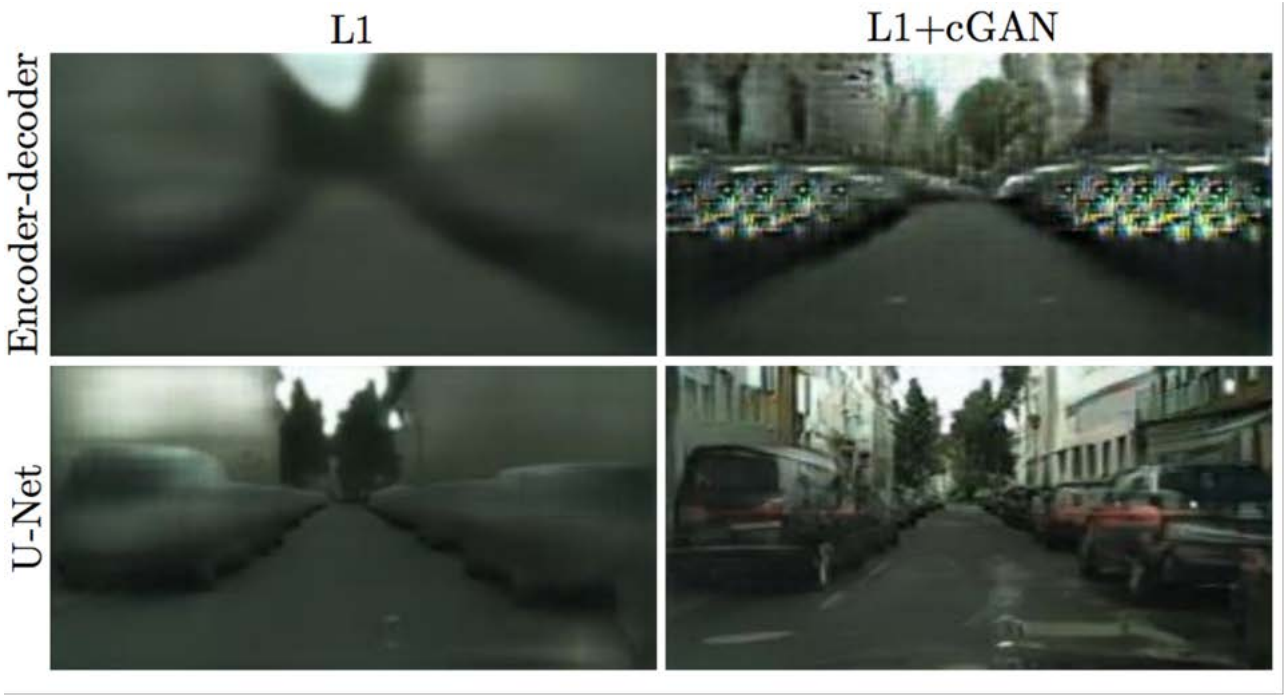


Image-to-image translation

- Generator loss: GAN loss plus L1 reconstruction penalty

$$G^* = \arg\min_G \max_D \mathcal{L}_{GAN}(G, D) + \lambda \sum_i \|y_i - G(x_i)\|_1$$

Generated output $G(x_i)$ should be close to ground truth target y_i

Image-to-image translation

- Generator loss: GAN loss plus L1 reconstruction penalty

$$G^* = \arg\min_G \max_D \mathcal{L}_{GAN}(G, D) + \lambda \sum_i \|y_i - G(x_i)\|_1$$

Image-to-image translation

- Discriminator: PatchGAN
  - Given input image x and second image y, decide whether y is a ground truth target or produced by the generator

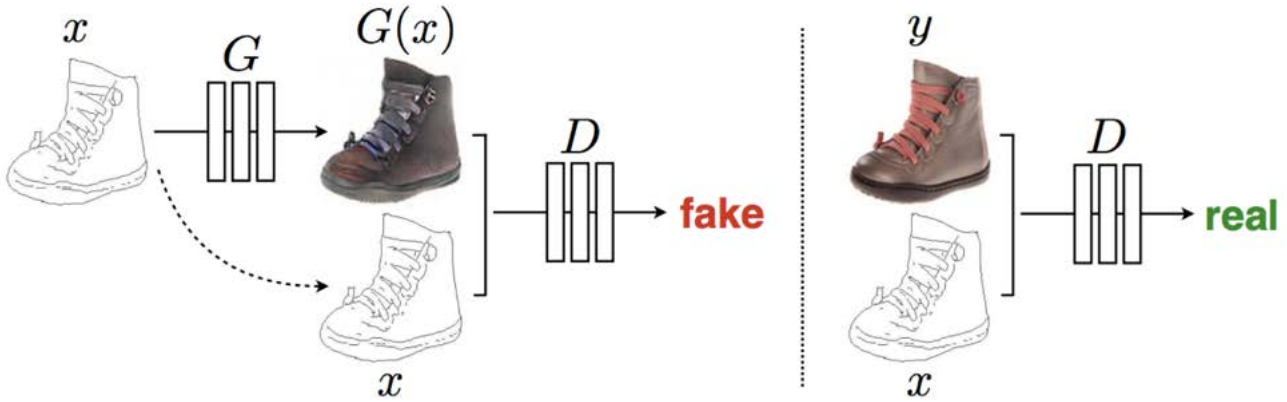


Image-to-image translation

- Discriminator: PatchGAN
  - Given input image x and second image y, decide whether y is a ground truth target or produced by the generator
  - Output is a 30 x 30 map where each value (0 to 1) represents the quality of the corresponding section of the output image, these values are averaged to obtain final discriminator loss
  - Fully convolutional network, effective patch size can be increased by increasing the depth

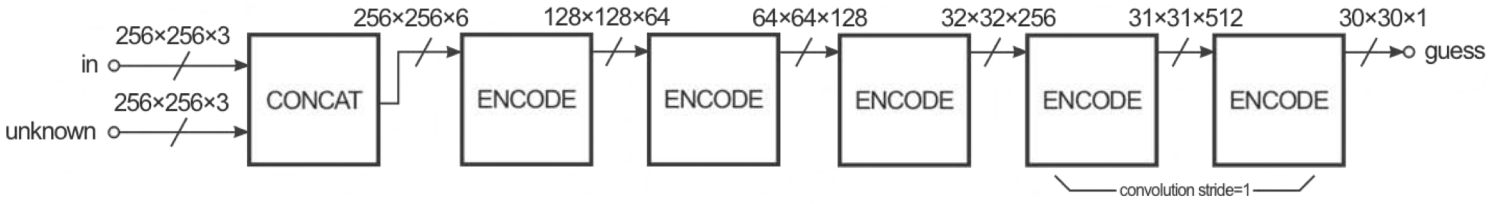


Figure source

Image-to-image translation

- Discriminator: PatchGAN
  - Given input image x and second image y, decide whether y is a ground truth target or produced by the generator
  - Output is a 30 x 30 map where each value (0 to 1) represents the quality of the corresponding section of the output image, these values are averaged to obtain final discriminator loss
  - Fully convolutional network, effective patch size can be increased by increasing the depth

Effect of discriminator patch size on generator output

Image-to-image translation: Results

- Translating between maps and aerial photos

Image-to-image translation: Results

- Translating between maps and aerial photos
- Human study:

	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%**

Image-to-image translation: Results

- Semantic labels to scenes

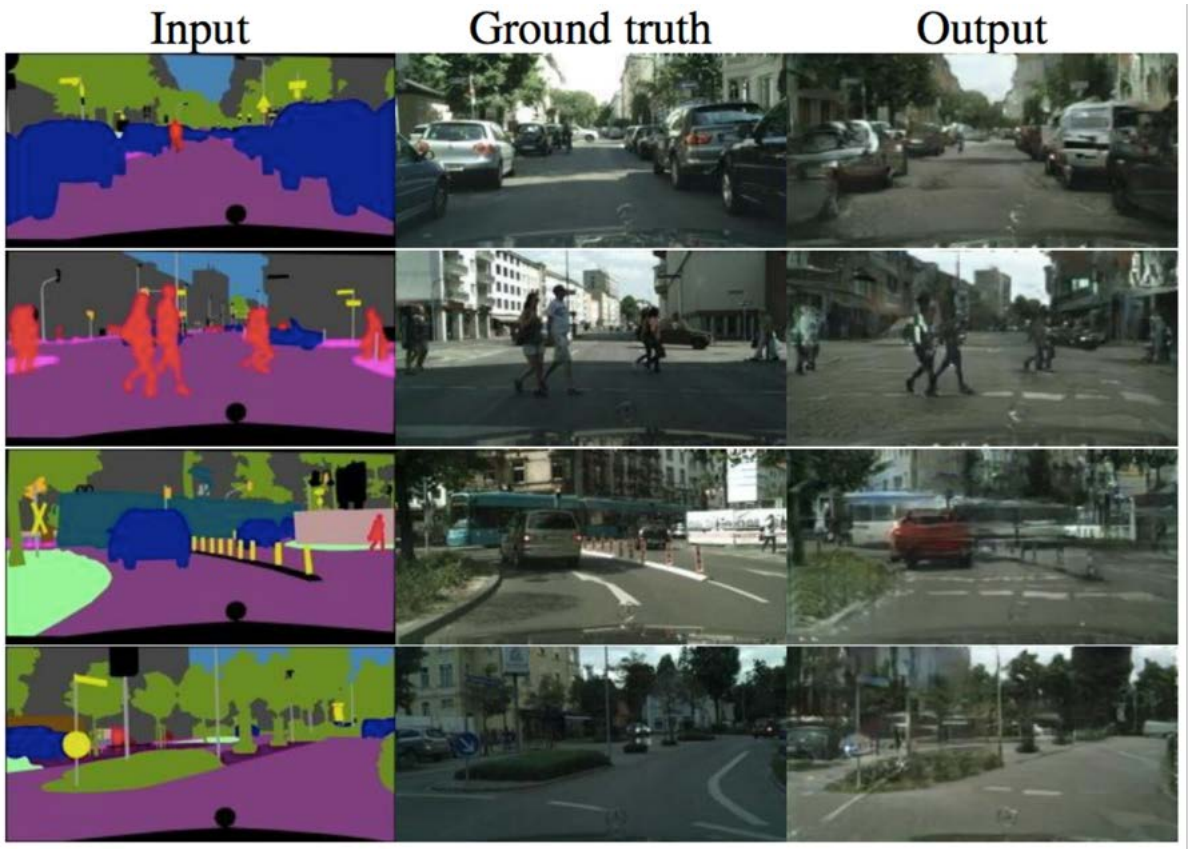


Image-to-image translation: Results

- Semantic labels to scenes
- Evaluation: FCN score
  - The higher the quality of the output, the better the FCN should do at recovering the original semantic labels

Loss	Per-pixel acc.	Per-class acc.	Class IOU
L1	0.42	0.15	0.11
GAN	0.22	0.05	0.01
cGAN	0.57	0.22	0.16
L1+GAN	0.64	0.20	0.15
L1+cGAN	**0.66**	**0.23**	**0.17**
Ground truth	0.80	0.26	0.21

Image-to-image translation: Results

- Scenes to semantic labels

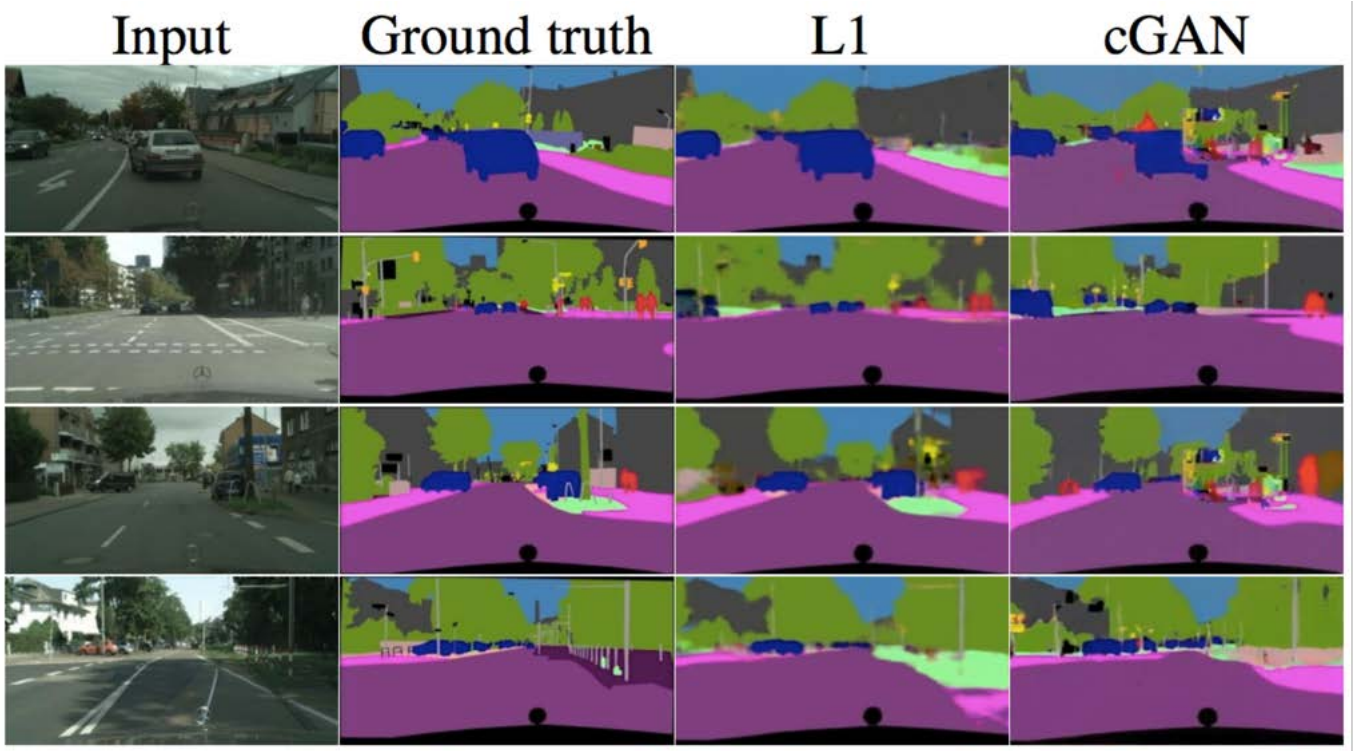


Image-to-image translation: Results

- Scenes to semantic labels
- Accuracy is worse than that of regular FCNs or generator with L1 loss

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

Image-to-image translation: Results

- Semantic labels to facades

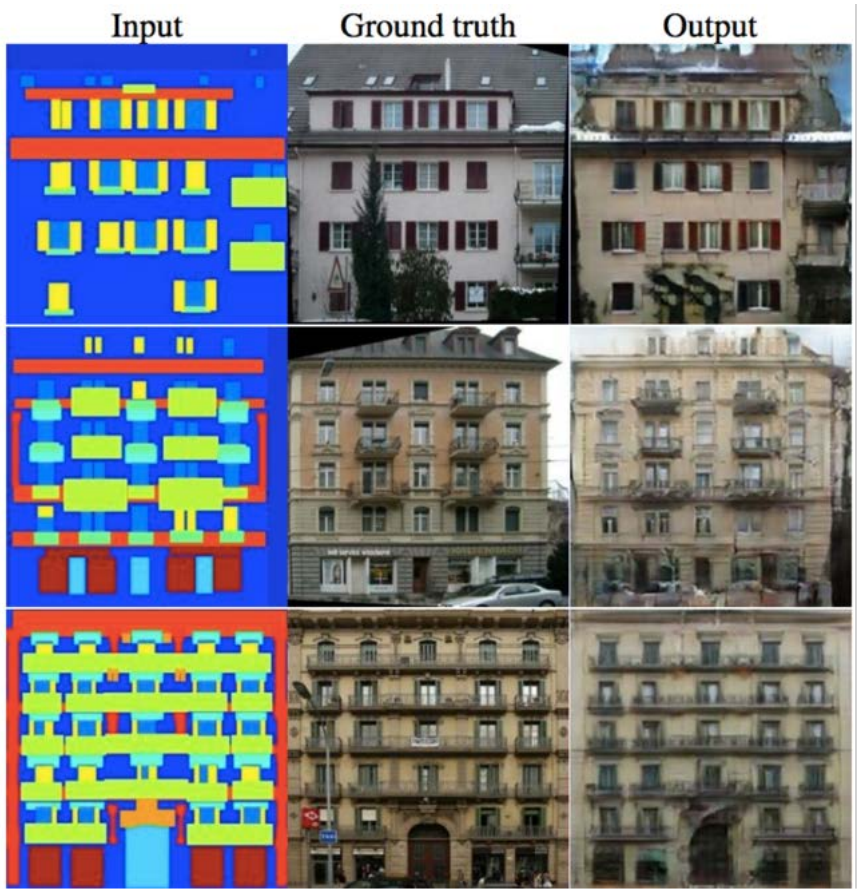


Image-to-image translation: Results

- Day to night

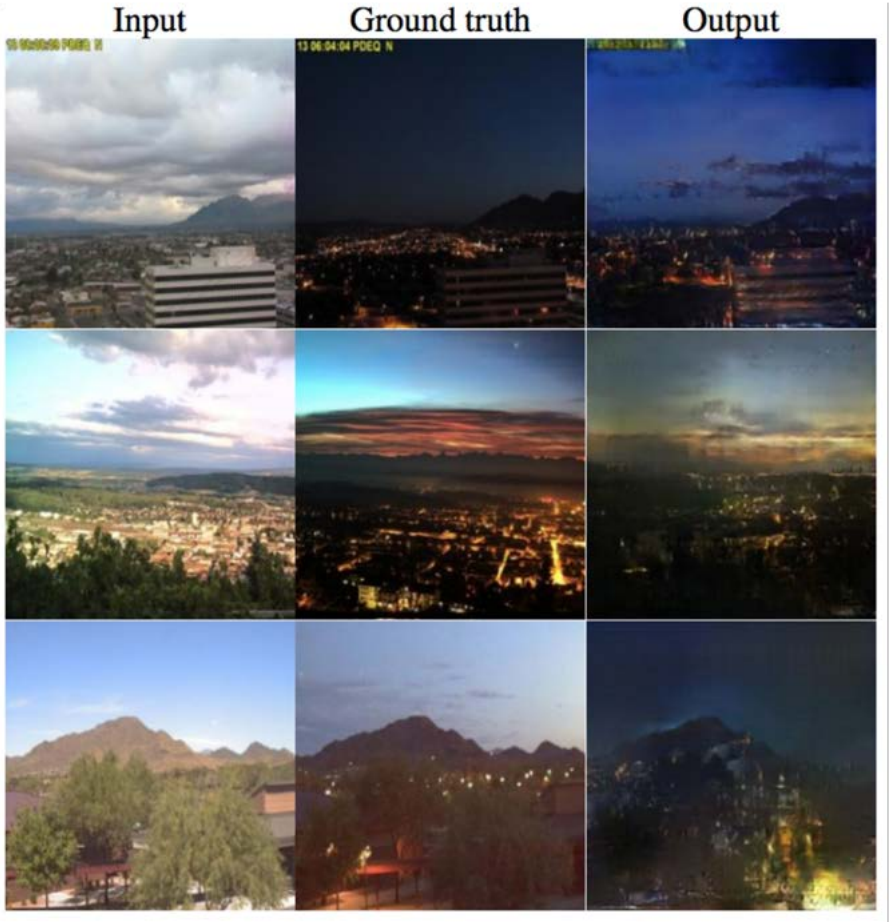


Image-to-image translation: Results

- Edges to photos

Image-to-image translation: Results

- [pix2pix demo]

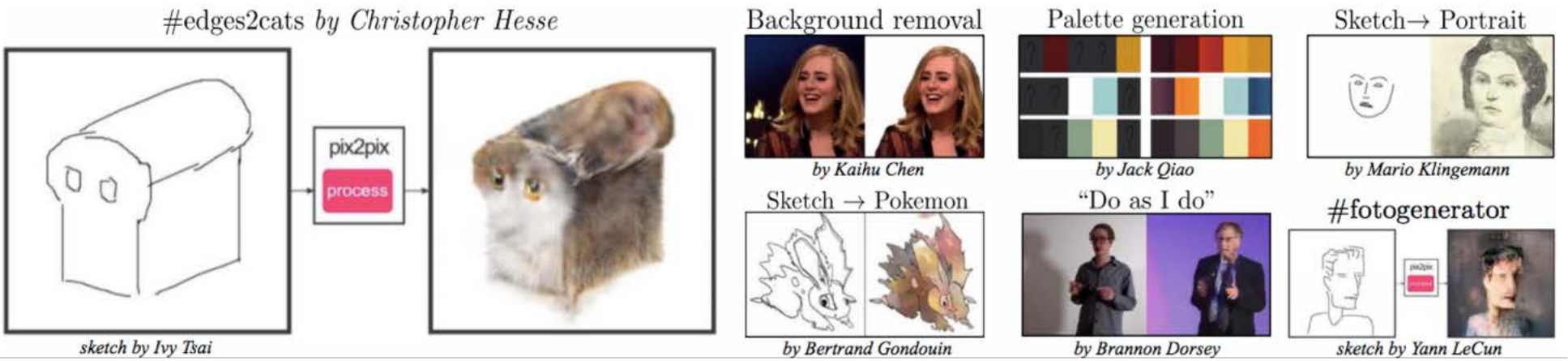
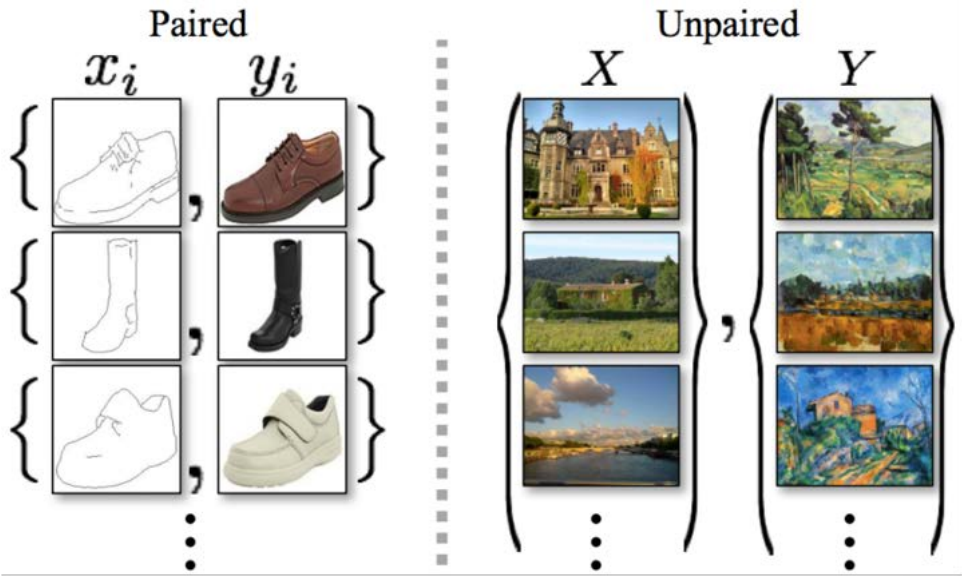


Image-to-image translation: Limitations

- Visual quality could be improved
- Requires x, y pairs for training
- Does not model conditional distribution $P(y|x)$, returns a single mode instead

Unpaired image-to-image translation

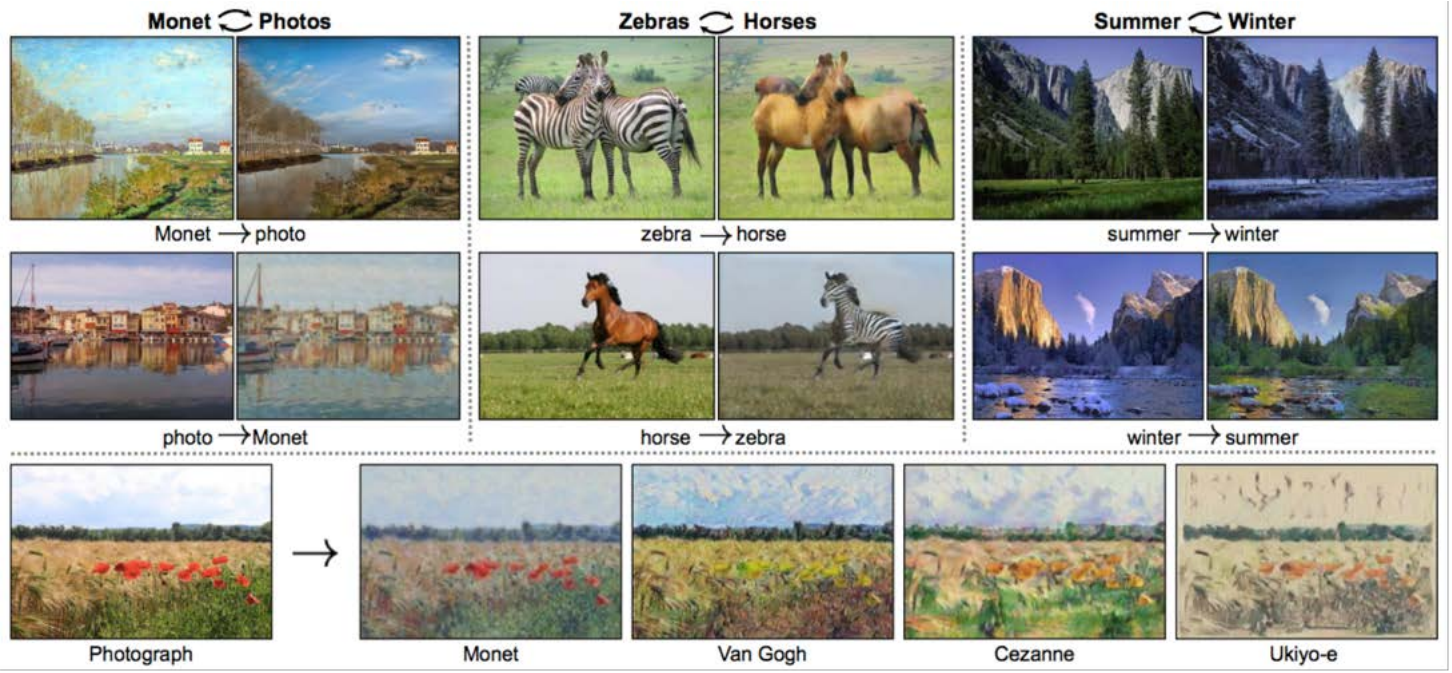
- Given two unordered image collections X and Y, learn to "translate" an image from one into the other and vice versa



J.-Y. Zhu, T. Park, P. Isola, A. Efros, Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks, ICCV 2017

Unpaired image-to-image translation

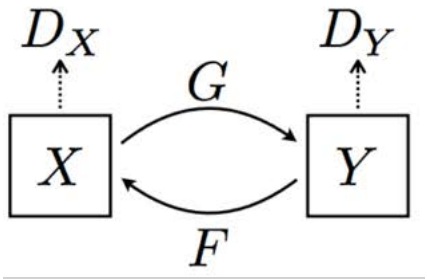
- Given two unordered image collections X and Y, learn to "translate" an image from one into the other and vice versa



J.-Y. Zhu, T. Park, P. Isola, A. Efros, Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks, ICCV 2017

CycleGAN

- Given: domains X and Y
- Train two generators F and G and two discriminators D_X and D_Y
  - G translates from X to Y, F translates from Y to X
  - D_X recognizes images from X, D_Y from Y
  - *Cycle consistency*: we want $F(G(x)) \approx x$ and $G(F(y)) \approx y$



CycleGAN: Architecture

- Generators (based on Johnson et al., 2016):

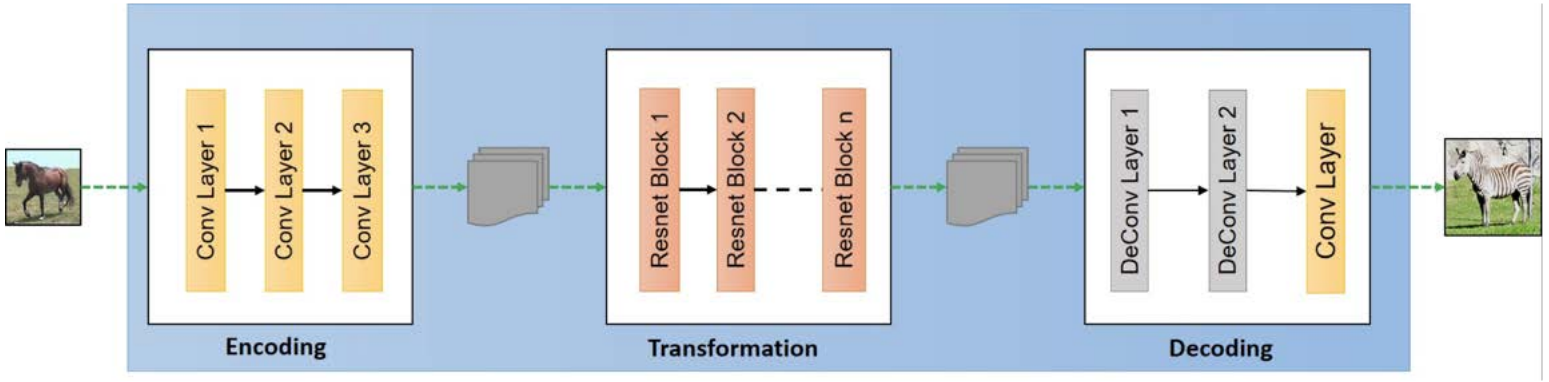


Figure source

- Discriminators: PatchGAN on 70 x 70 patches

CycleGAN: Loss

- Requirements:
  - G translates from X to Y, F translates from Y to X
  - D_X recognizes images from X, D_Y from Y
  - We want $F(G(x)) \approx x$ and $G(F(y)) \approx y$
- CycleGAN discriminator loss: LSGAN

$$\mathcal{L}_{\text{GAN}}(D_Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)}[(D_Y(y) - 1)^2] + \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[D_Y\left(G(x)\right)^2\right]$$

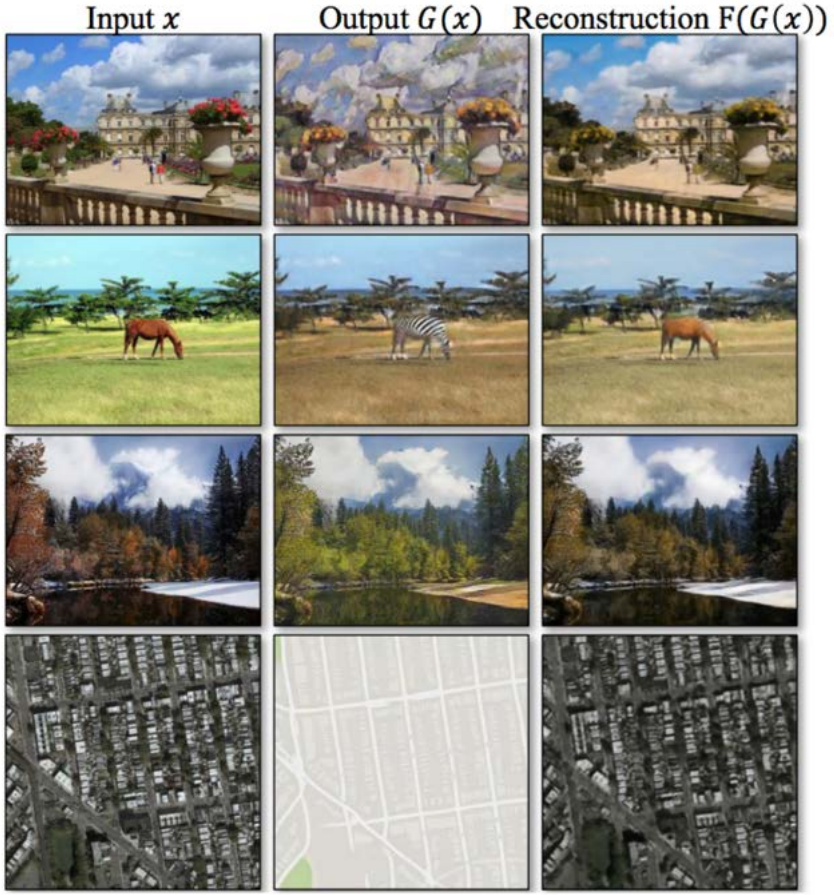
$$\mathcal{L}_{\text{GAN}}(D_X) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[(D_X(x) - 1)^2] + \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[D_X\left(F(y)\right)^2\right]$$

- CycleGAN generator loss:

$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[D_Y(G(x) - 1)^2] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[D_X(F(y) - 1)^2]$$
$$+ \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[\left\|F\left(G(x)\right) - x\right\|_1\right] + \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[\left\|G\left(F(y)\right) - y\right\|_1\right]$$

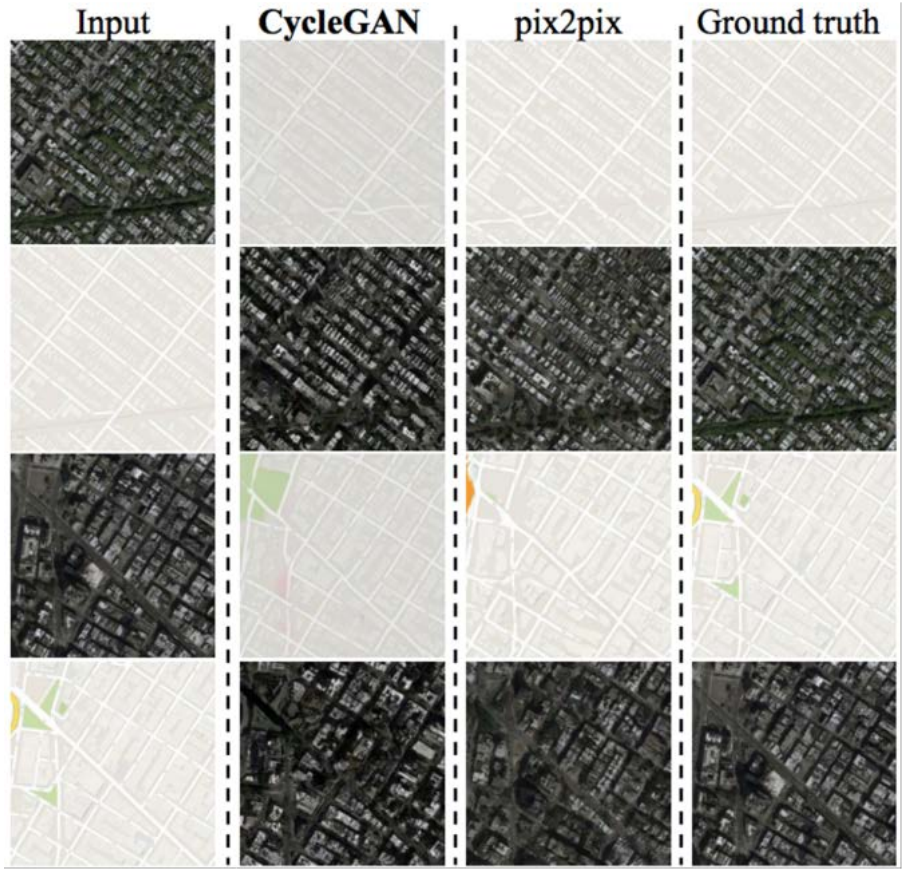
CycleGAN

- Illustration of cycle consistency:



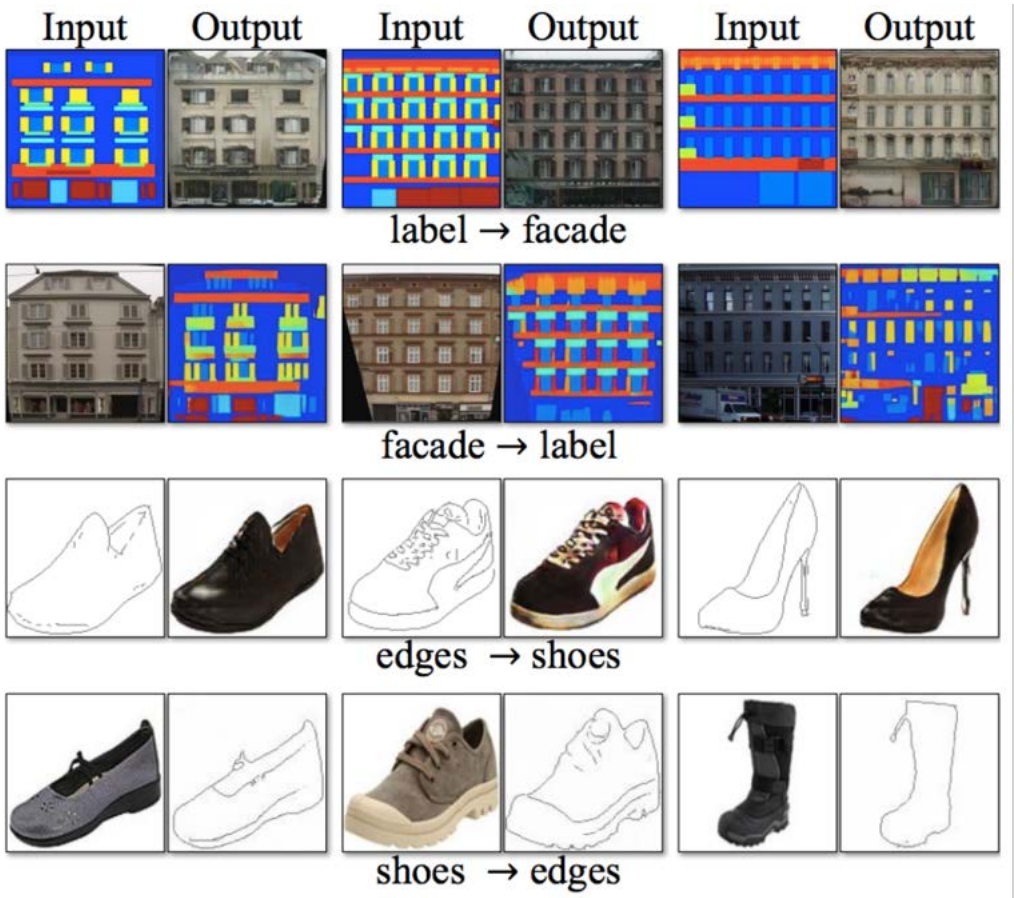
CycleGAN: Results

- Translation between maps and aerial photos



CycleGAN: Results

- Other pix2pix tasks



CycleGAN: Results

- Scene to labels and labels to scene
  - Worse performance than pix2pix due to lack of paired training data

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

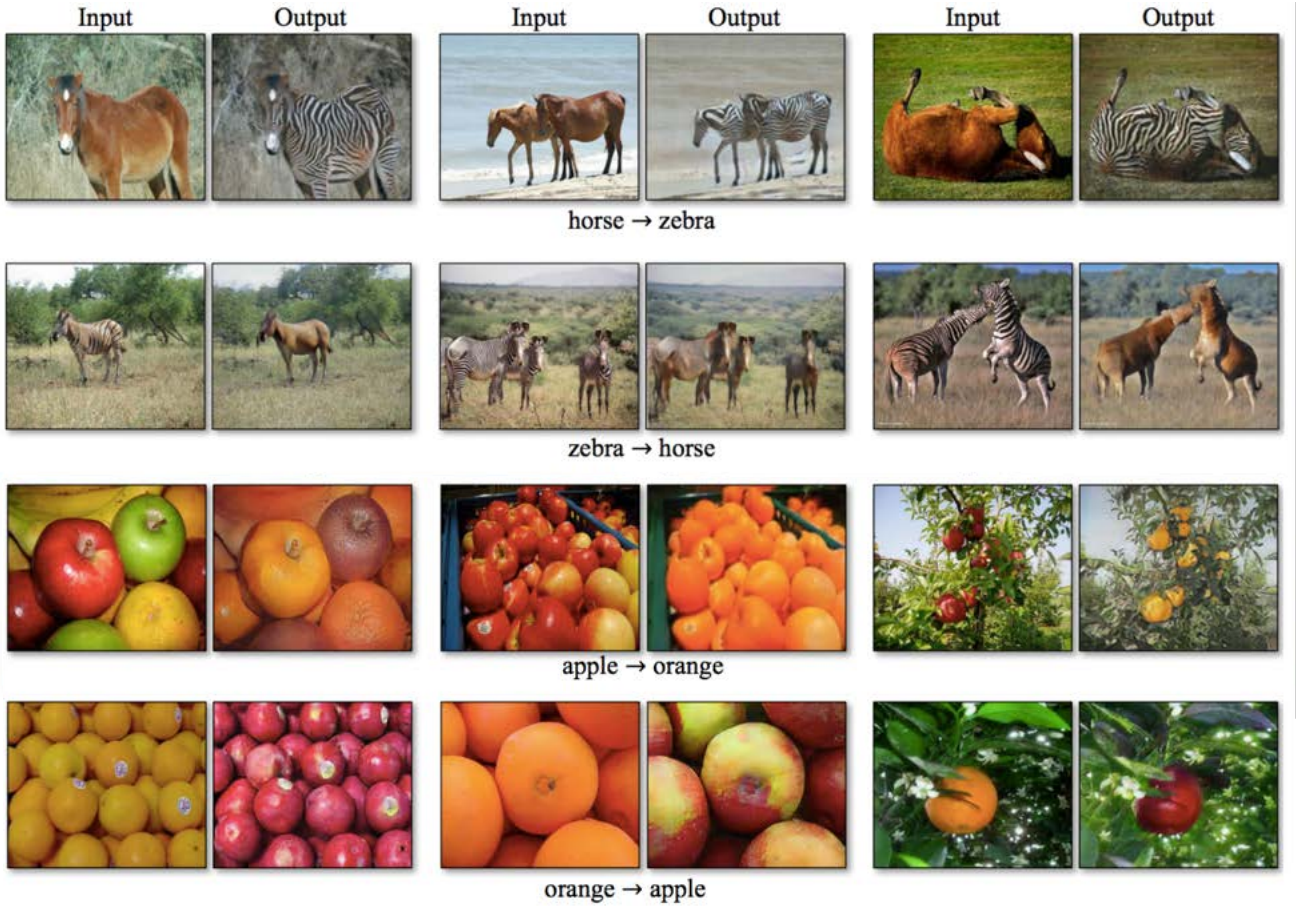
Table 2: FCN-scores for different methods, evaluated on Cityscapes labels→photo.

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.

CycleGAN: Results

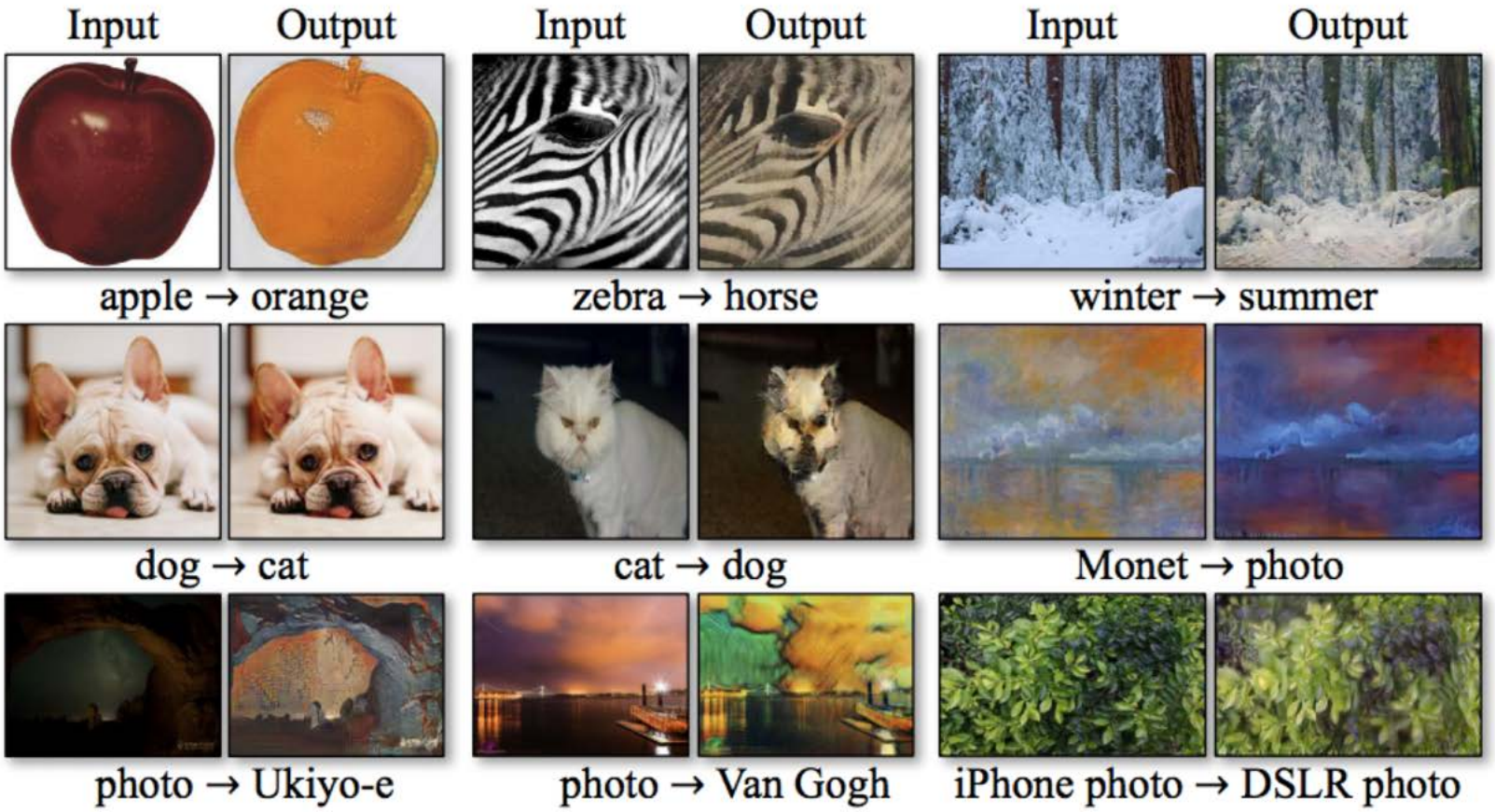
- Tasks for which paired data is unavailable



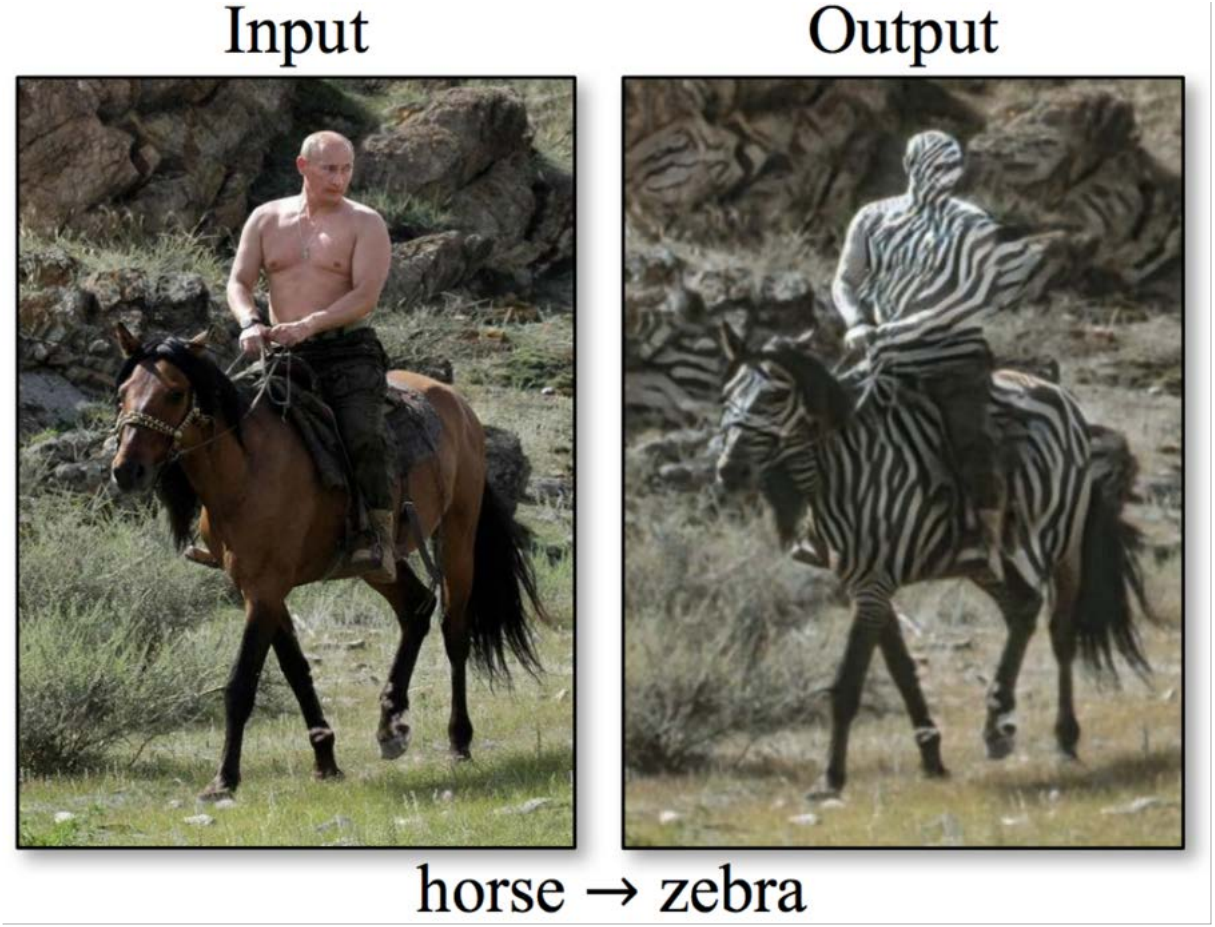
CycleGAN: Results

- Style transfer

CycleGAN: Failure cases



CycleGAN: Failure cases



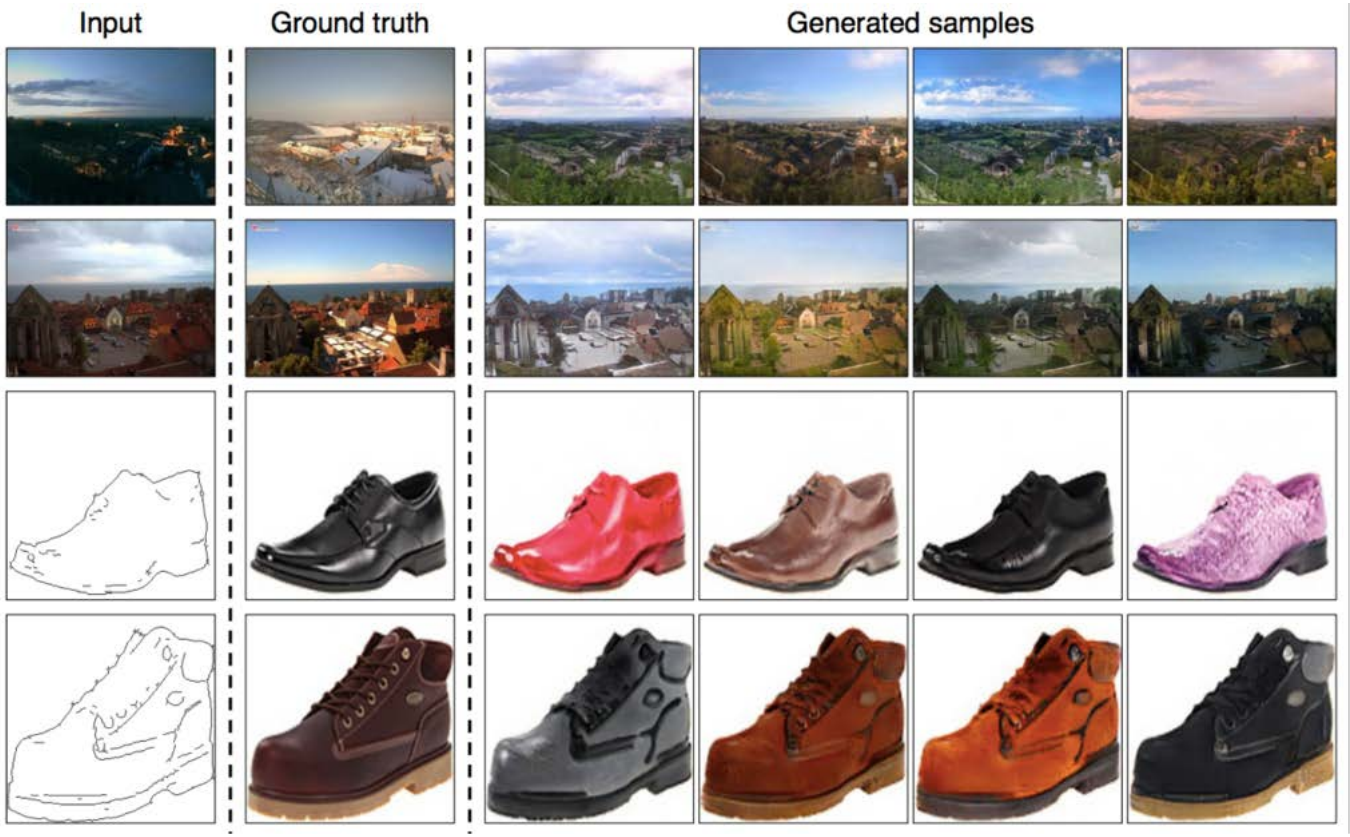
CycleGAN: Limitations

- Cannot handle shape changes (e.g., dog to cat)
- Can get confused on images outside of the training domains (e.g., horse with rider)
- Cannot close the gap with paired translation methods
- Does not account for the fact that one transformation direction may be more challenging than the other

Outline

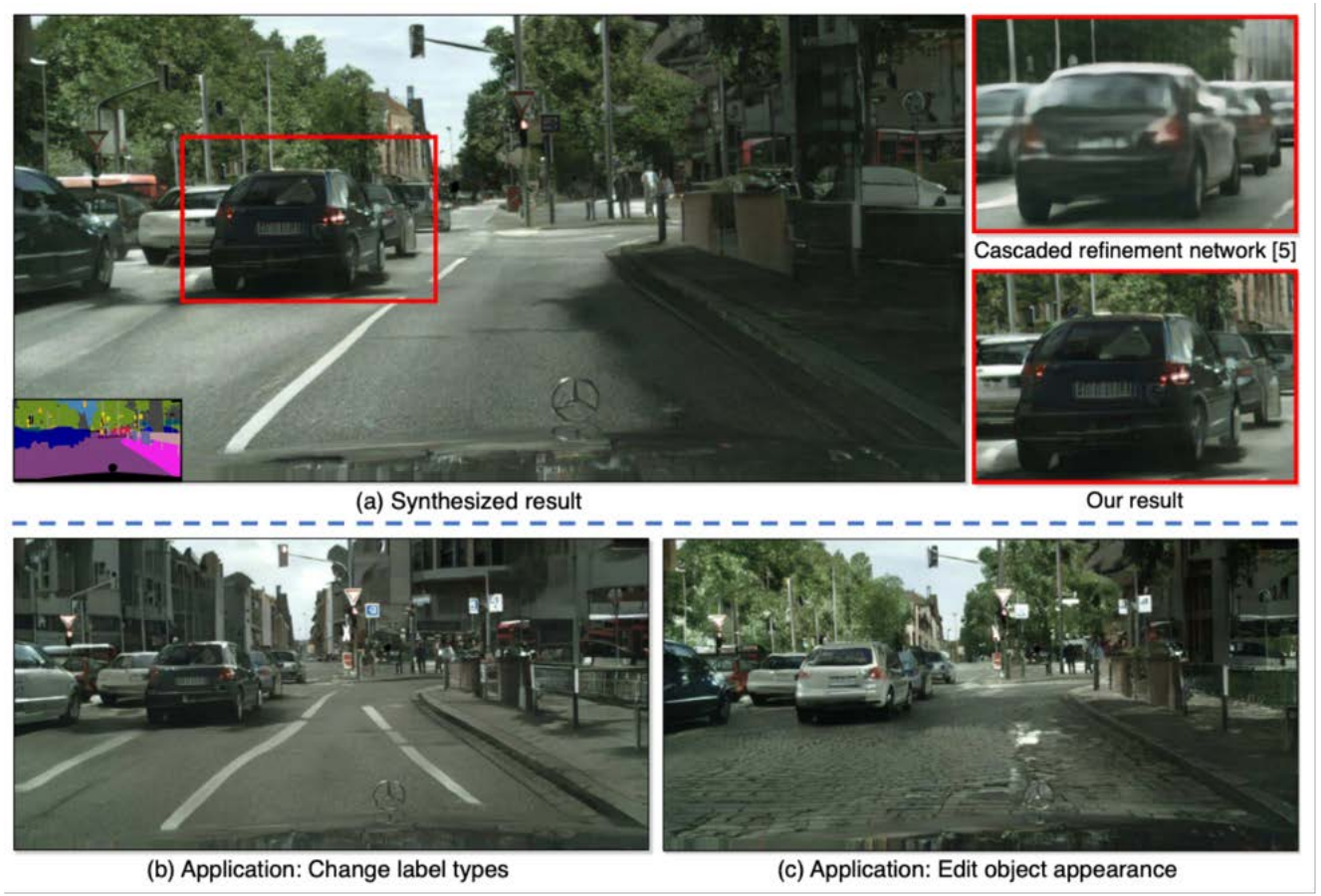
- Introduction
- Generation conditioned on class
  - Self-attention GAN
  - BigGAN
- Generation conditioned on image
  - Paired image-to-image translation: pix2pix
  - Unpaired image-to-image translation: CycleGAN
- Some recent highlights

Multimodal image-to-image translation



J.Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, E. Shechtman, Toward Multimodal Image-to-Image Translation, NIPS 2017

High-resolution, high-quality pix2pix

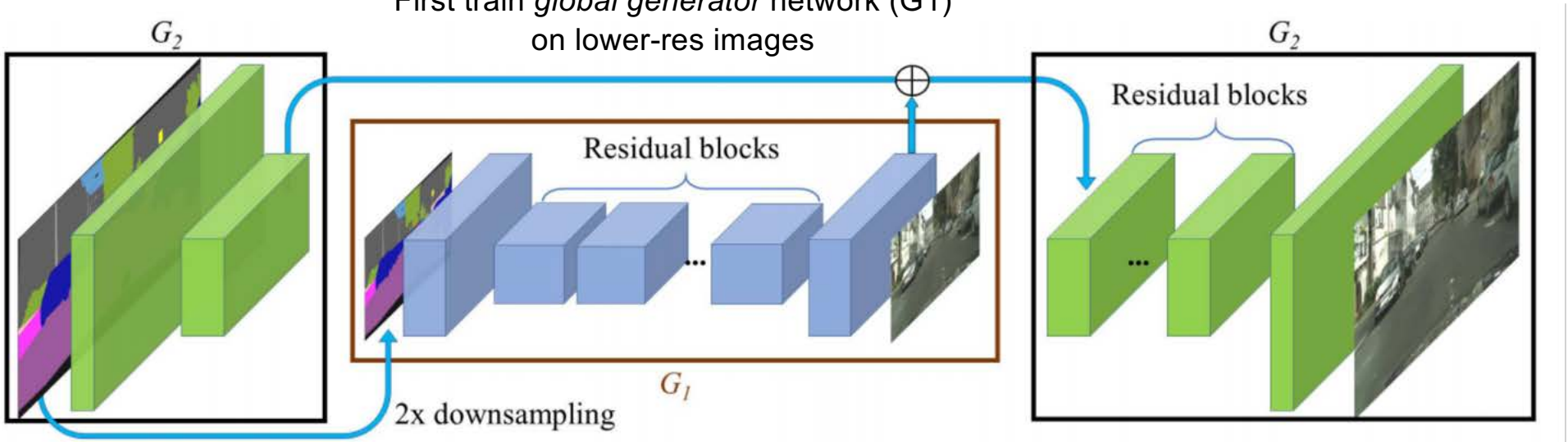


T.-C. Wang et al., High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs, CVPR 2018

High-resolution, high-quality pix2pix

- Two-scale generator architecture (up to 2048 x 1024 resolution)

First train *global generator* network (G1) on lower-res images



Then append higher-res *enhancer network* (G2) blocks and train G1 and G2 jointly

T.-C. Wang et al., High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs, CVPR 2018

High-resolution, high-quality pix2pix

- Two-scale generator architecture (up to 2048 x 1024 resolution)
- Three-scale discriminator architecture (full res, 2x and 4x downsampled)
- Incorporate feature matching loss into discriminator

Human generation conditioned on pose

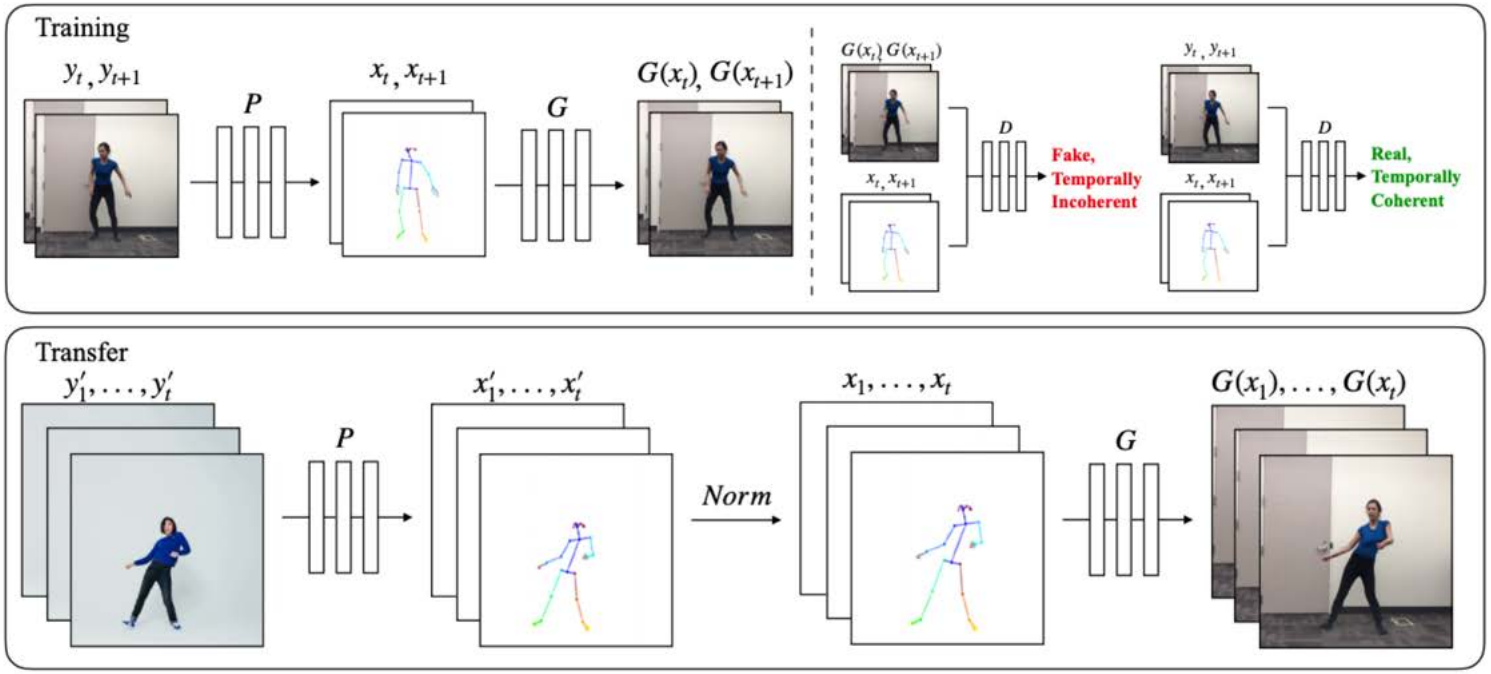
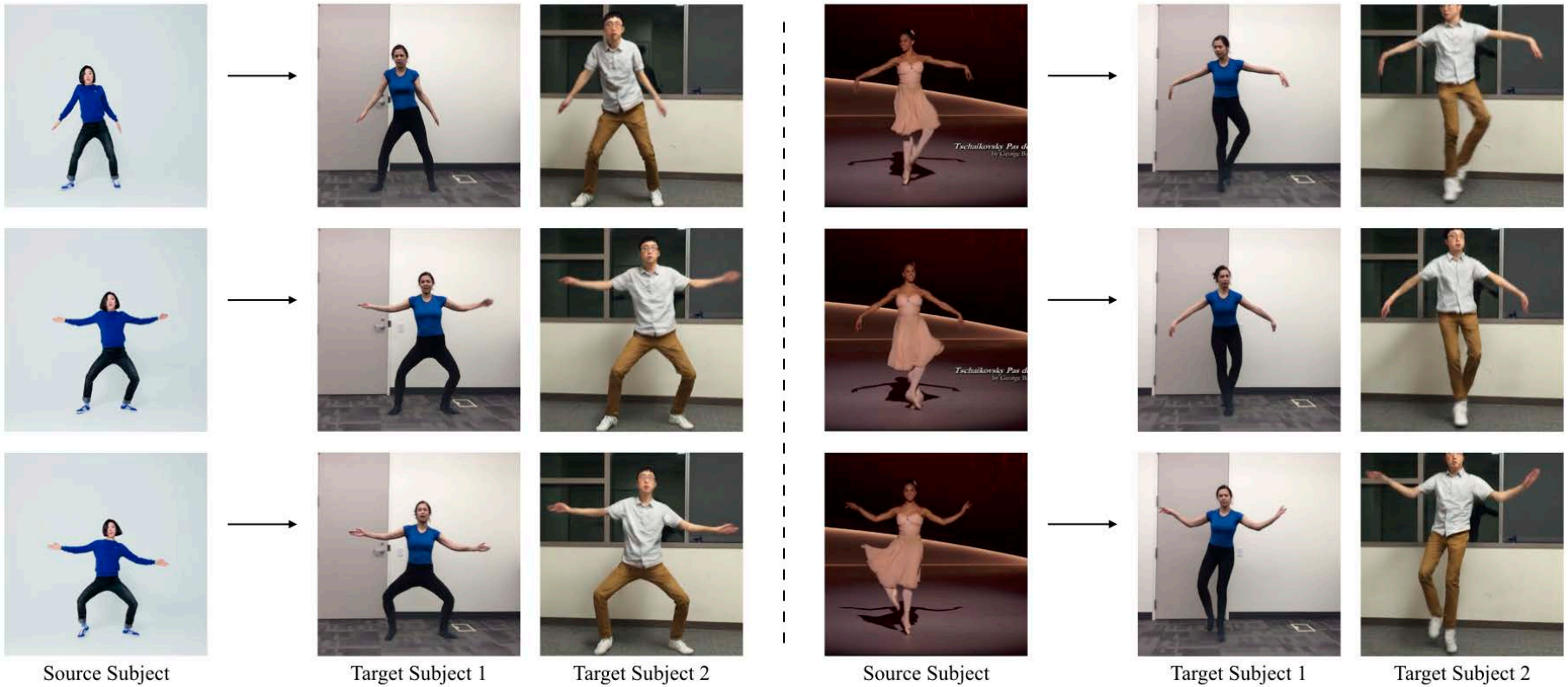


Figure 3: (Top) **Training**: Our model uses a pose detector P to create pose stick figures from video frames of the target subject. We learn the mapping G alongside an adversarial discriminator D which attempts to distinguish between the "real" correspondences $(x_t, x_{t+1}), (y_t, y_{t+1})$ and the "fake" sequence $(x_t, x_{t+1}), (G(x_t), G(x_{t+1}))$. (Bottom) **Transfer**: We use a pose detector P to obtain pose joints for the source person that are transformed by our normalization process $Norm$ into joints for the target person for which pose stick figures are created. Then we apply the trained mapping G.

C. Chan, S. Ginosar, T. Zhou, A. Efros. Everybody Dance Now. ICCV 2019

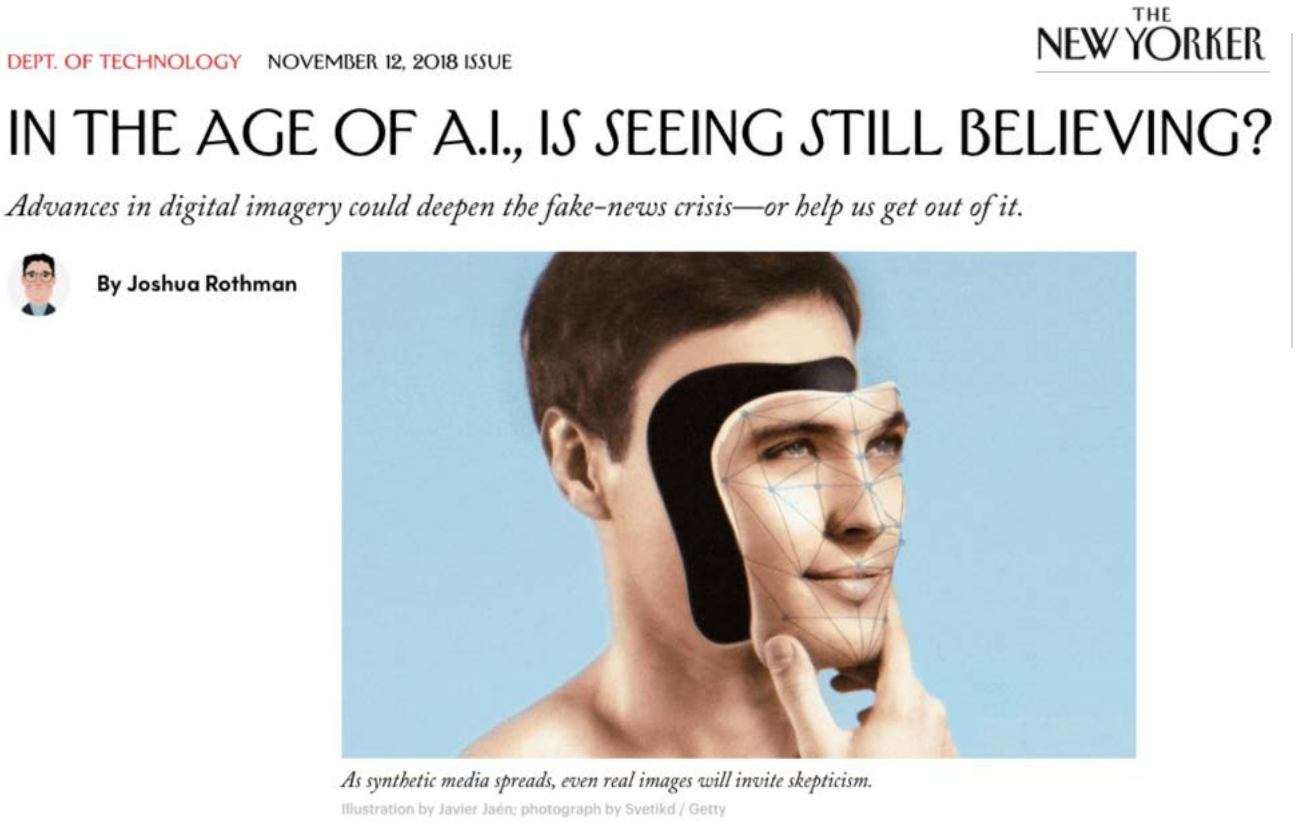
Human generation conditioned on pose



https://carolineec.github.io/everybody_dance_now/

C. Chan, S. Ginosar, T. Zhou, A. Efros. Everybody Dance Now. ICCV 2019

DeepFakes (coming up at the end of the course…)



DEPT. OF TECHNOLOGY NOVEMBER 12, 2018 ISSUE

THE NEW YORKER

IN THE AGE OF A.I., IS SEEING STILL BELIEVING?

Advances in digital imagery could deepen the fake-news crisis—or help us get out of it.

By Joshua Rothman

As synthetic media spreads, even real images will invite skepticism.

Illustration by Javier Jaén; photograph by Svetikd / Getty

https://www.newyorker.com/magazine/2018/11/12/in-the-age-of-ai-is-seeing-still-believing