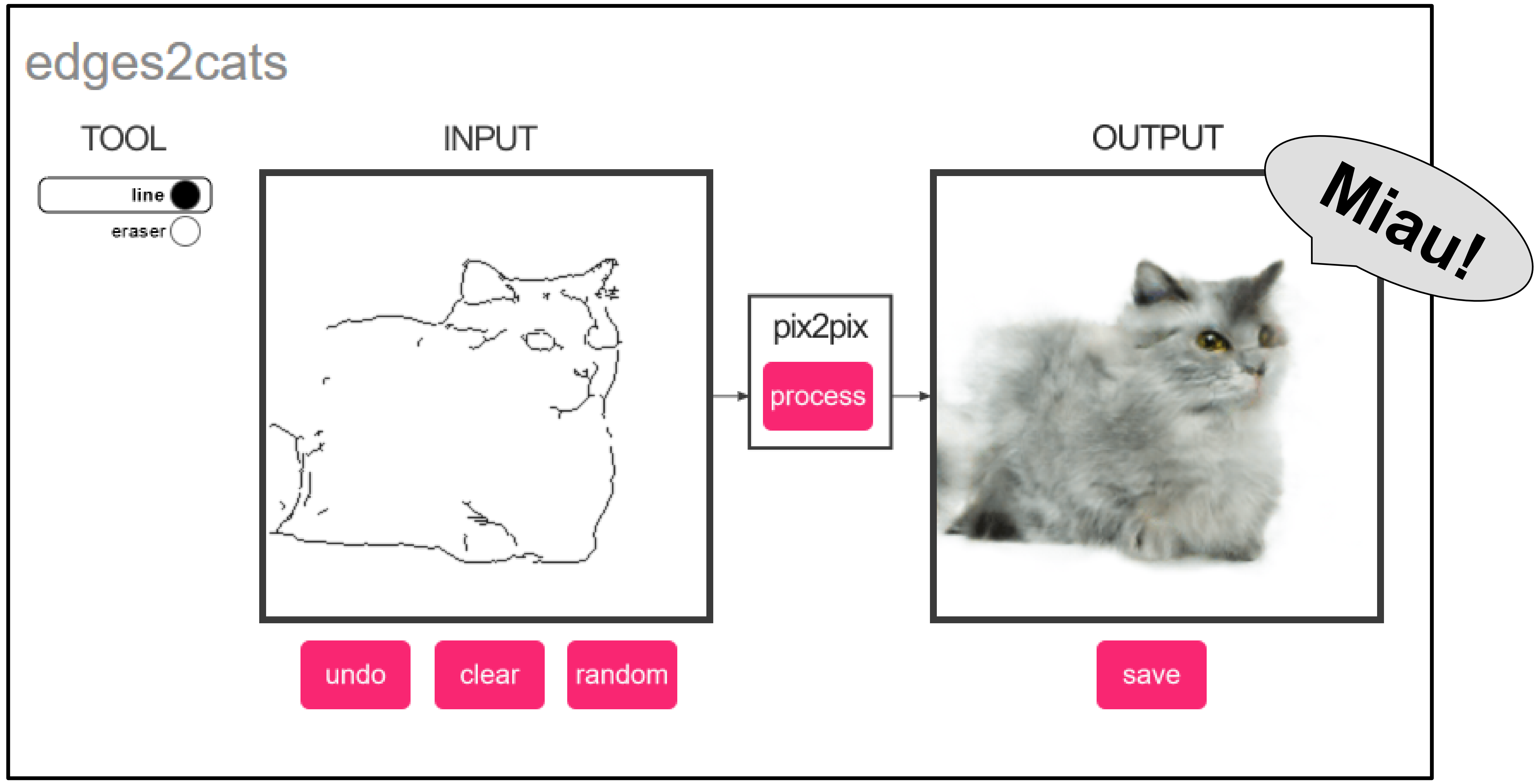
Conditional Adversarial Networks

(or “mapping from A to B”)

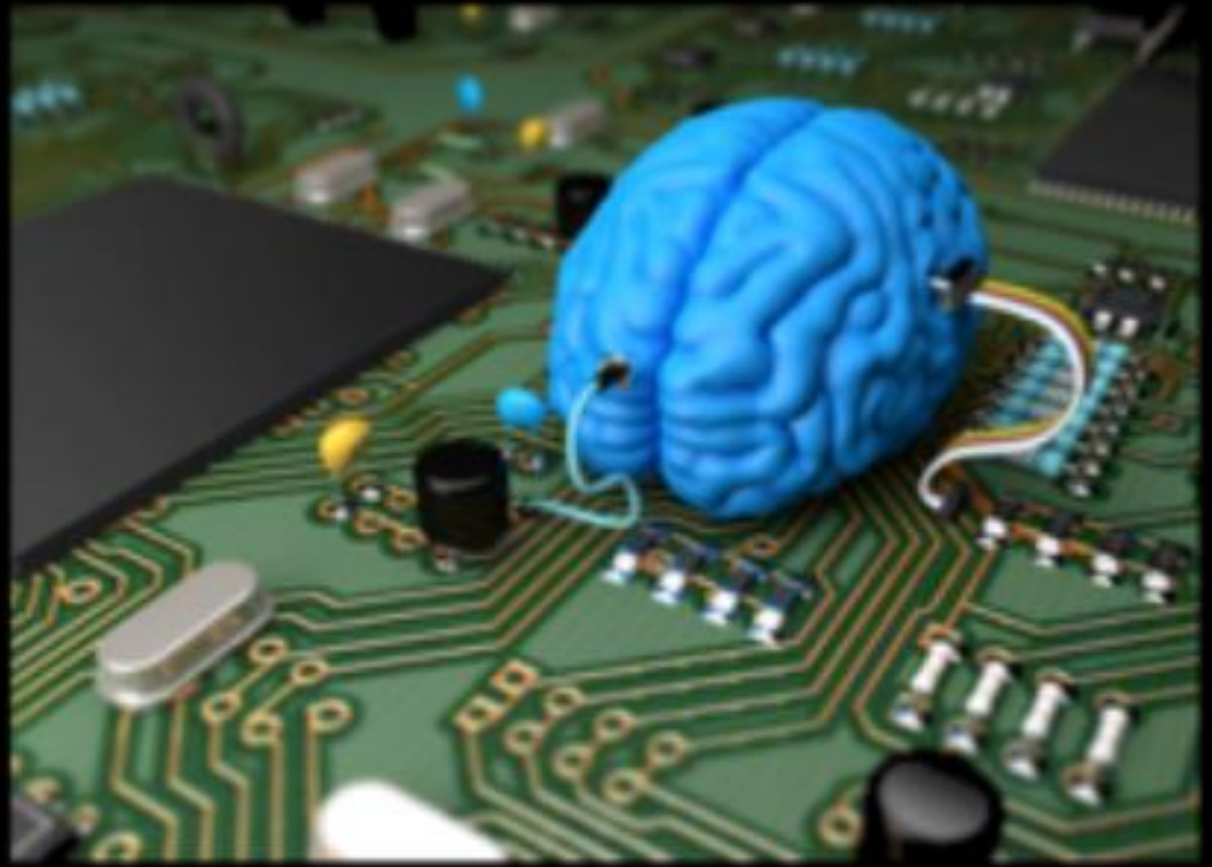


CS448V — Computational Video Manipulation

May 22th, 2019

Deep Learning

What society thinks I do



What my friends think I do

What other computer scientists think I do

What mathematicians think I do

What I think I do

```
In [1]:
import keras
Using TensorFlow backend.
```

What I actually do

Why? - Cool! Trendy! - Google Scholar

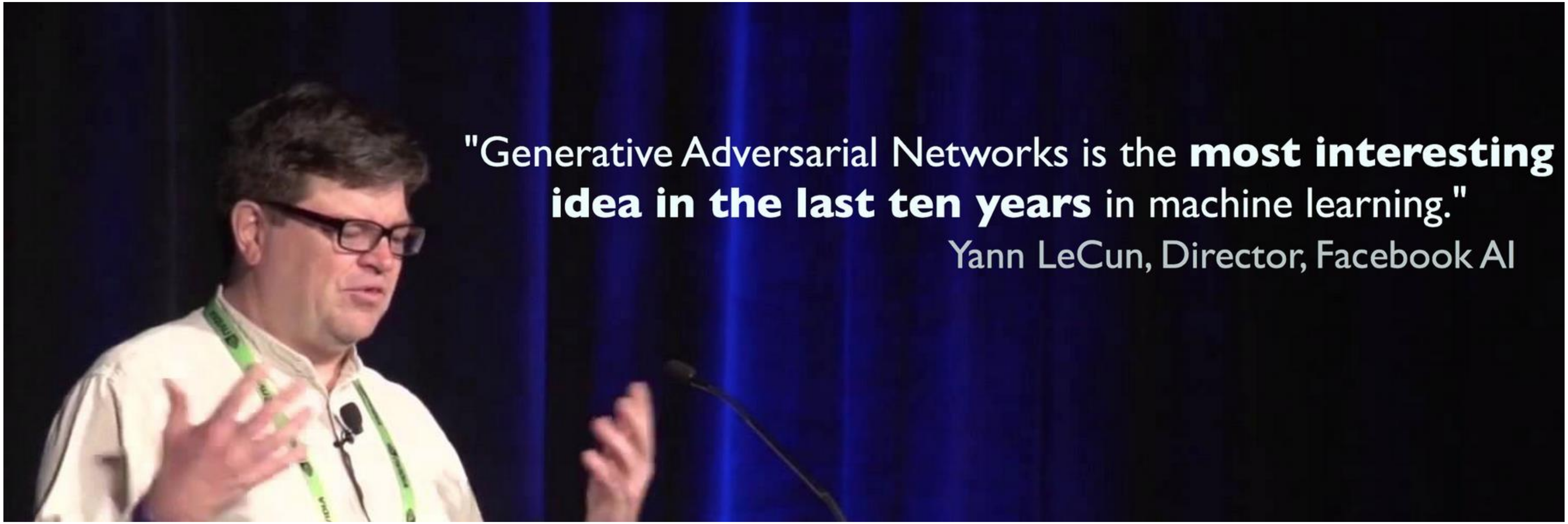
TITLE		CITED BY	YEAR
Generative adversarial nets I Goodfellow, J Pouget-Abadie, M Mirza, B Xu, D Warde-Farley, S Ozair, ... Advances in neural information processing systems, 2672-2680		8405	2014
Image-to-image translation with conditional adversarial networks P Isola, JY Zhu, T Zhou, AA Efros Proceedings of the IEEE conference on computer vision and pattern …	**Pix2Pix**	2137	2017
Unpaired image-to-image translation using cycle-consistent adversarial networks JY Zhu, T Park, P Isola, AA Efros Proceedings of the IEEE international conference on computer vision, 2223-2232	**CycleGAN**	1722	2017

⋮

Hundreds of applications
and follow-up works

⋮

Why? - Cool! Trendy! - Google Scholar

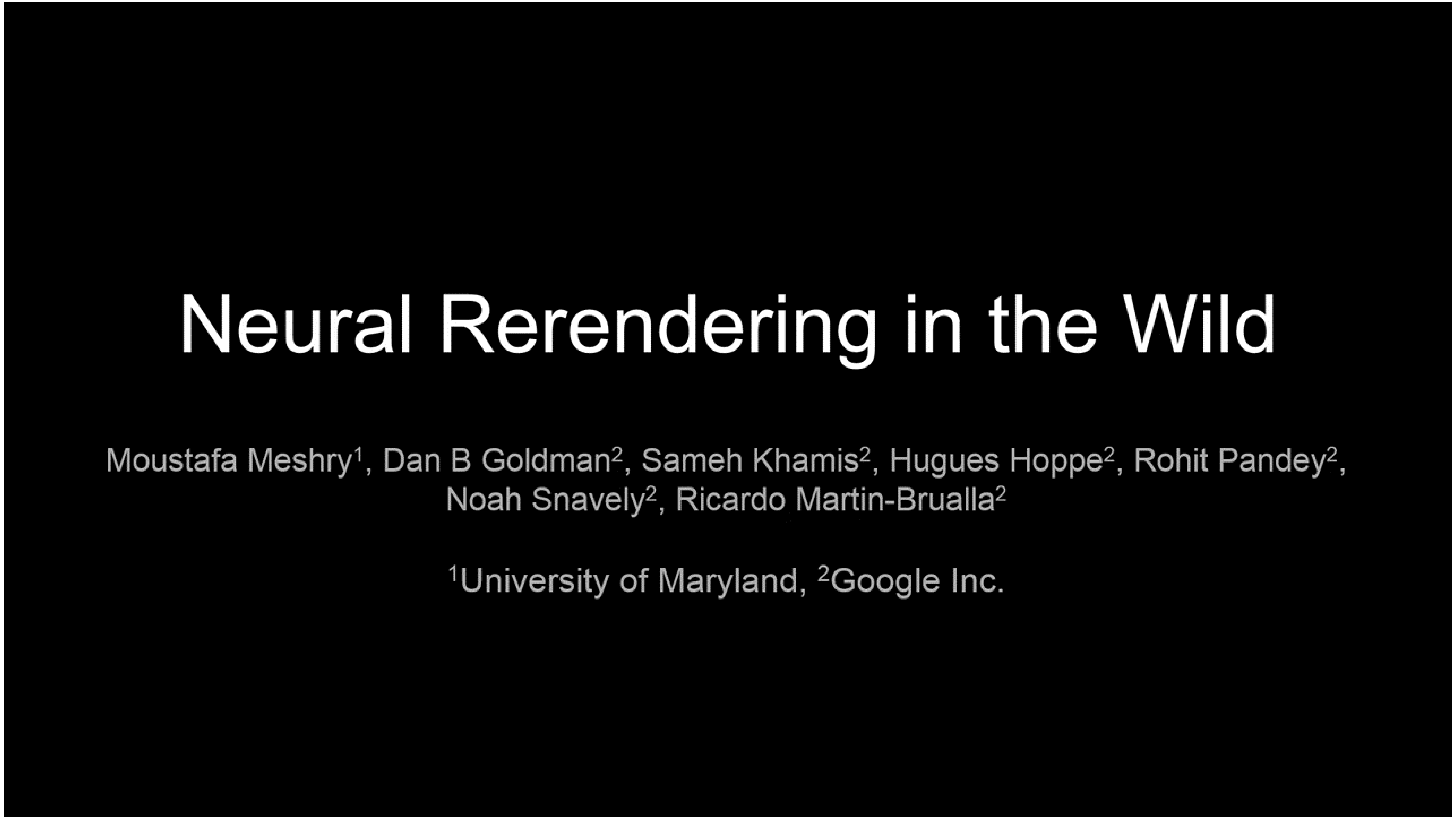


⋮

Hundreds of applications
and follow-up works

⋮

Enhancing Transitions



Single-Photo Facial Animation

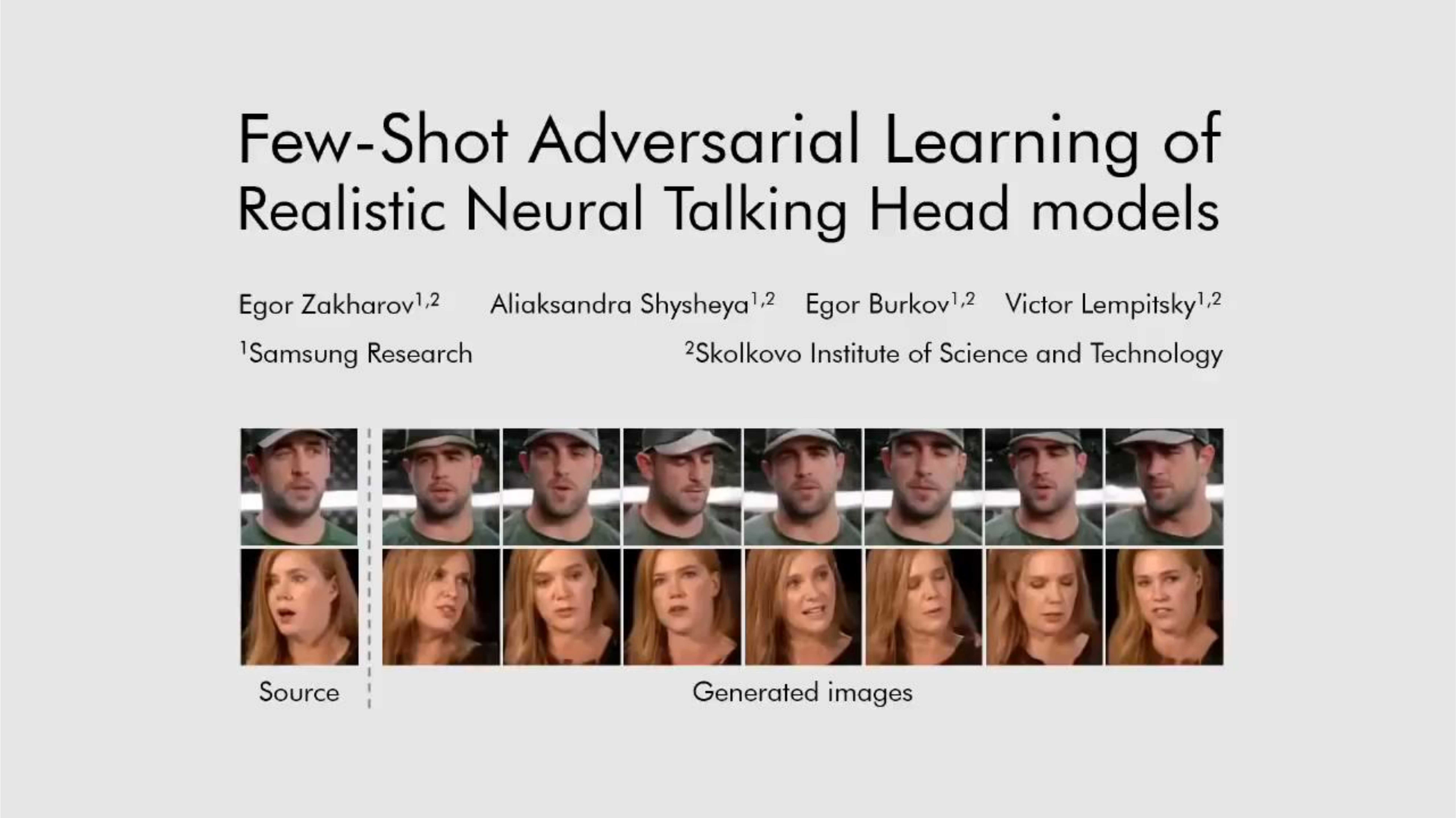
Text-based Editing

Adding New Words

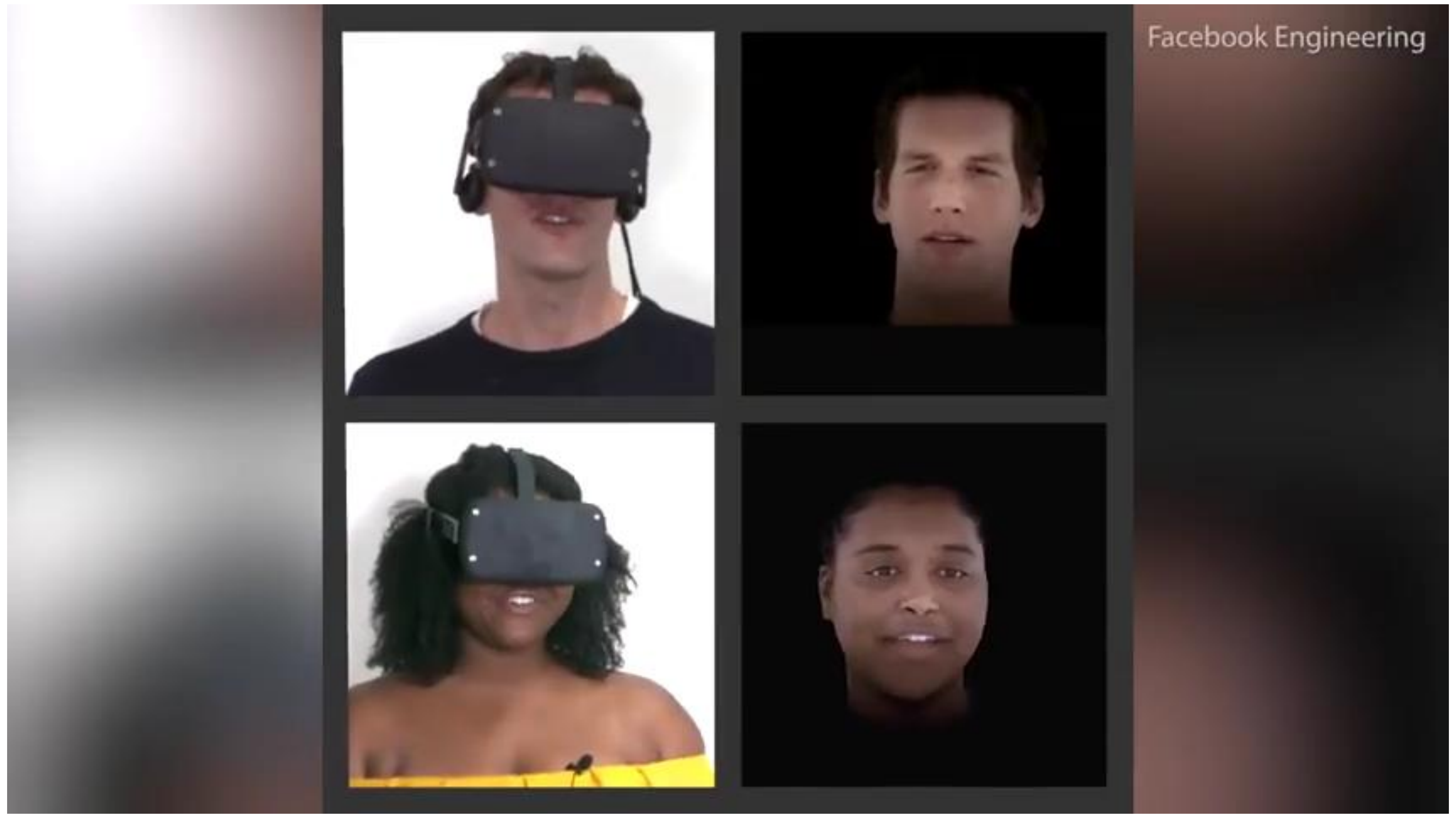
Original Video

I love the smell of napalm in the morning.

Few-Shot Reenactment



Digital Humans



facebook
Reality Labs

Overview

- Convolutional Neural Networks

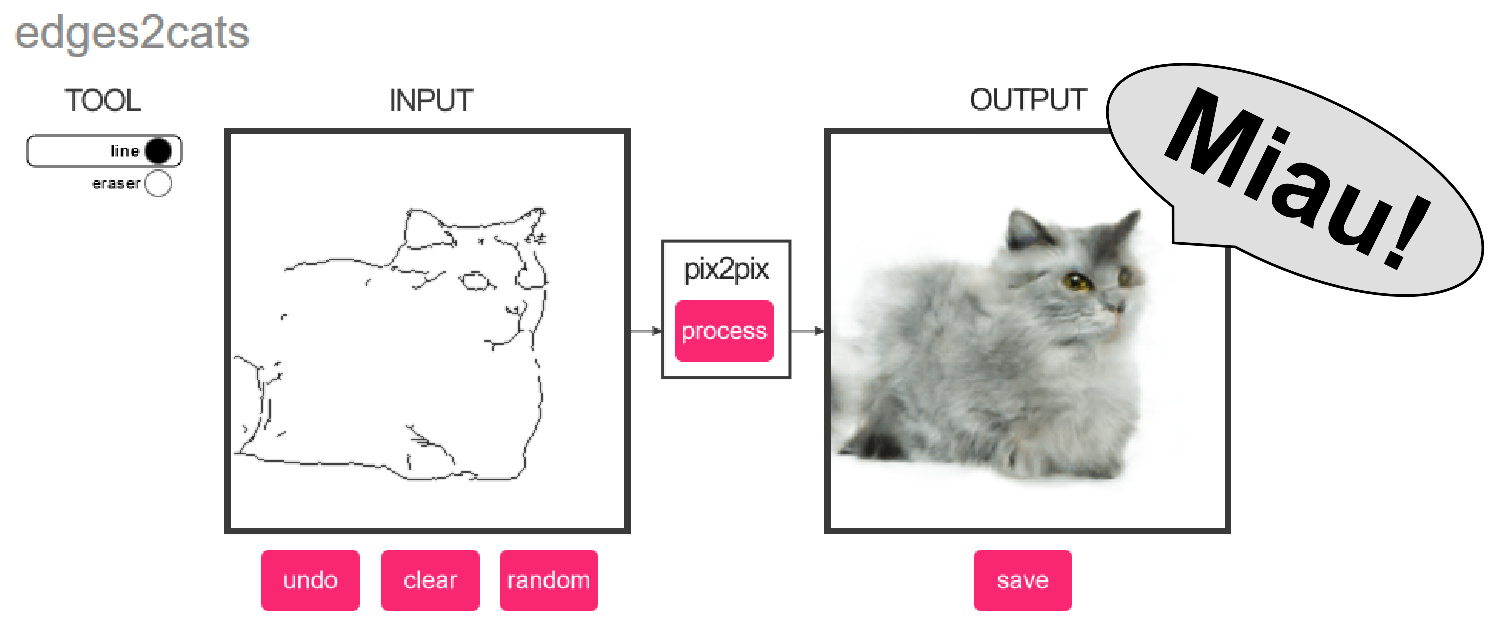
$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\, d\tau.$$

- Generative Modeling

2014 2015 2016 2017

(Brundage et al, 2018)

- Pix2Pix (“mapping from A to B”)



Convolutional Neural Network

Components?

- 2D Convolution Layers (Conv2D)
- Subsampling Layers (MaxPool, …)
- Non-linearity Layers (ReLU, …)
- Normalization Layers (BatchNorm, …)
- Upsampling Layers (TransposedConv, …)
- …

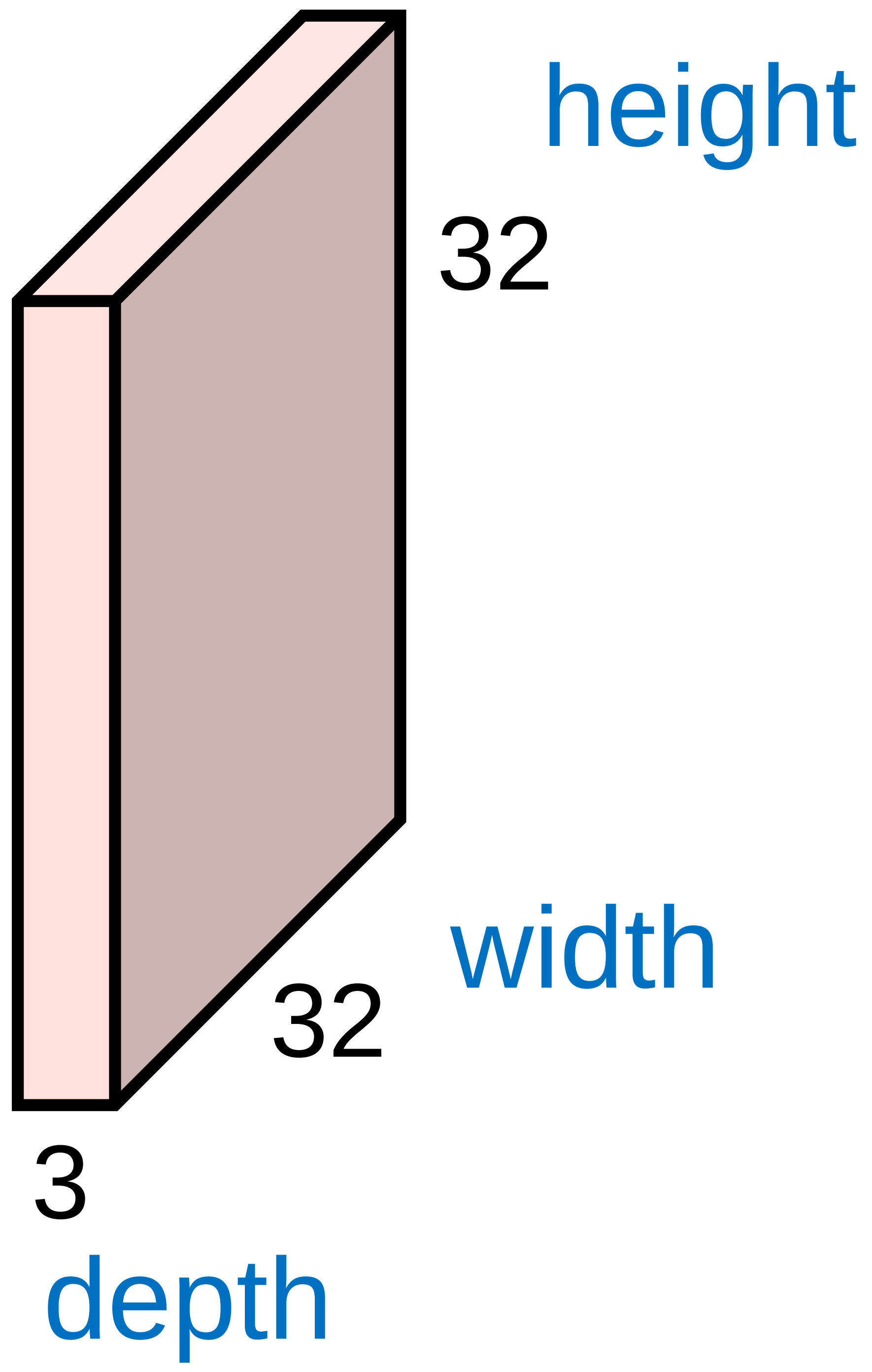
Convolutional Neural Network

Components?

- 2D Convolution Layers (Conv2D)
- Subsampling Layers (MaxPool, …)
- Non-linearity Layers (ReLU, …)
- Normalization Layers (BatchNorm, …)
- Upsampling Layers (TransposedConv, …)
- …

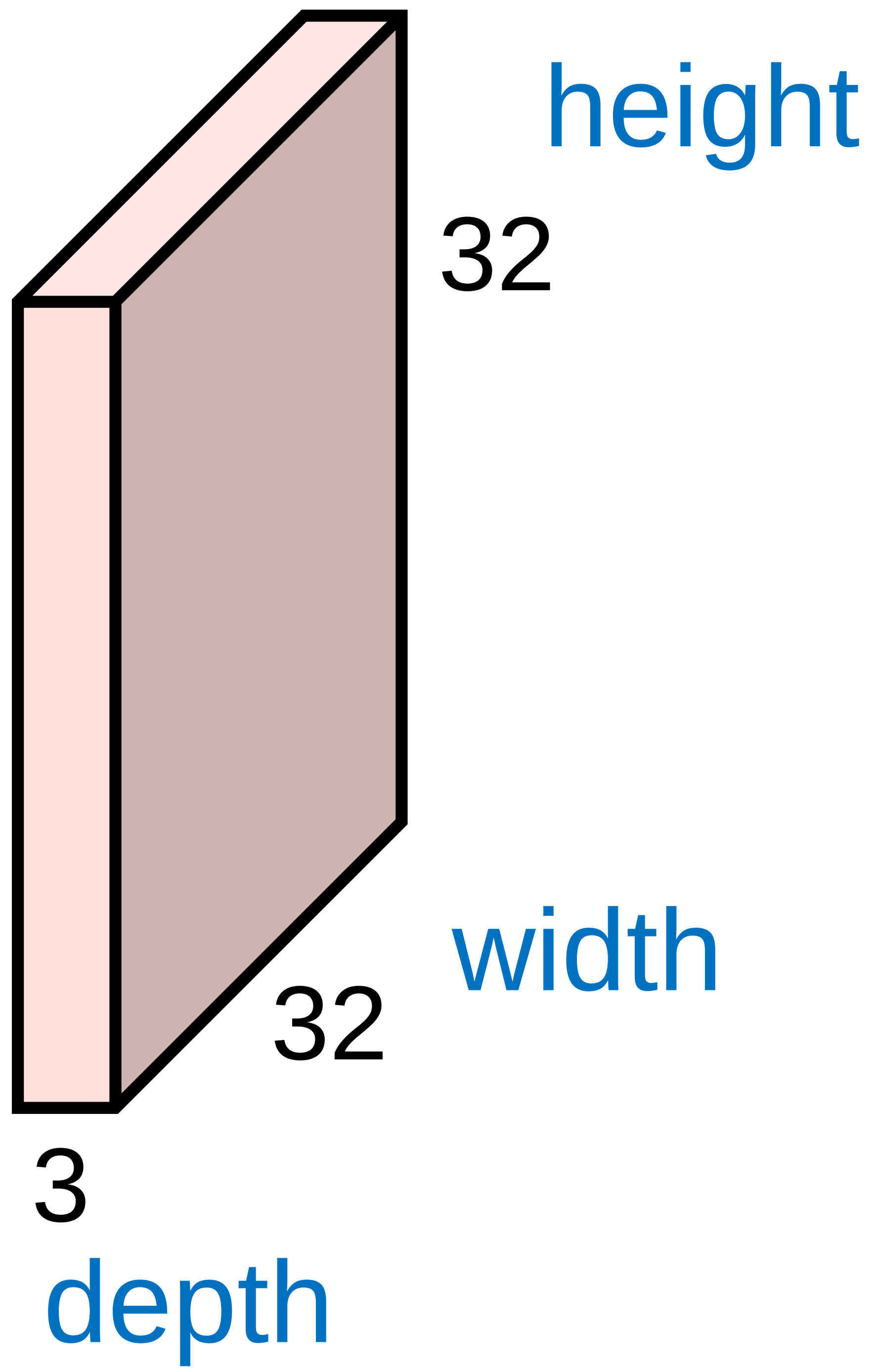
Convolution

32x32x3 image

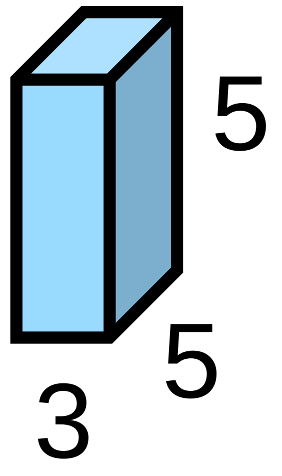


Convolution

32x32x3 image



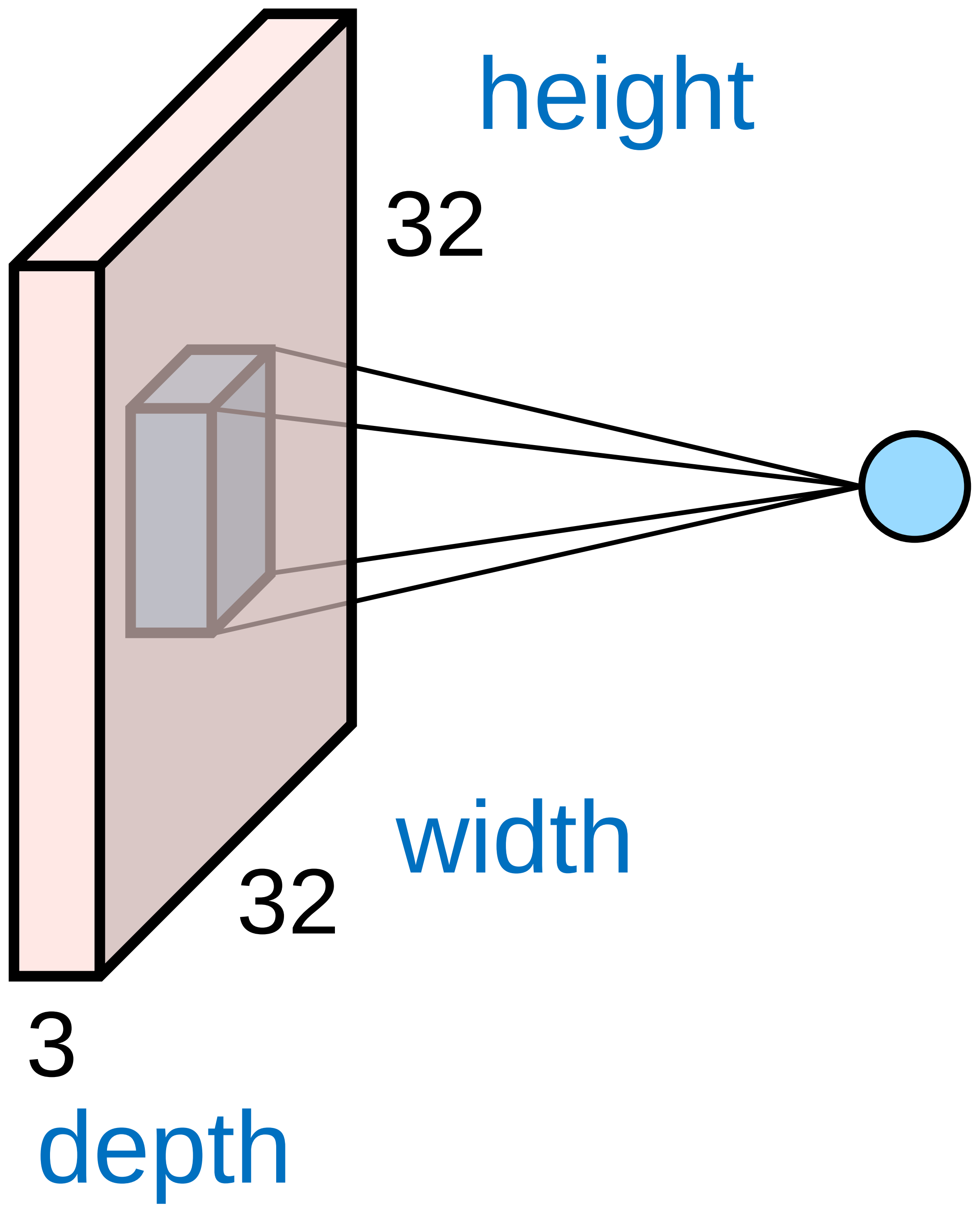
5x5x3 filter



Convolve the filter with the image, i.e., “slide over the image spatially, computing dot products”

Convolution

32x32x3 image

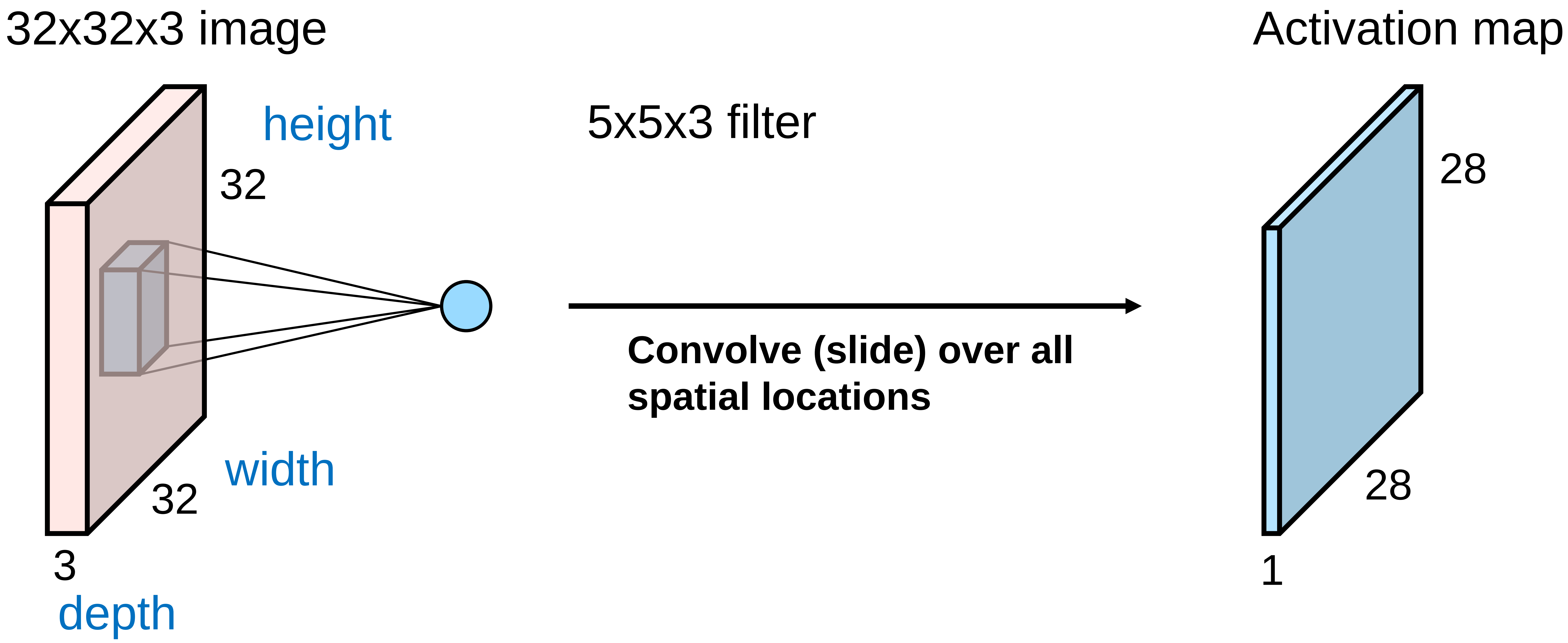


5x5x3 filter

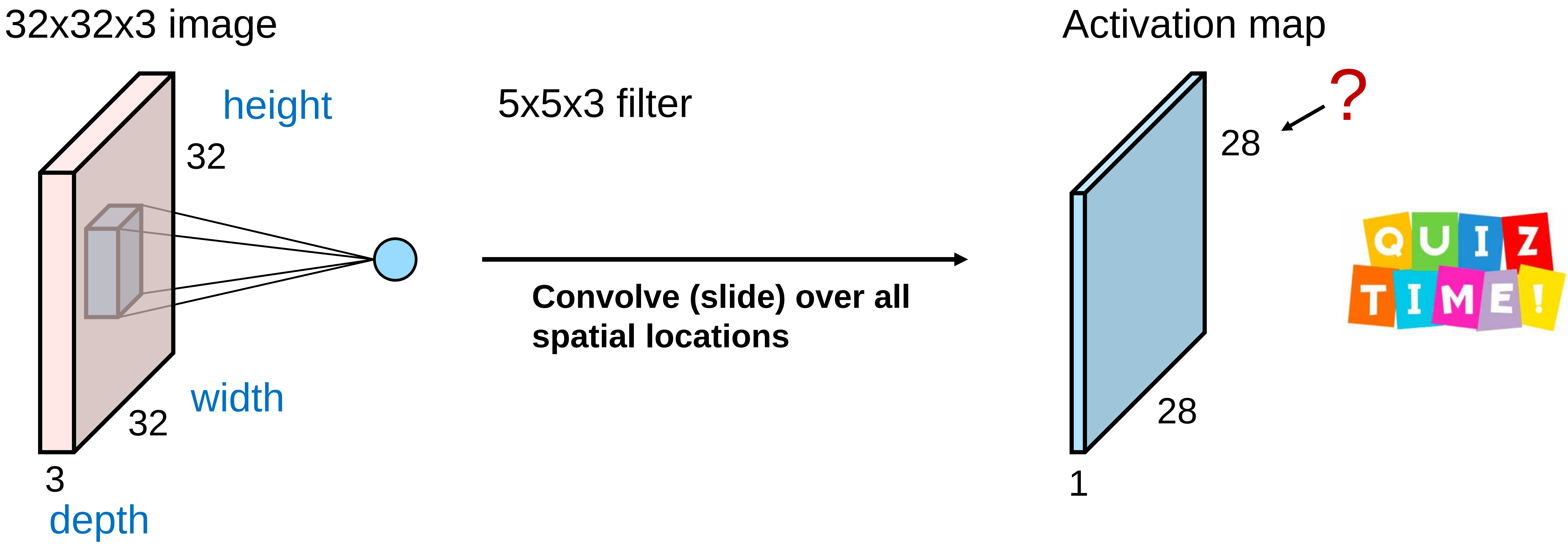
Result: 1 number, the result of taking the dot product between the filter and a small 5x5x3 chunk of the image, i.e., 5x5x3 = 70-dimensional dot product + bias

$$w^T x + b$$

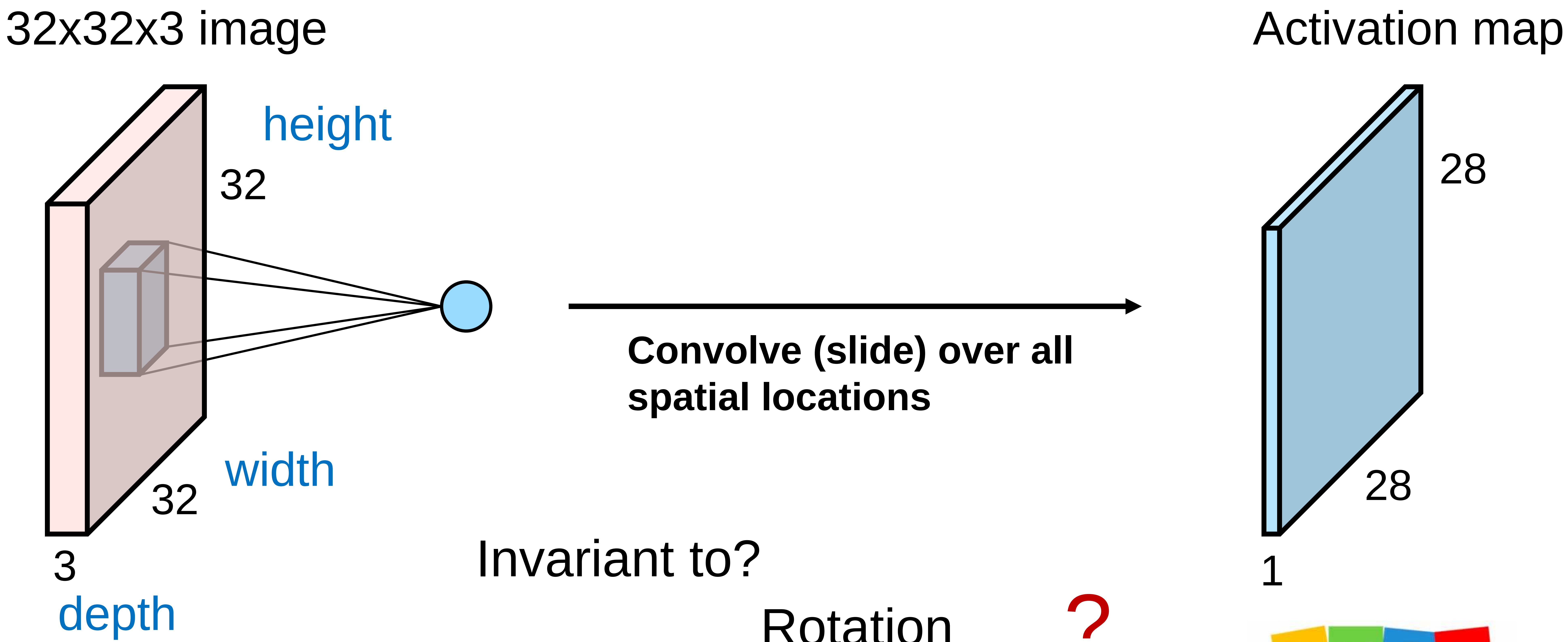
Convolution



Convolution



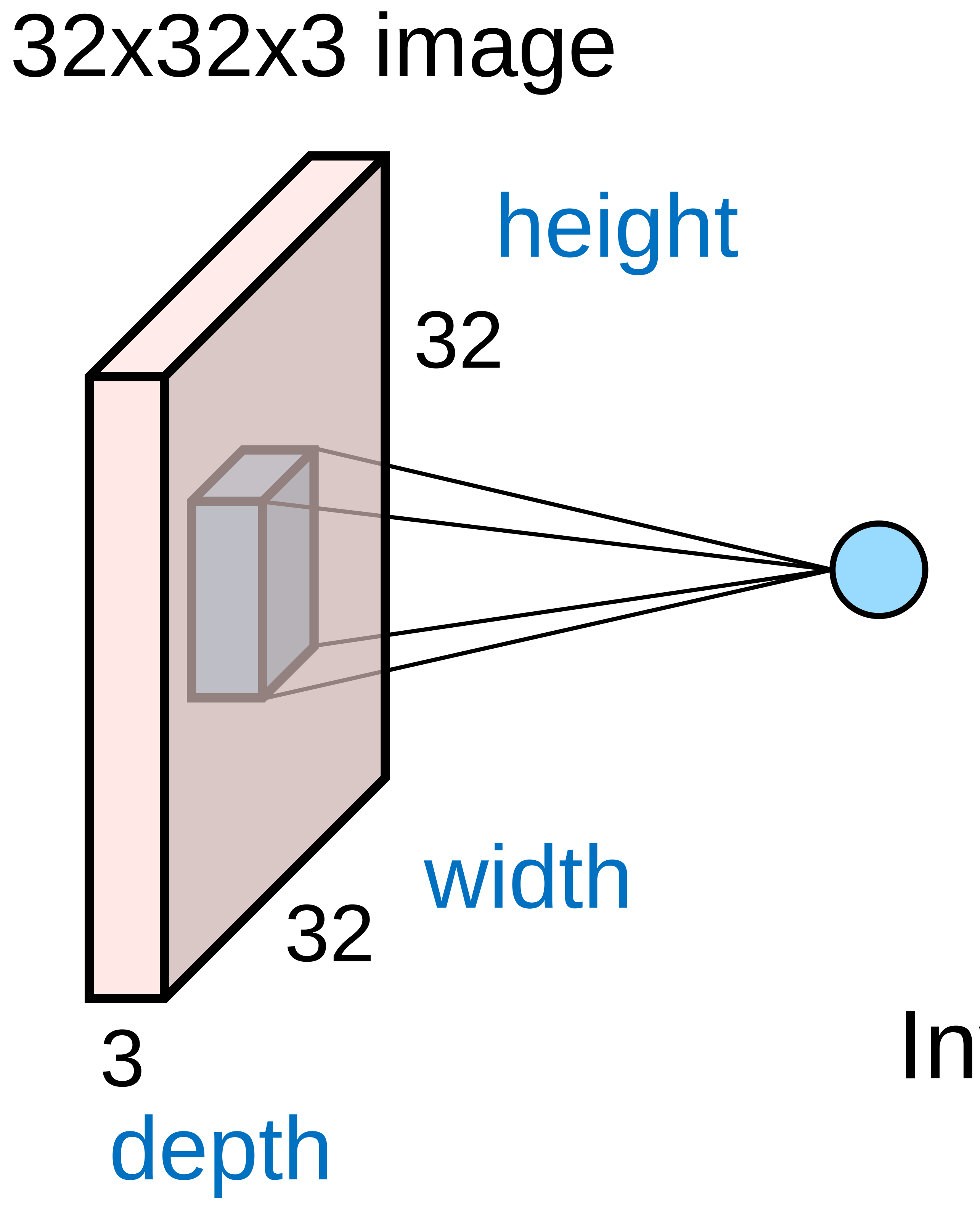
Convolution



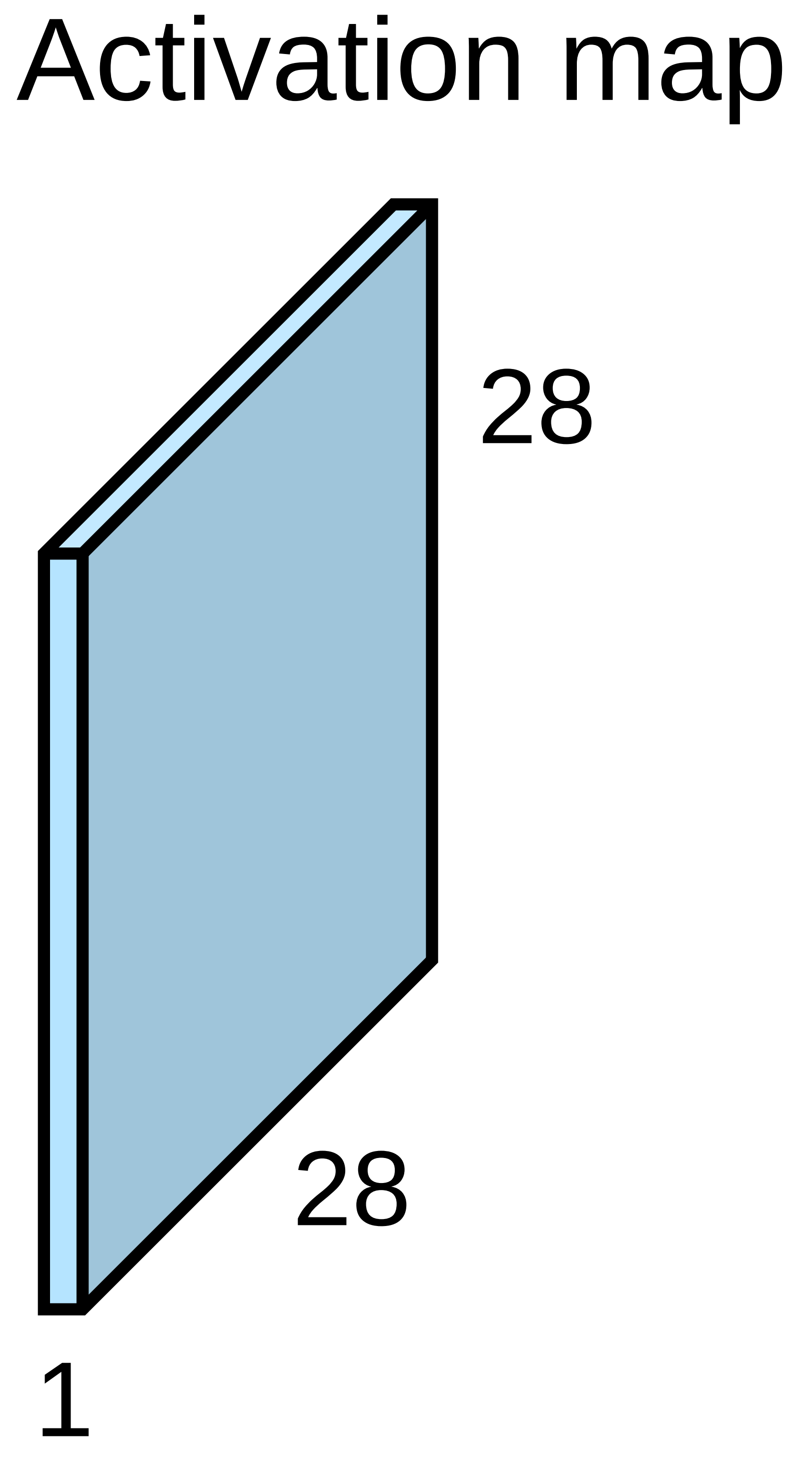
Invariant to?

Rotation ?
Translation ?
Scaling ?

Convolution



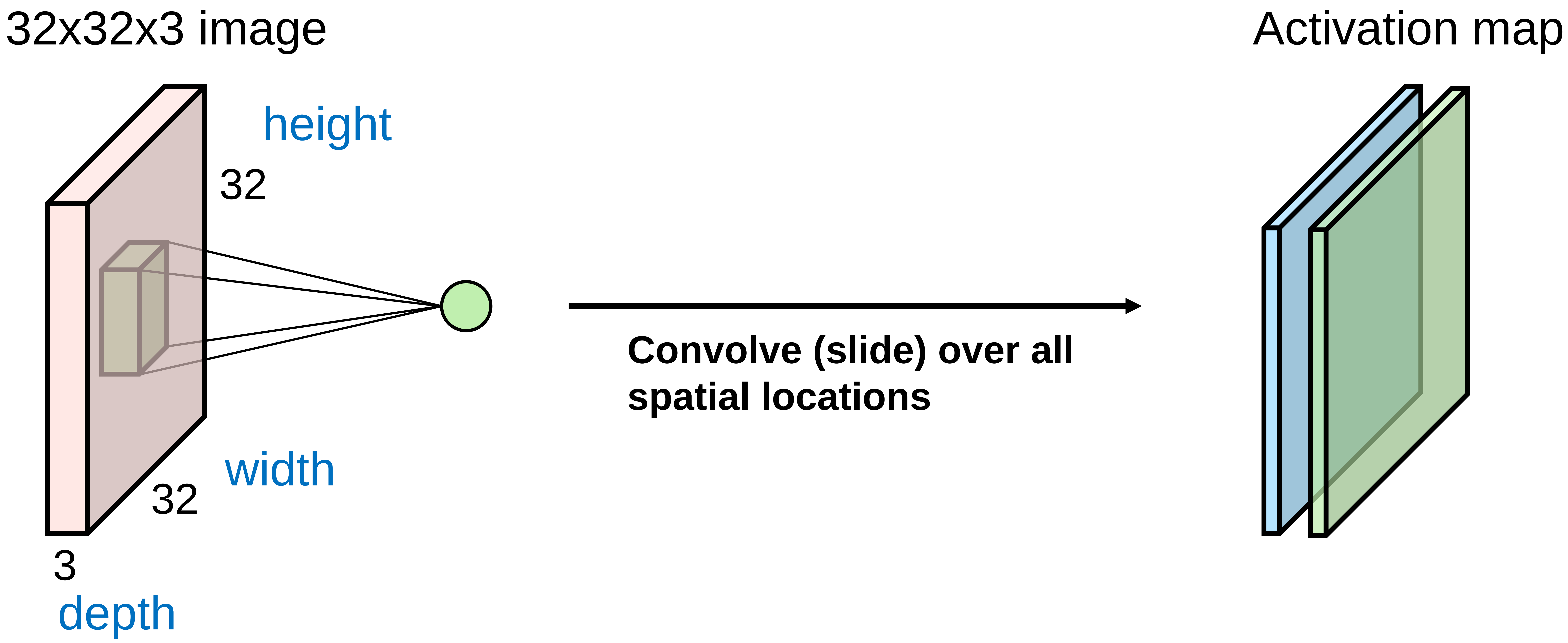
Convolve (slide) over all spatial locations



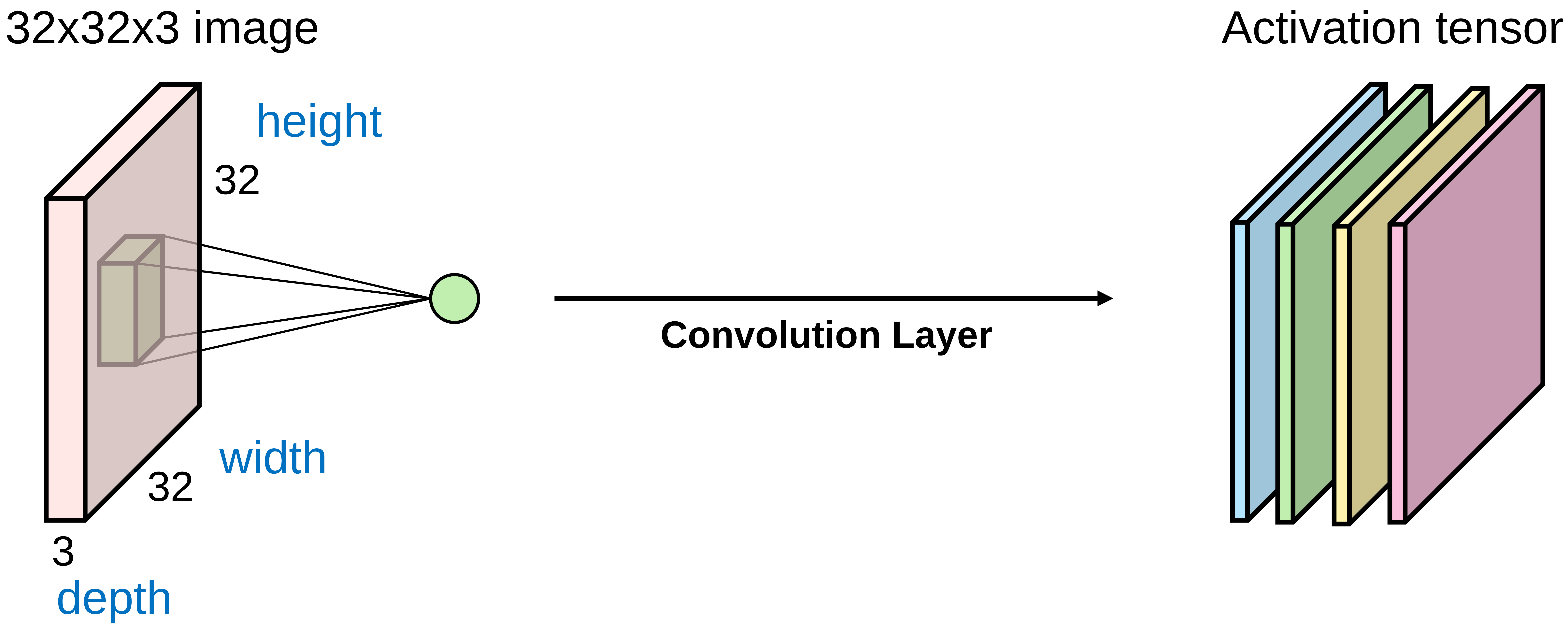
Invariant to?

Rotation	✗
Translation	✓
Scaling	✗

Convolution

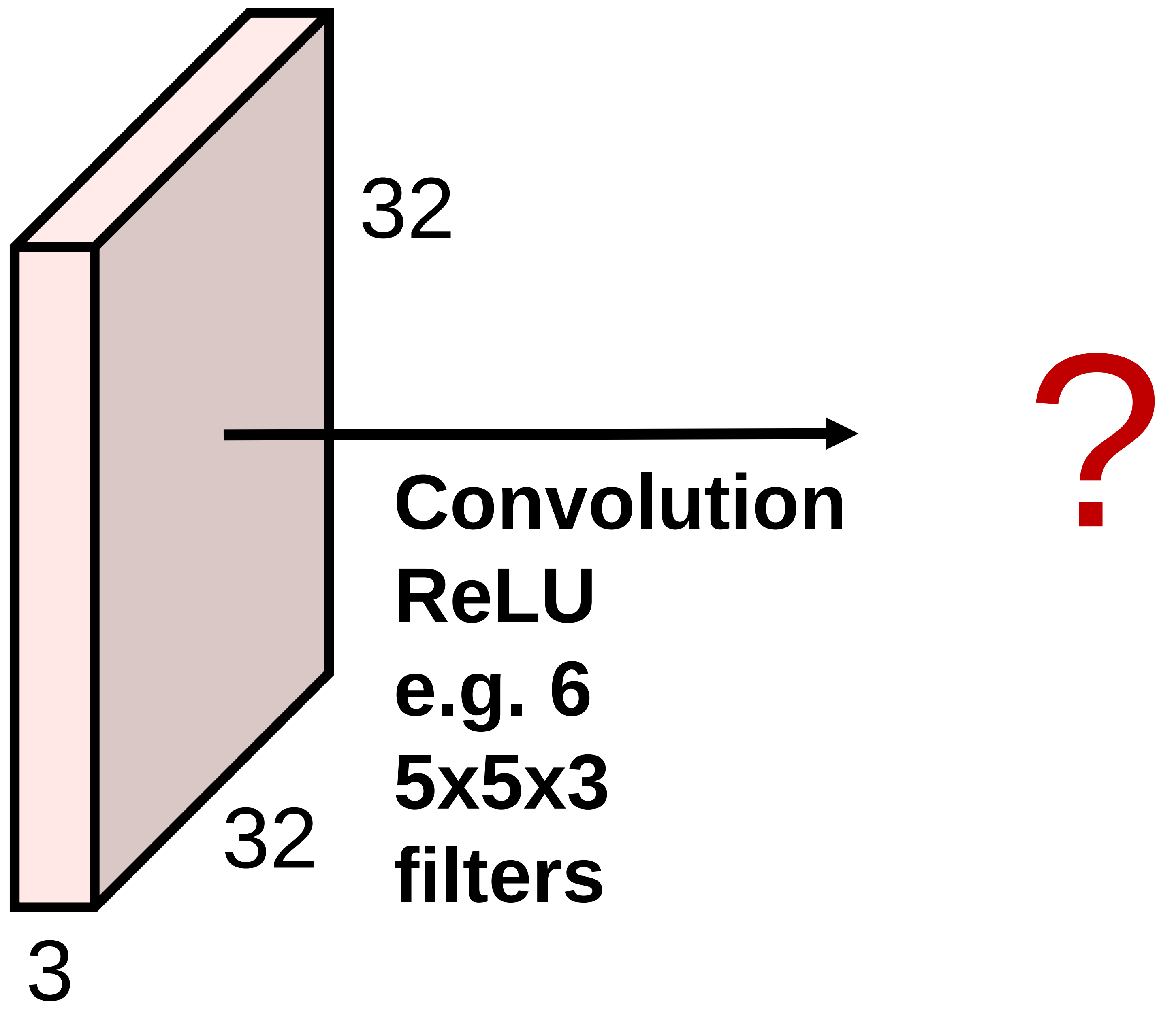


Convolution Layer

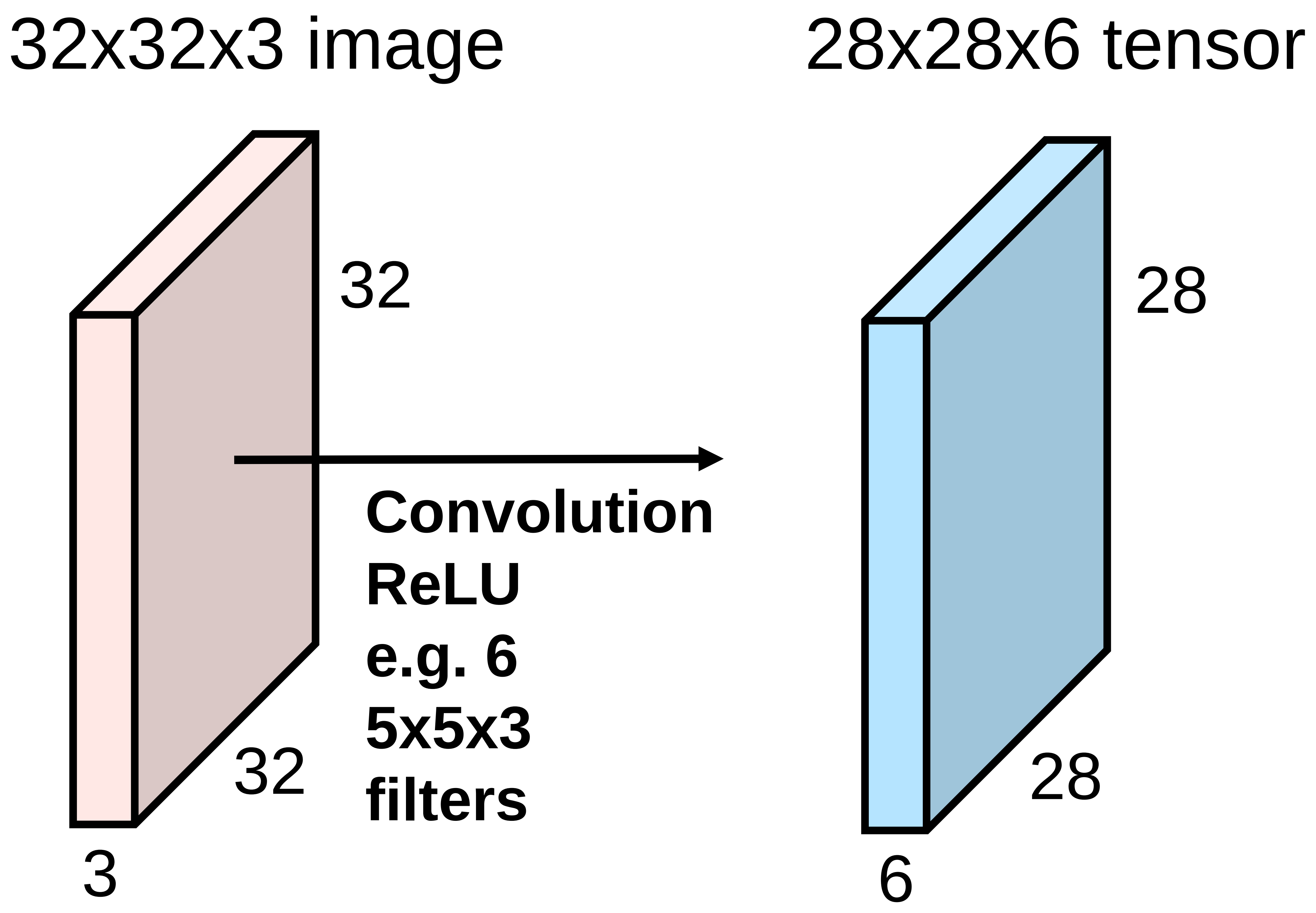


Convolutional Neural Network

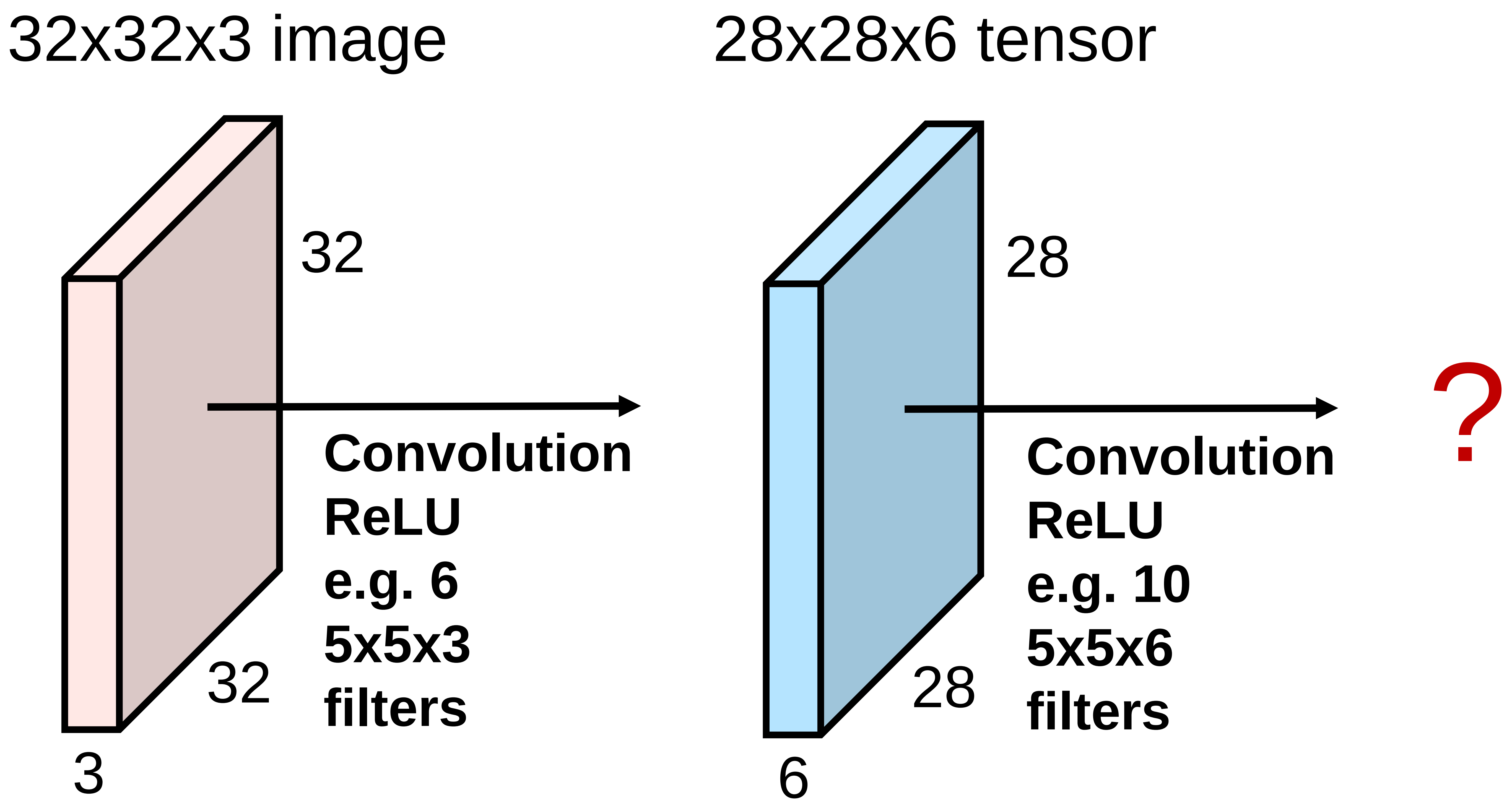
32x32x3 image



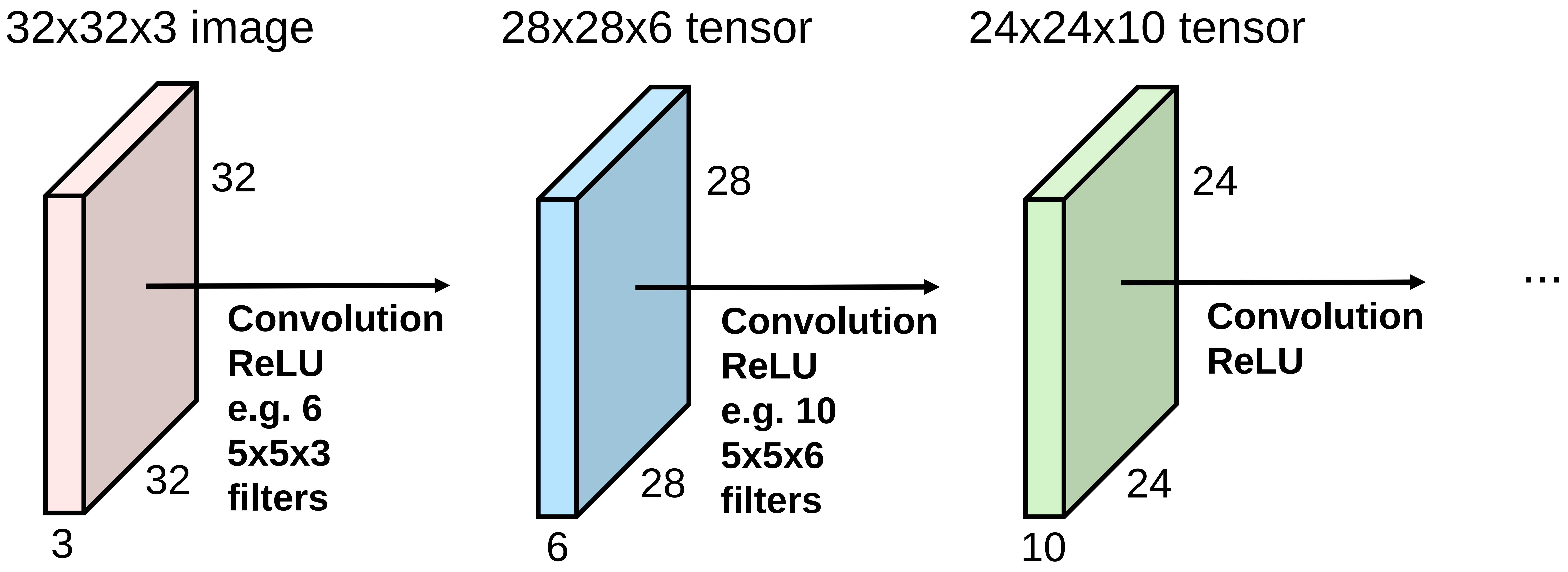
Convolutional Neural Network



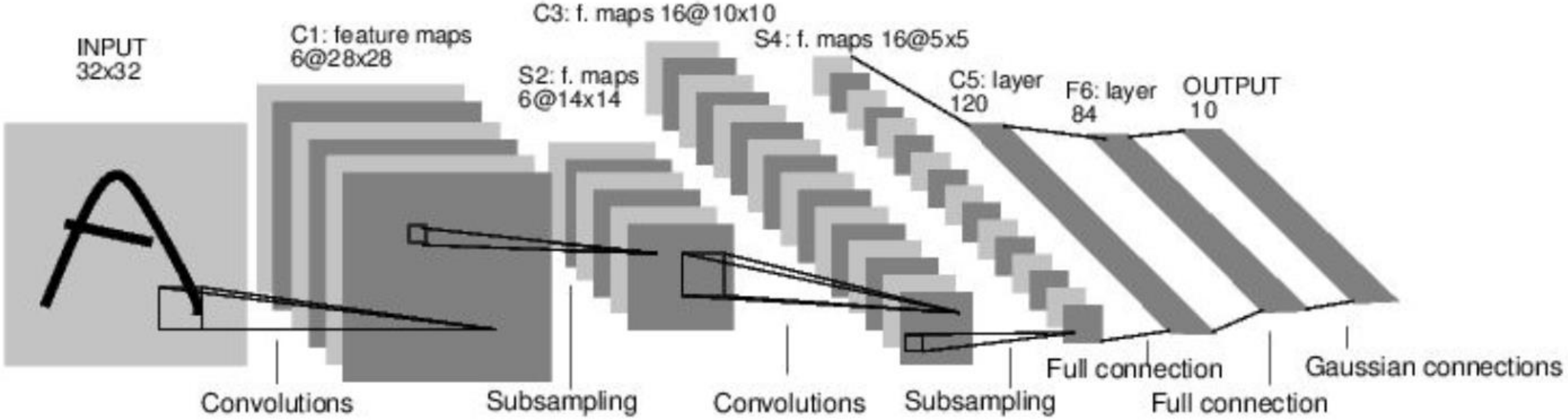
Convolutional Neural Network



Convolutional Neural Network



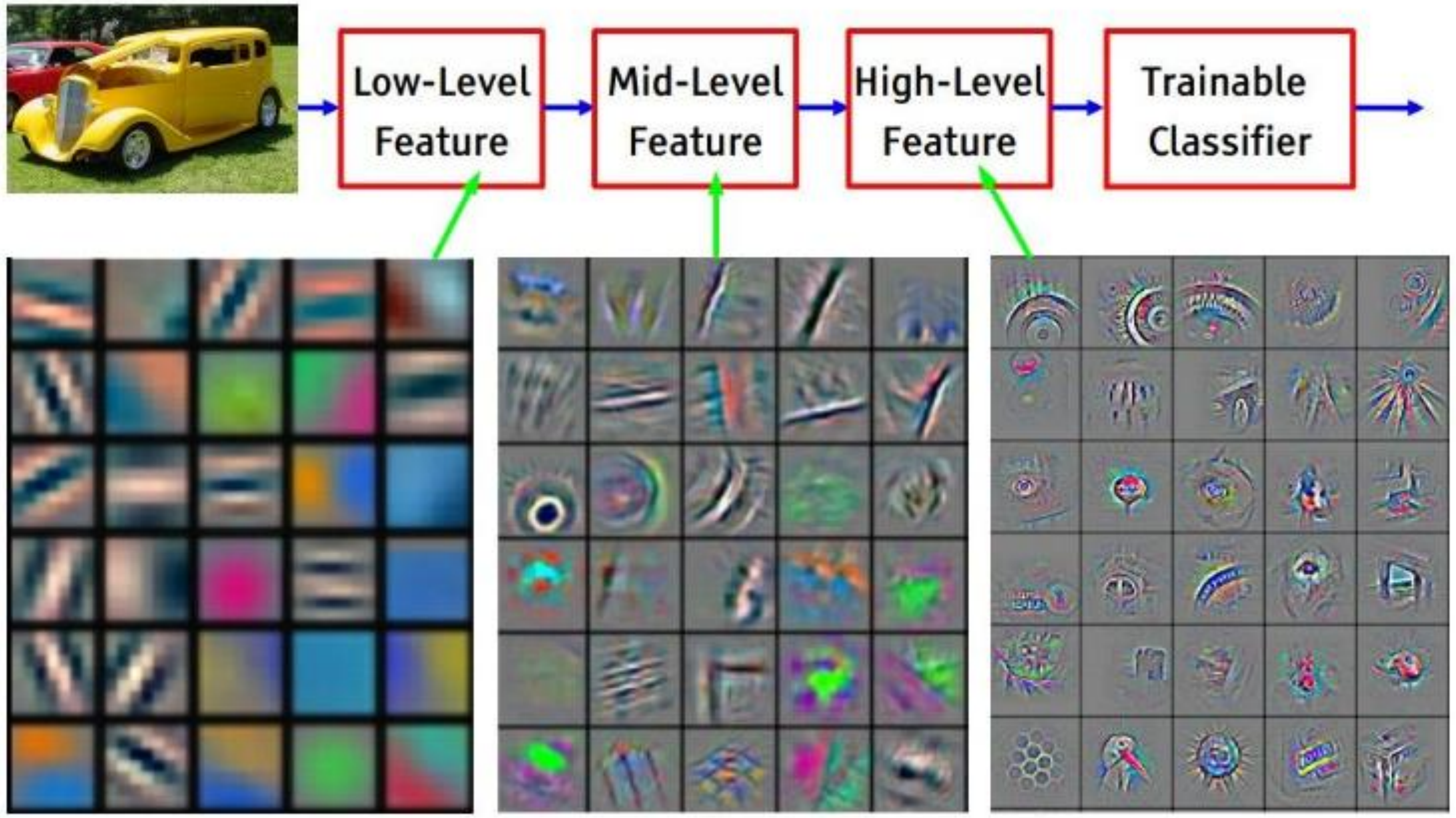
Convolutional Neural Networks



[LeNet-5, LeCun 1980]

Feature Hierarchy

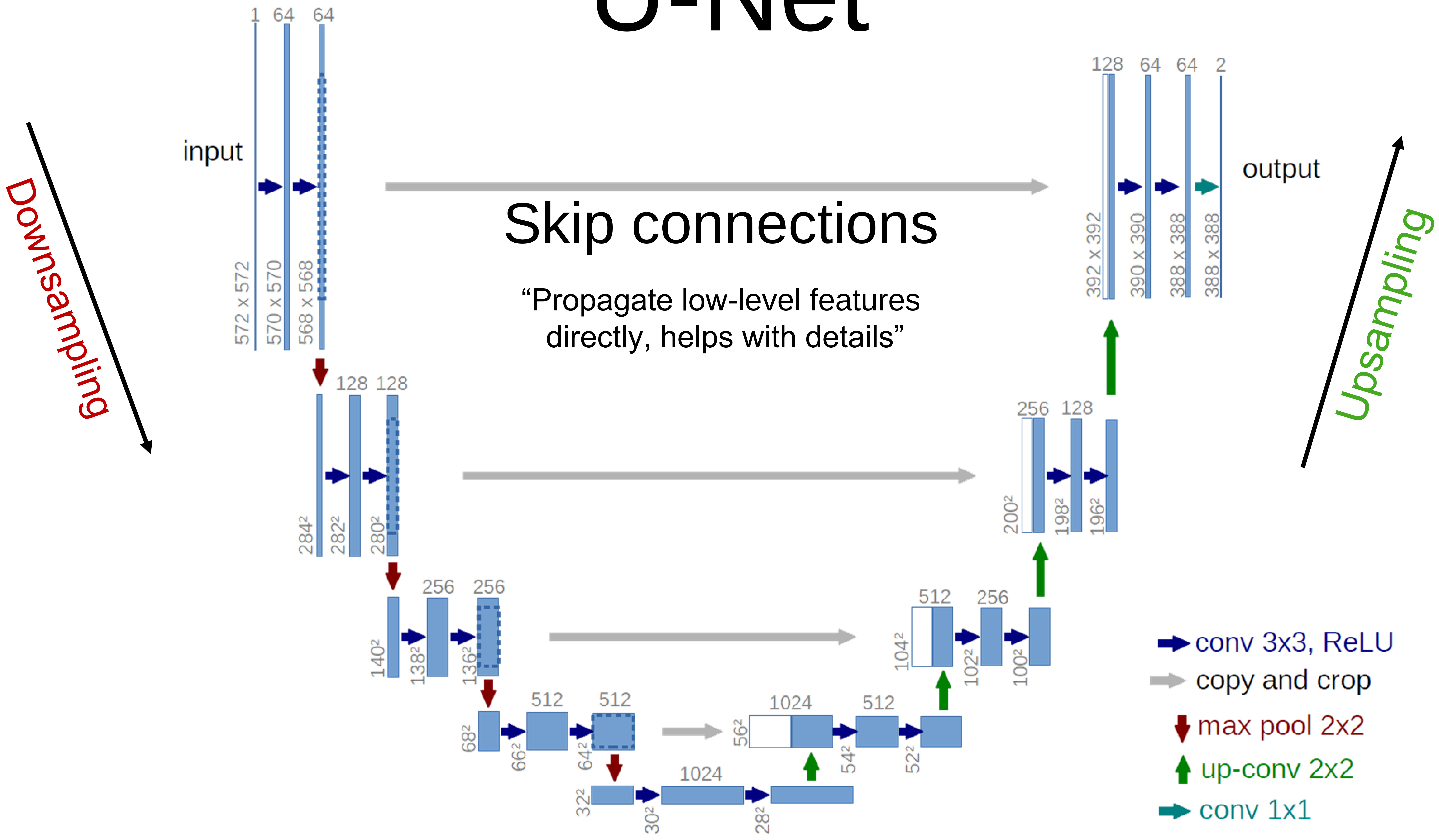
[From recent Yann LeCun slides]



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Learn the features from data instead of hand engineering them!
(If enough data is available)

U-Net



Skip connections

"Propagate low-level features directly, helps with details"

Overview

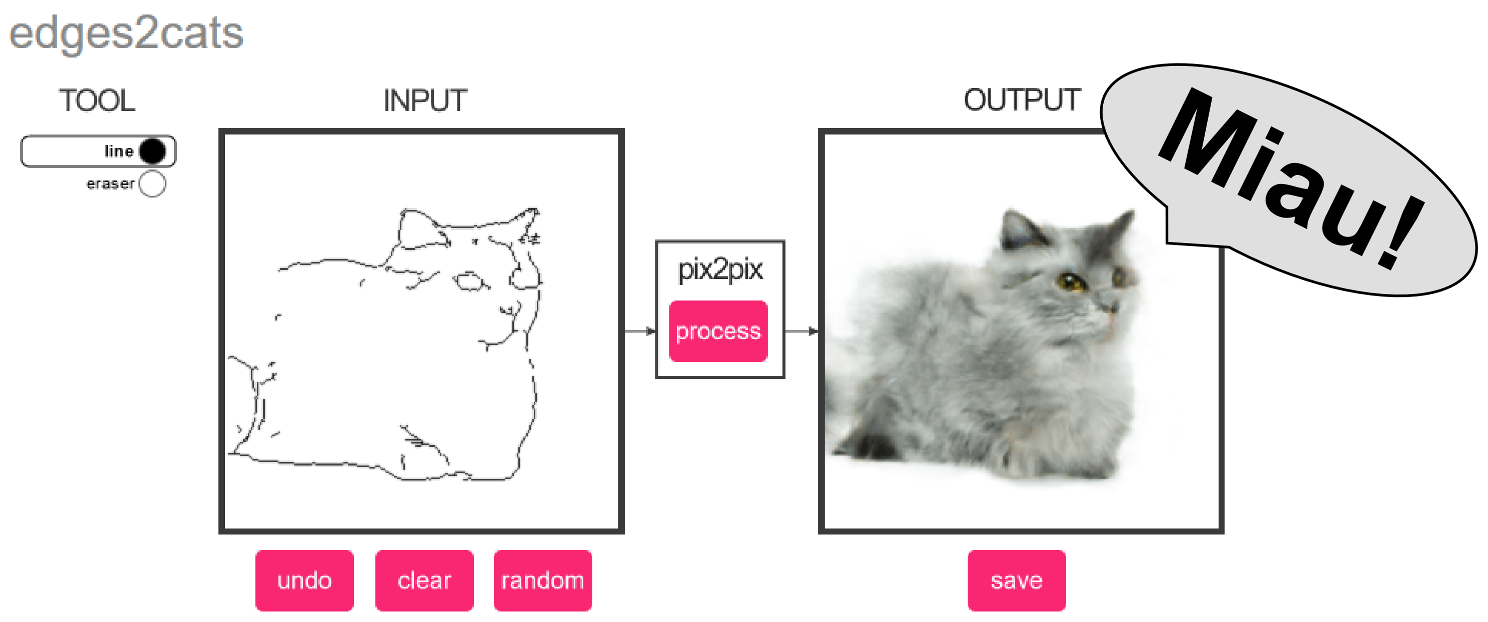
- Convolutional Neural Networks

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau) g(t-\tau)\, d\tau.$$

- Generative Modeling

(Brundage et al, 2018)

- Pix2Pix



Overview

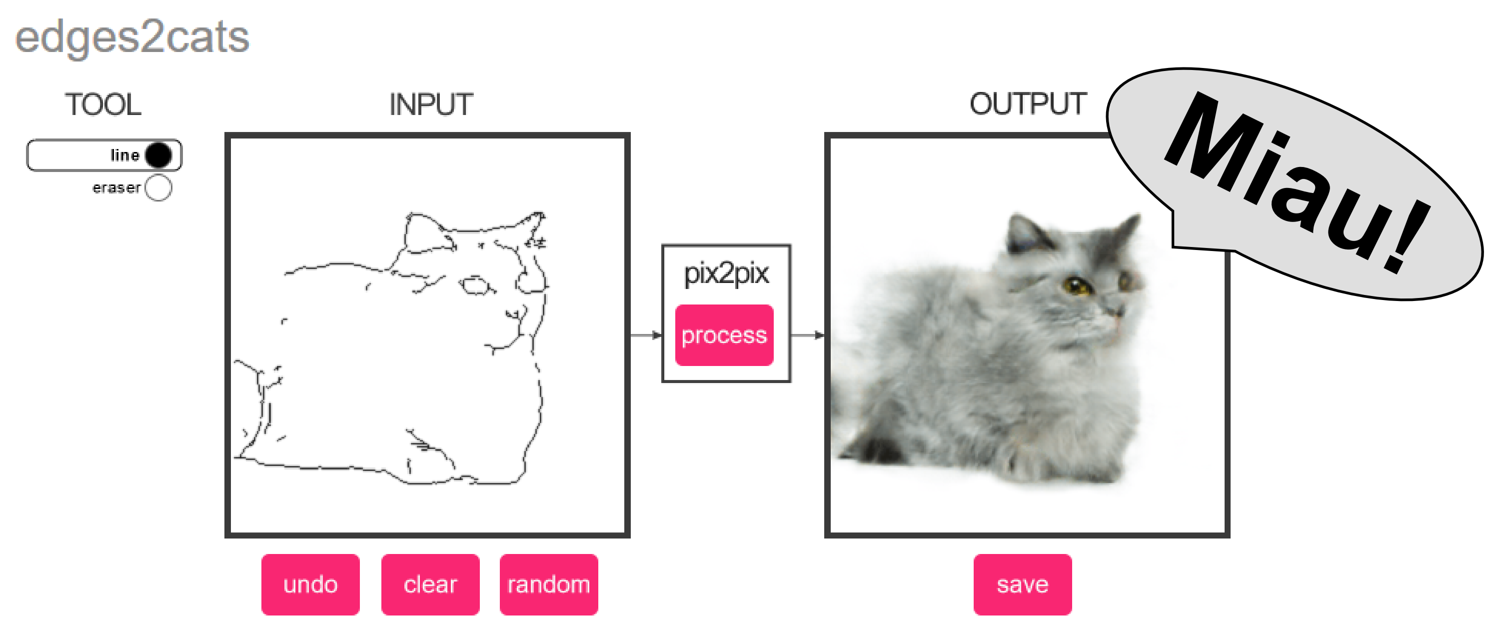
- Convolutional Neural Networks

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau.$$

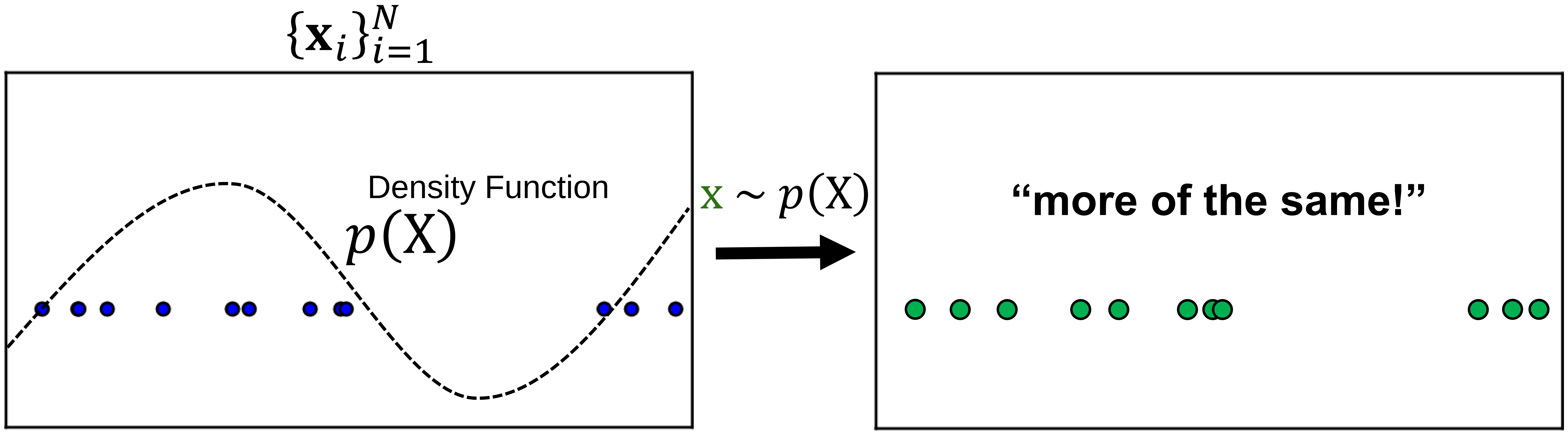
- Generative Modeling

(Brundage et al. 2018)

- Pix2Pix



Generative Modeling



Training Data

New Samples

We want to learn $p(\mathrm{X})$ from data, such that we can “sample from it”!

Generative 2D Face Modeling

$\{\mathbf{x}_i\}_{i=1}^{N}$

$\mathrm{x} \sim p(\mathrm{X})$

Training Data

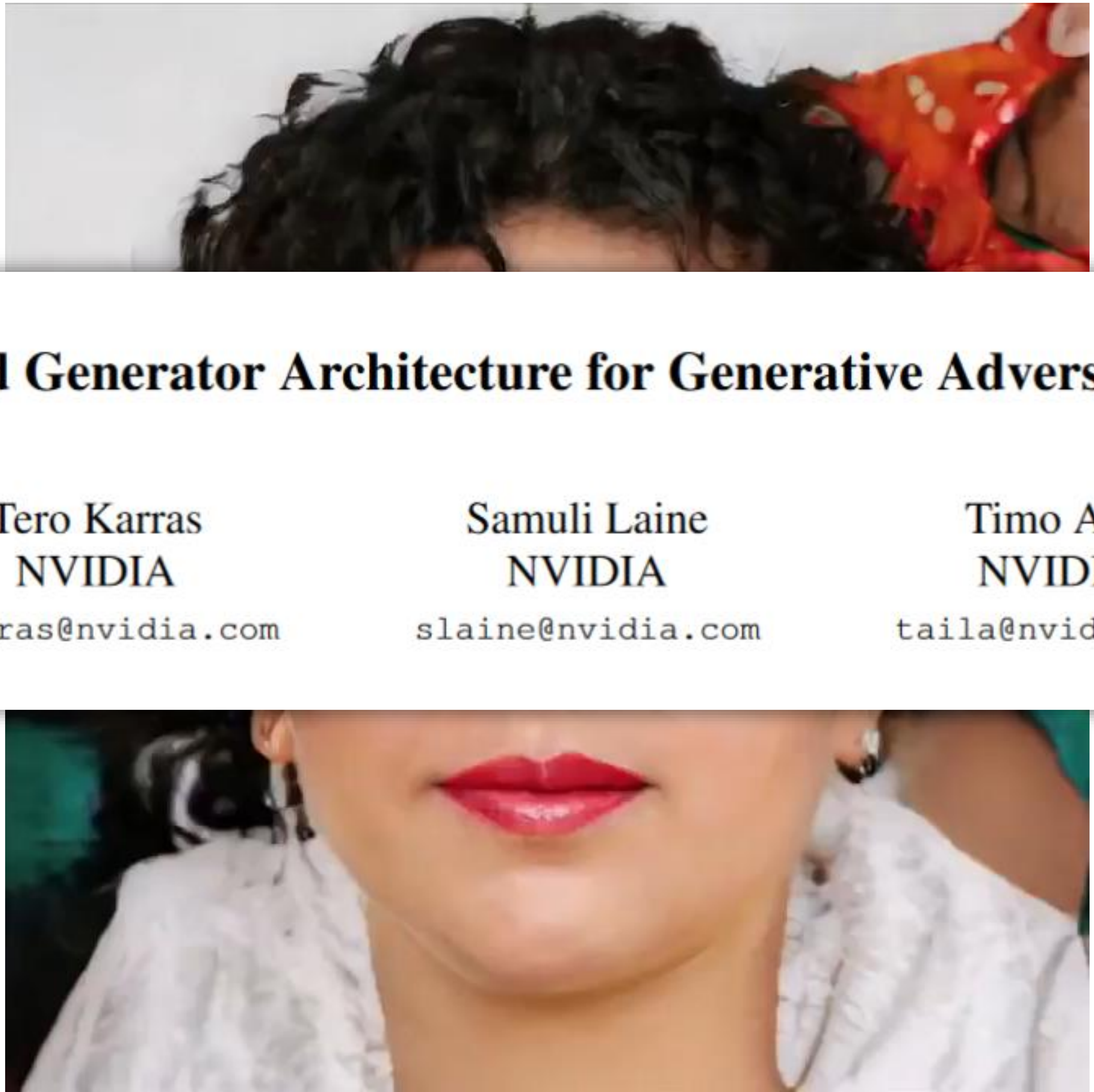
New Samples

The world needs more celebrities
… or not … ?

3.5 Years of Progress on Faces

(Brundage et al, 2018)

https://thispersondoesnotexist.com



A Style-Based Generator Architecture for Generative Adversarial Networks

Tero Karras
NVIDIA
tkarras@nvidia.com

Samuli Laine
NVIDIA
slaine@nvidia.com

Timo Aila
NVIDIA
taila@nvidia.com

2018

StyleGAN - Interpolation

Overview

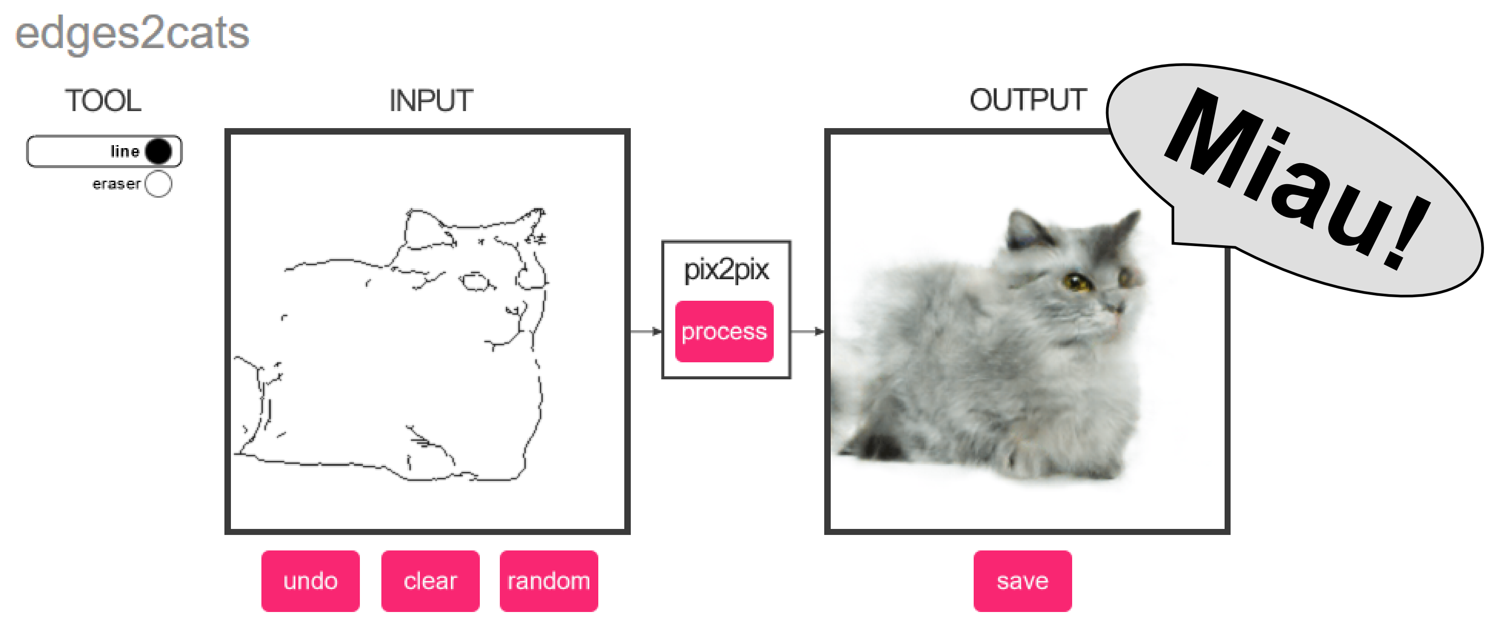
- Convolutional Neural Networks

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau.$$

- Generative Modeling

(Brundage et al, 2018)

- Pix2Pix ("mapping from A to B")



Overview

- Convolutional Neural Networks

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau.$$

- Generative Modeling

(Brundage et al. 2018)

- Pix2Pix ("mapping from A to B")

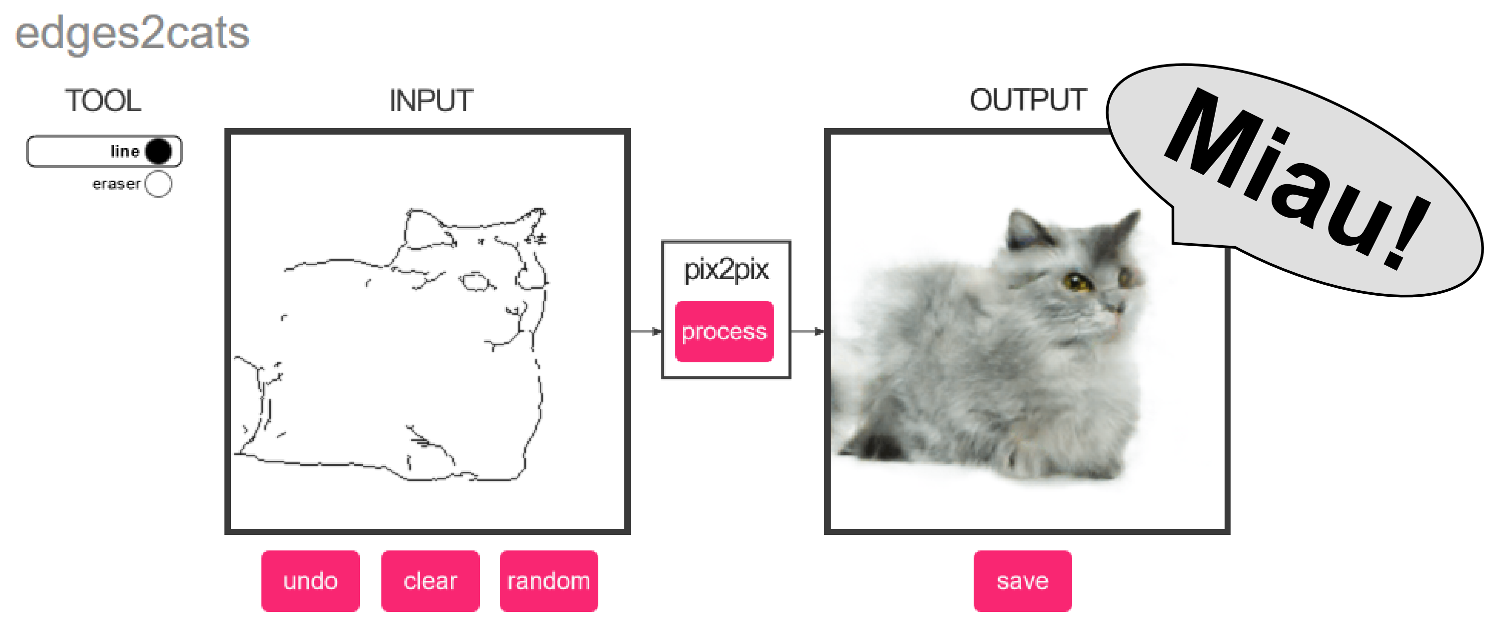


Image-to-Image Translation with Conditional Adversarial Networks

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

Berkeley AI Research (BAIR) Laboratory, UC Berkeley

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

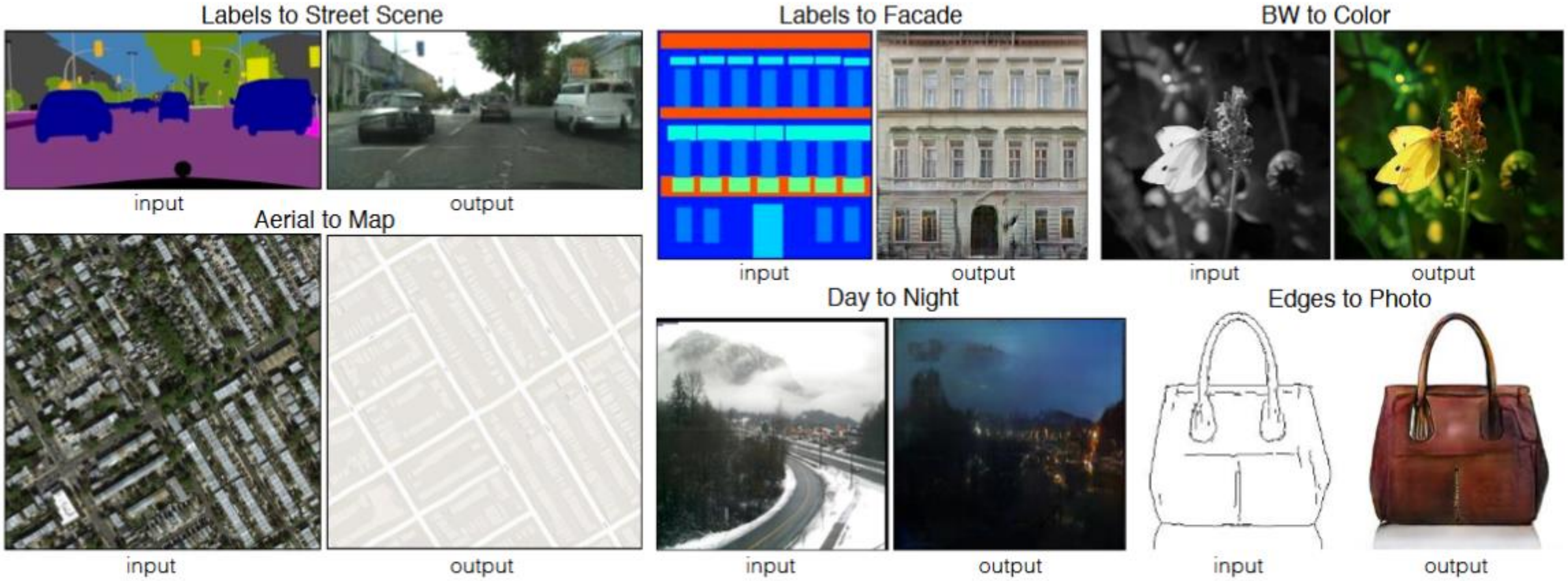
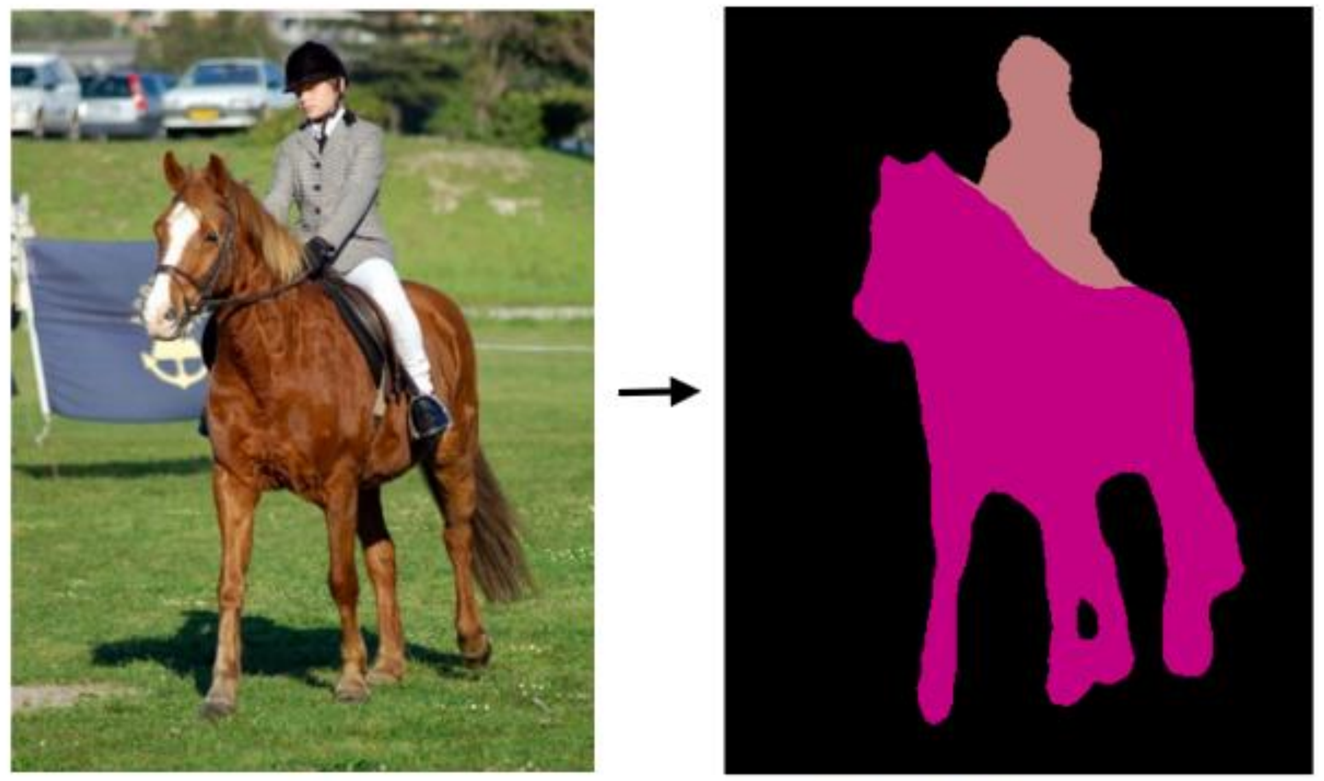


Image-to-Image Translation

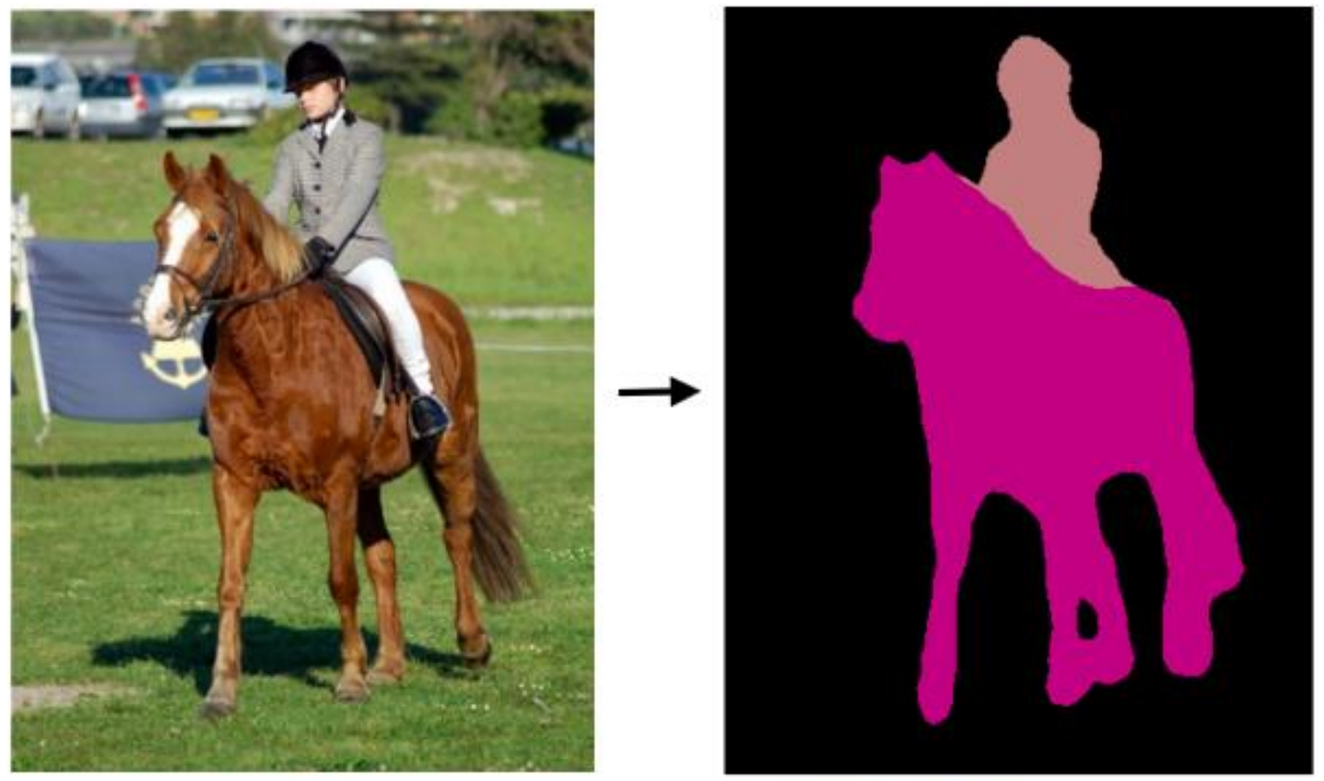
Object labeling



[Long et al. 2015]

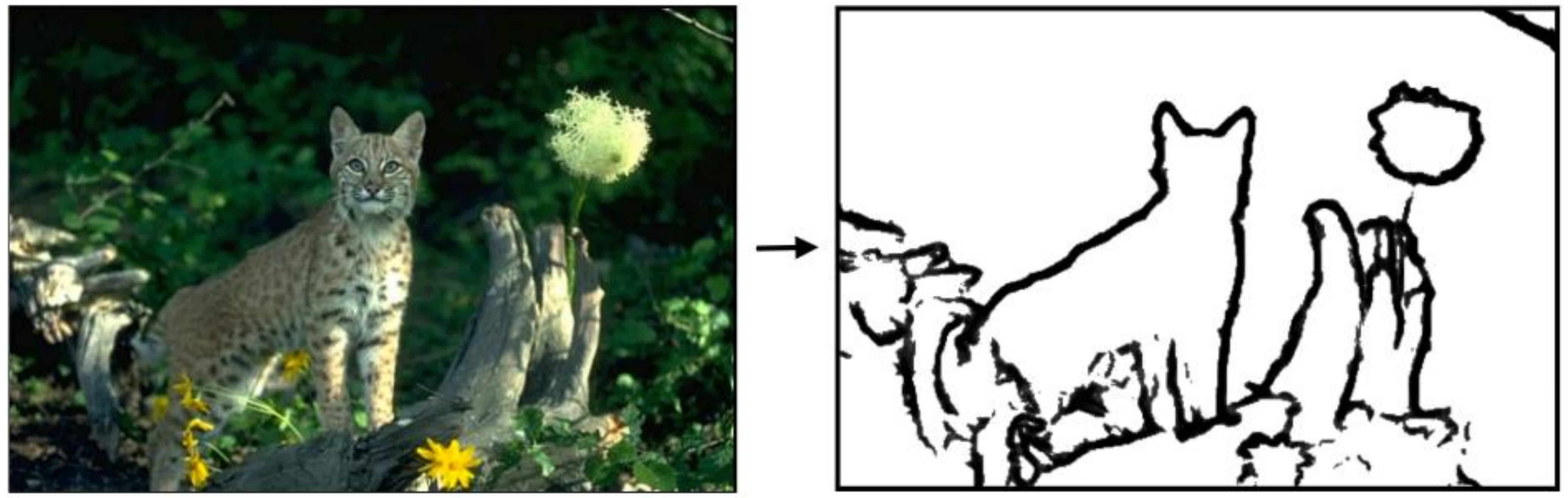
Image-to-Image Translation

Object labeling



[Long et al. 2015]

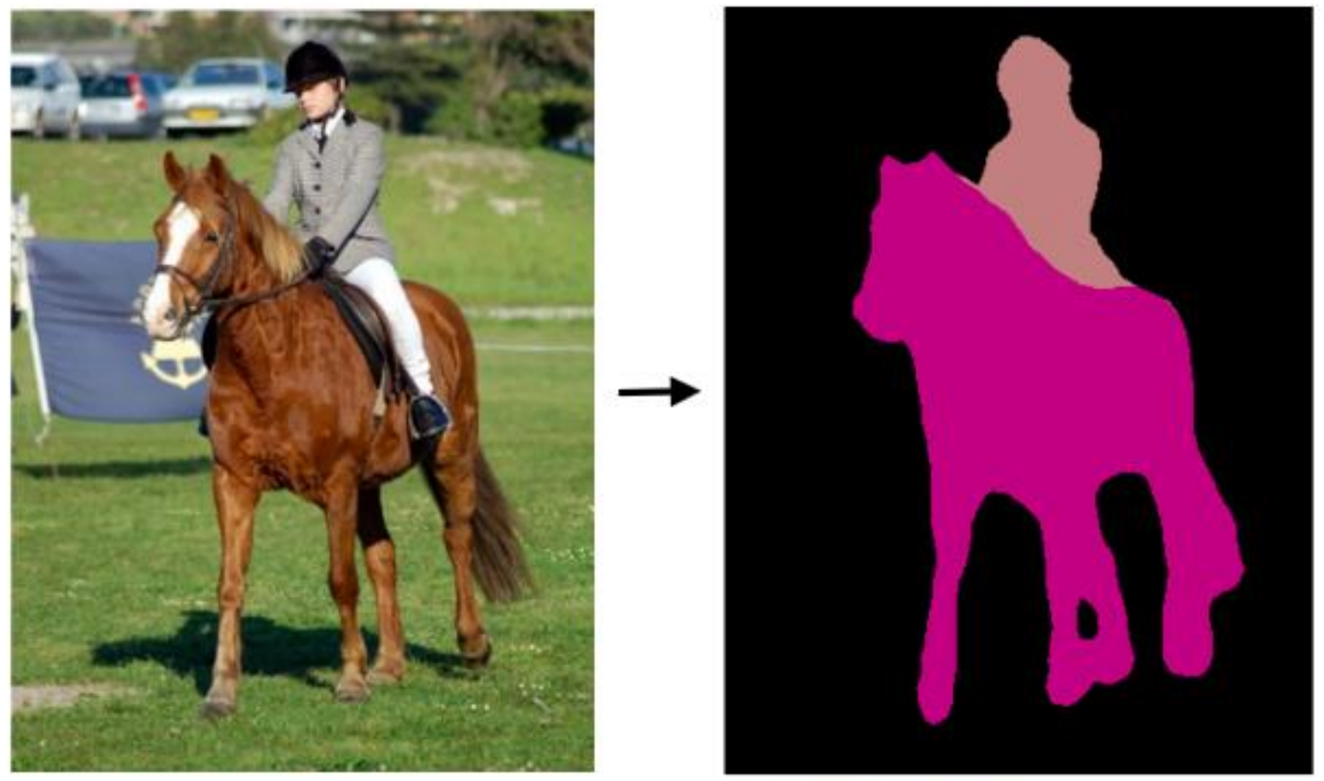
Edge Detection



[Xie et al. 2015]

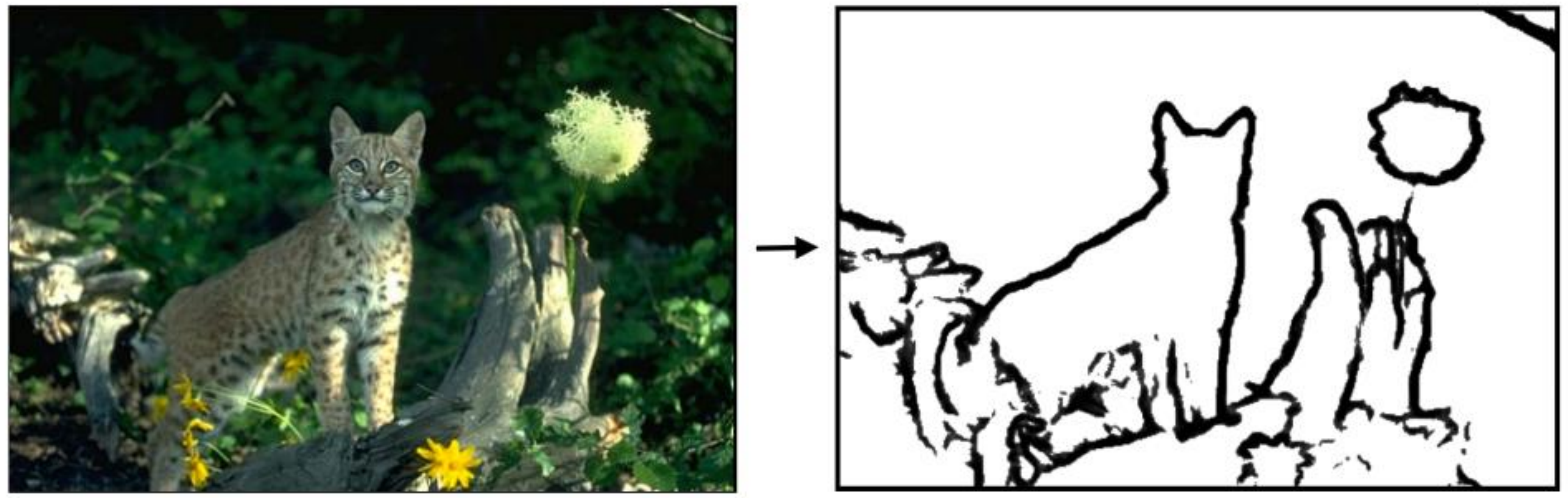
Image-to-Image Translation

Object labeling



[Long et al. 2015]

Edge Detection



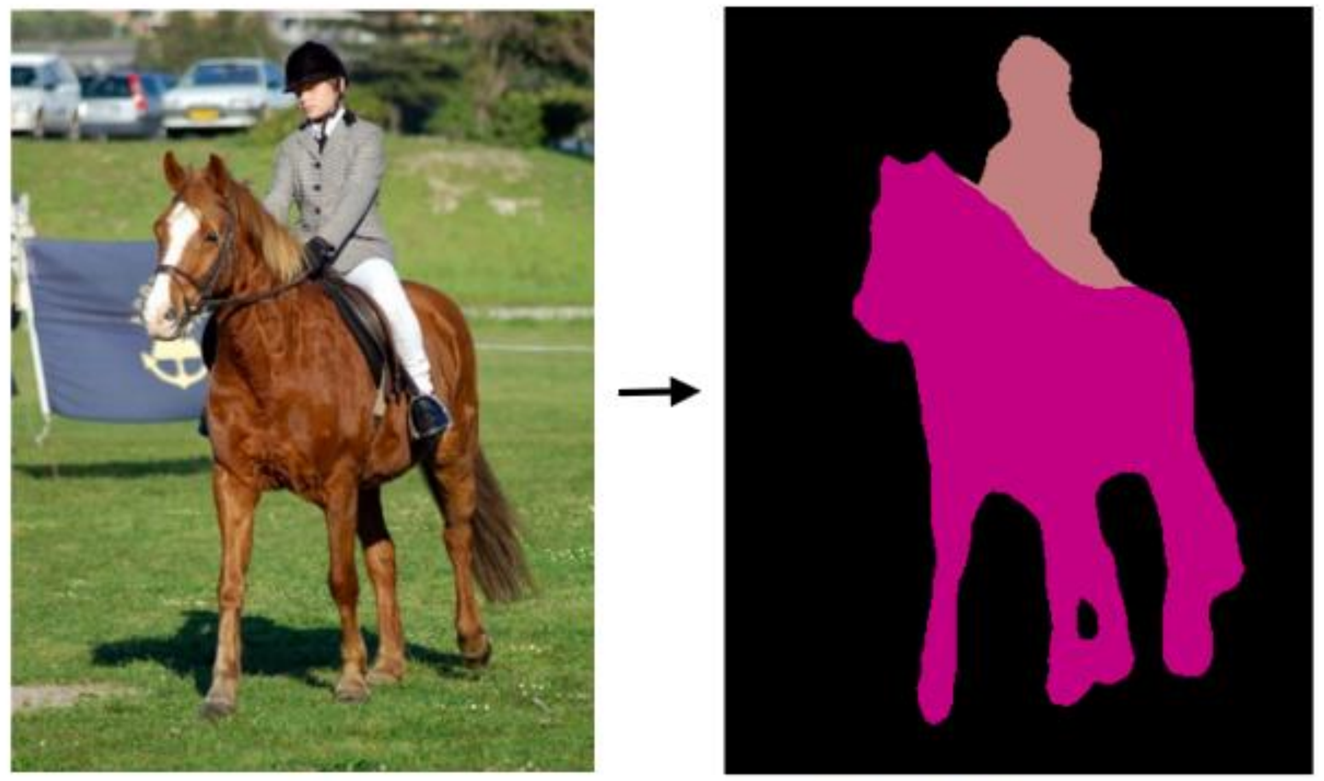
[Xie et al. 2015]

Season change

[Laffont et al. 2014]

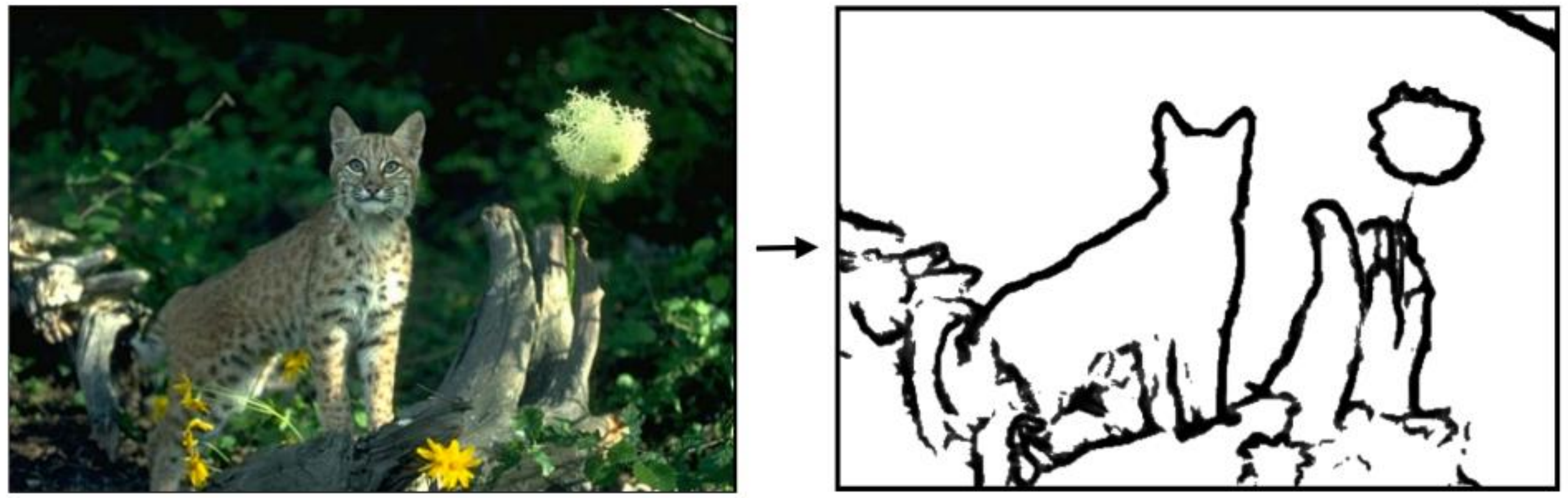
Image-to-Image Translation

Object labeling



[Long et al. 2015]

Edge Detection

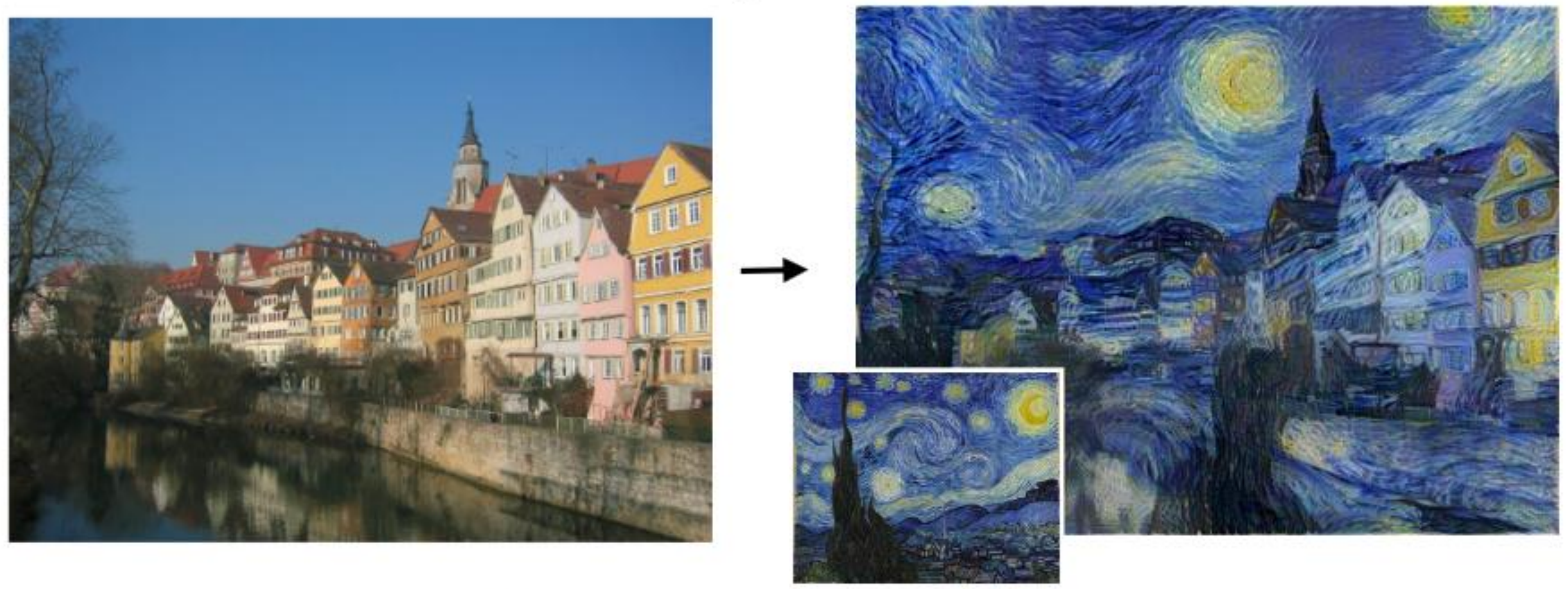


[Xie et al. 2015]

Season change

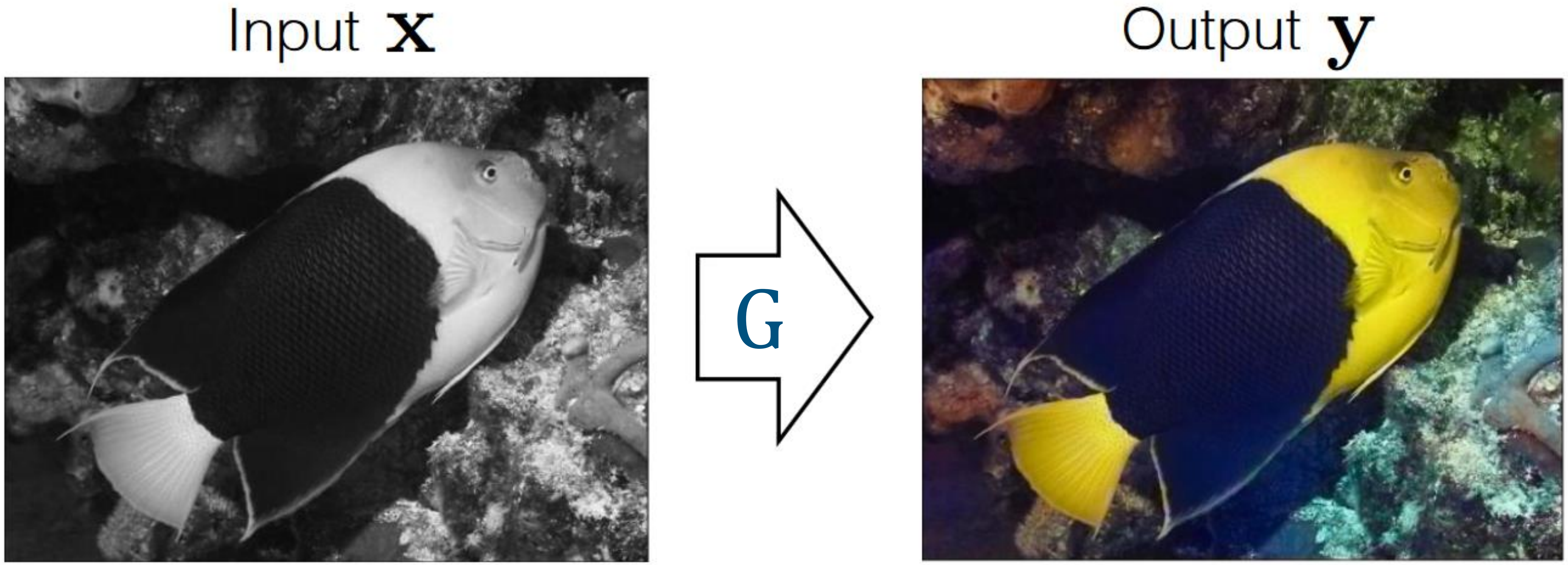
[Laffont et al. 2014]

Artistic style transfer



[Gatys et al. 2016]

Image-to-Image Translation

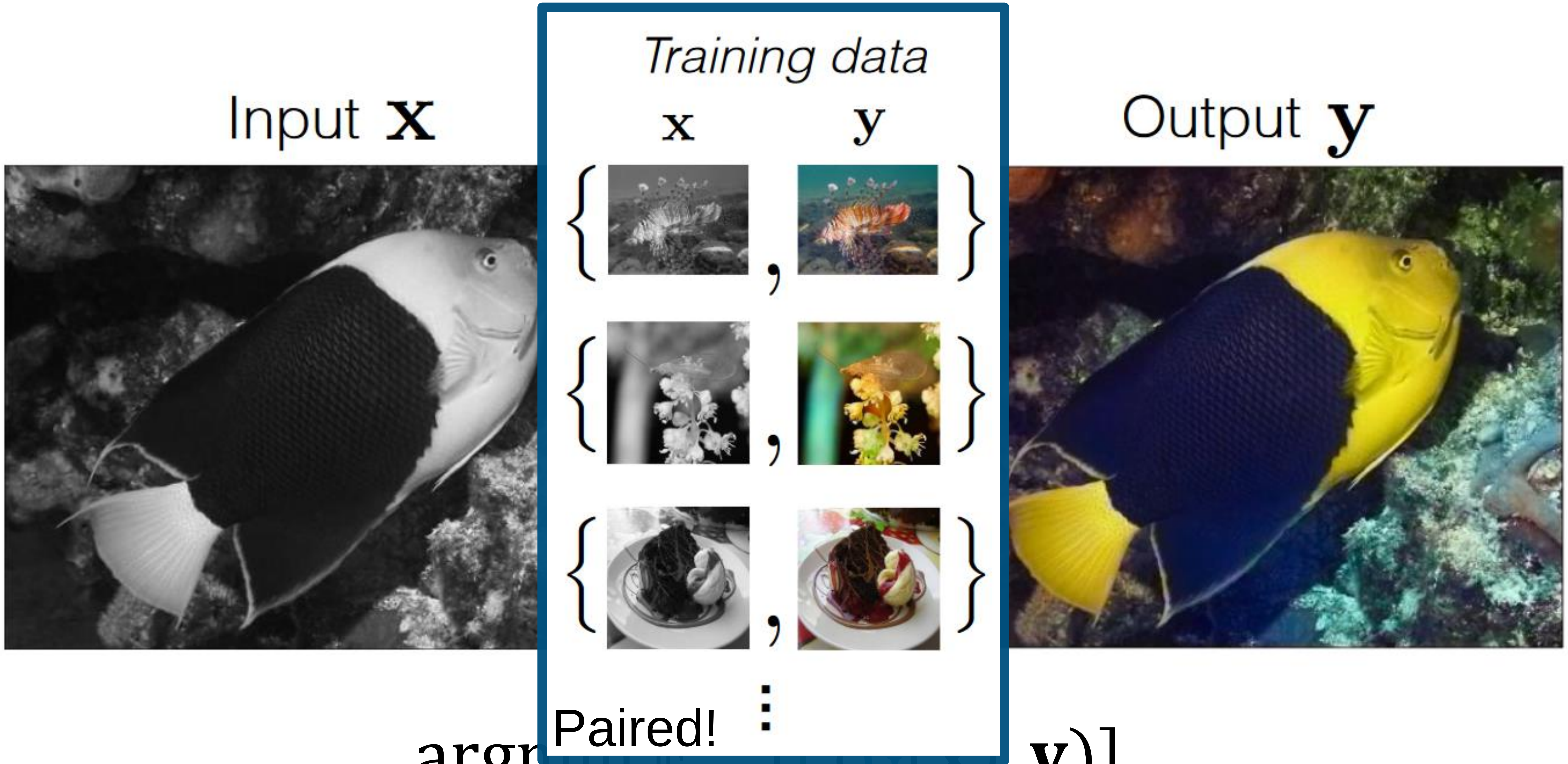


$$\underset{G}{\arg\min}\ \mathbb{E}_{\mathbf{x},\mathbf{y}}[L(G(\mathbf{x}),\mathbf{y})]$$

Loss Neural Network

[Zhang et al., ECCV 2016]

Image-to-Image Translation

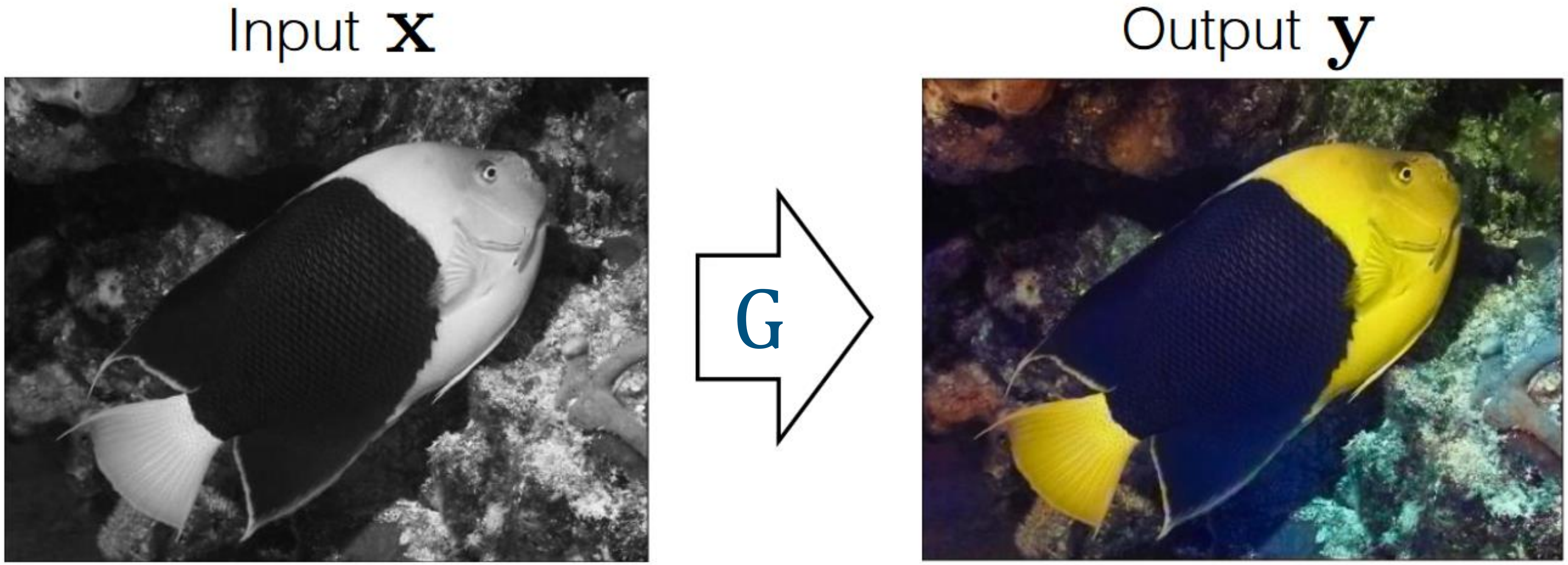


$$\arg\min_{G} \mathbb{E}_{\mathbf{x},\mathbf{y}}[L(G(\mathbf{x}), \mathbf{y})]$$

Loss Neural Network

[Zhang et al., ECCV 2016]

Image-to-Image Translation

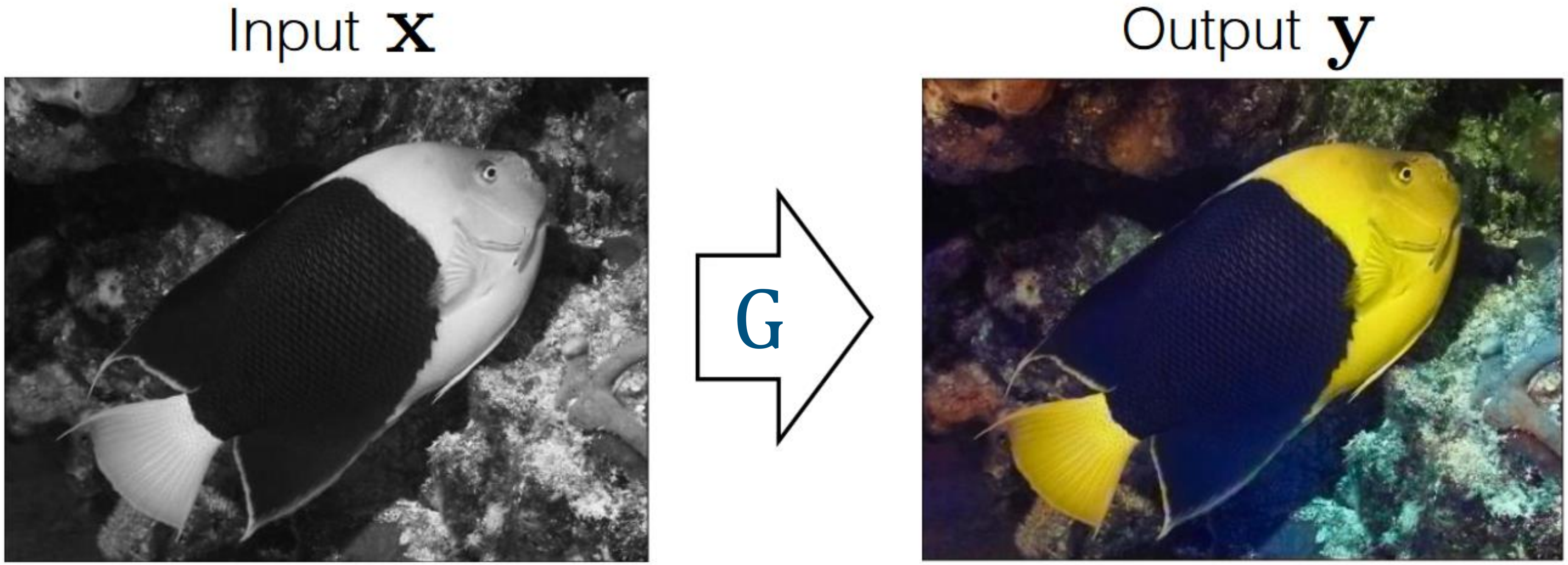


$$\underset{G}{\arg\min}\ \mathbb{E}_{\mathbf{x},\mathbf{y}}[L(G(\mathbf{x}),\mathbf{y})]$$

Loss Neural Network

[Zhang et al., ECCV 2016]

Image-to-Image Translation

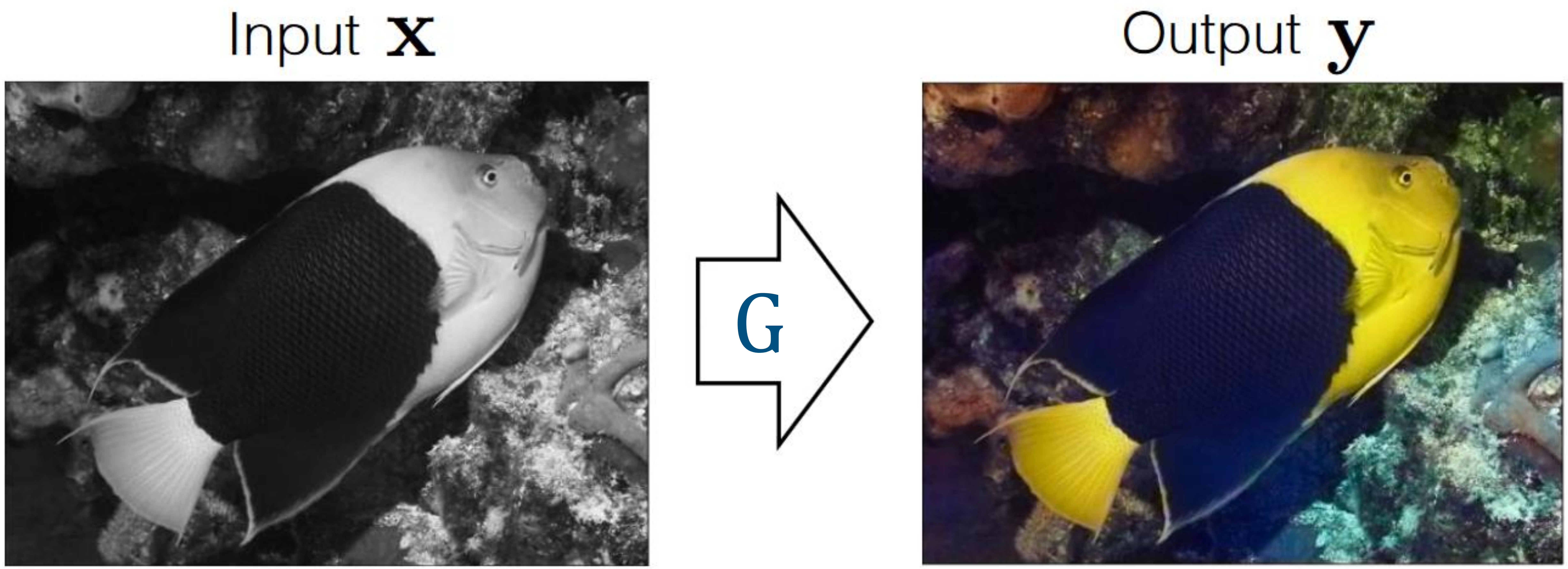


$$\underset{G}{\operatorname{argmin}}\ \mathbb{E}_{\mathbf{x},\mathbf{y}}[L(G(\mathbf{x}),\mathbf{y})]$$

"**What** should I do?" Neural Network

[Zhang et al., ECCV 2016]

Image-to-Image Translation



$$\underset{G}{\arg\min}\ \mathbb{E}_{\mathbf{x},\mathbf{y}}[L(G(\mathbf{x}),\mathbf{y})]$$

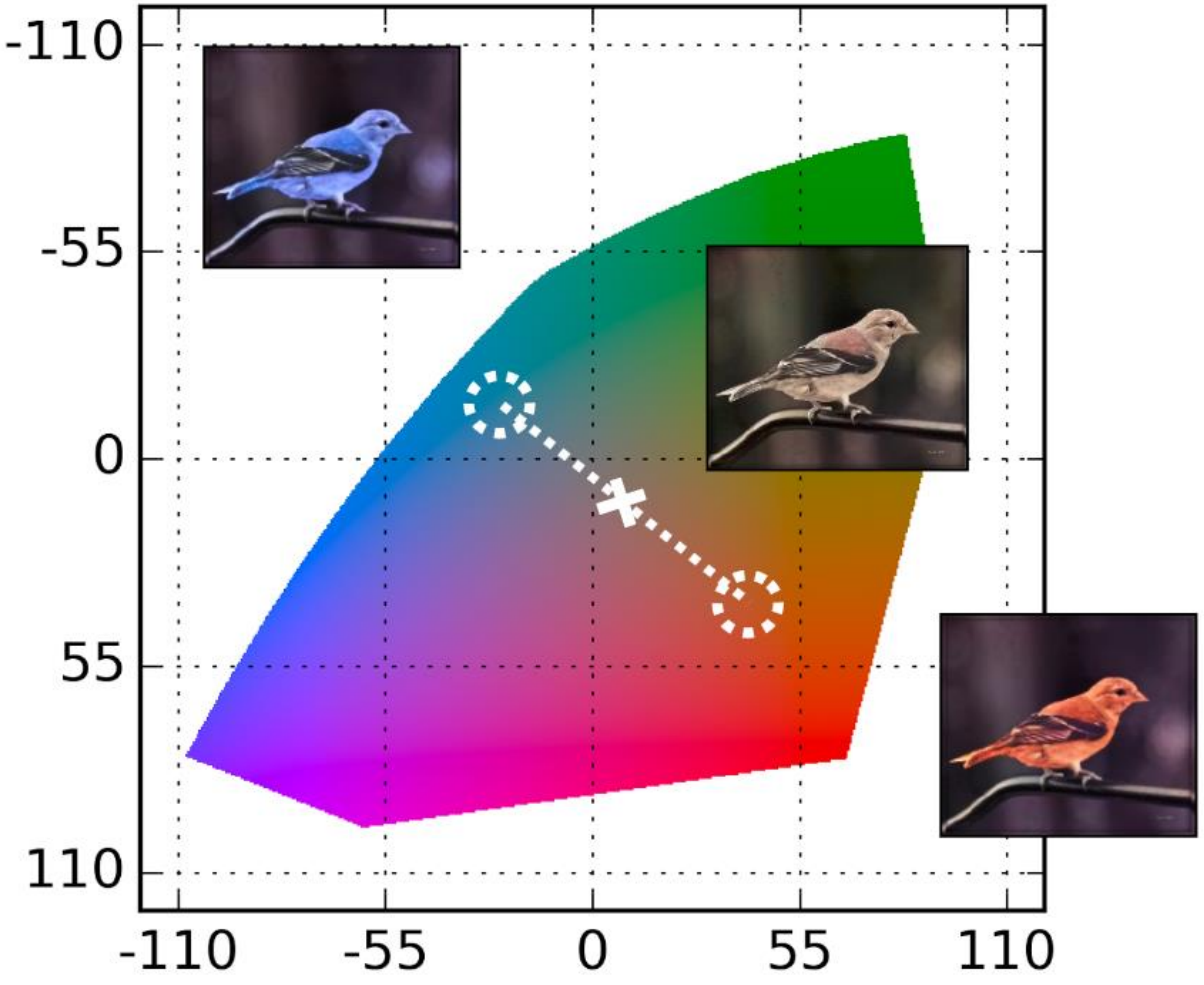
“**What** should I do?” “**How** should I do it?”

[Zhang et al., ECCV 2016]

Be careful what you wish for!

$$L(\hat{\mathbf{y}}, \mathbf{y}) = ||\hat{\mathbf{y}} - \mathbf{y}||_2^2$$

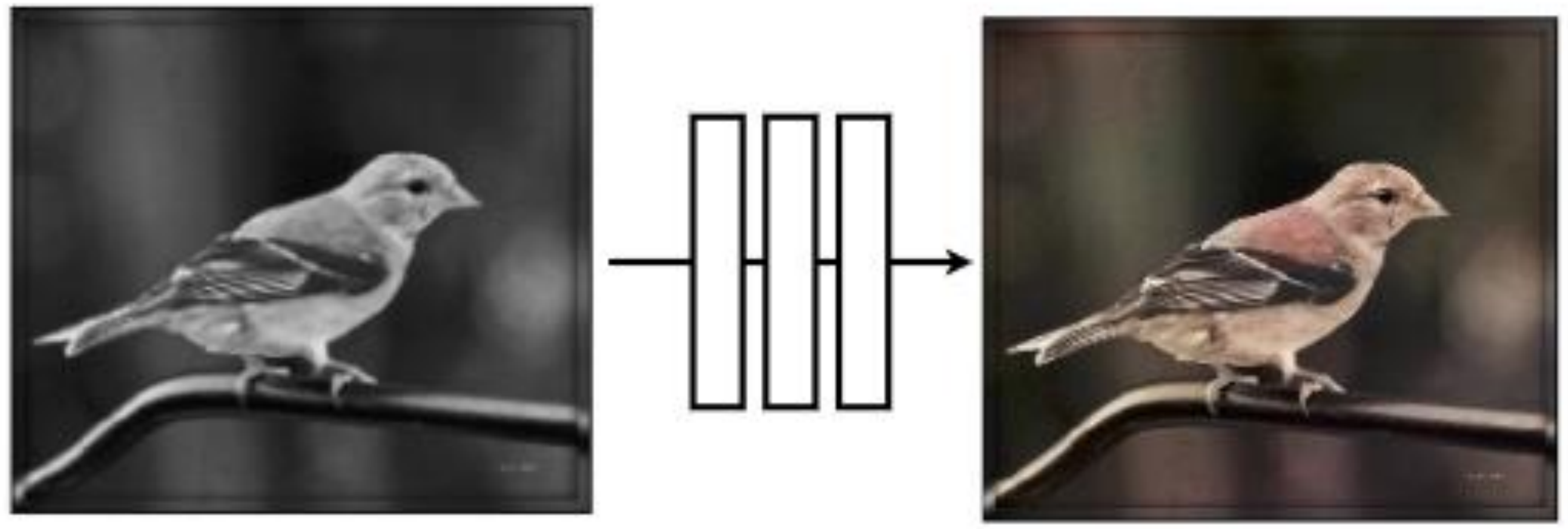
Degradation to the mean!



$$L(\hat{\mathbf{y}}, \mathbf{y}) = \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2$$

Automate Design of the Loss?

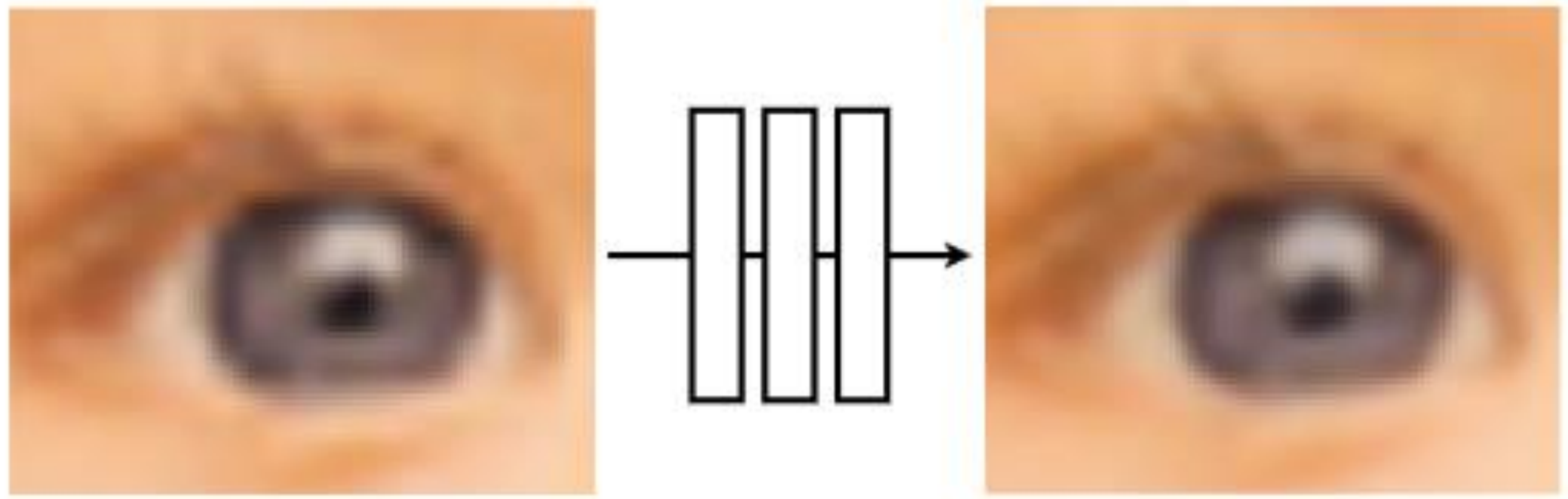
Image colorization



L2 regression

[Zhang, Isola, Efros, ECCV 2016]

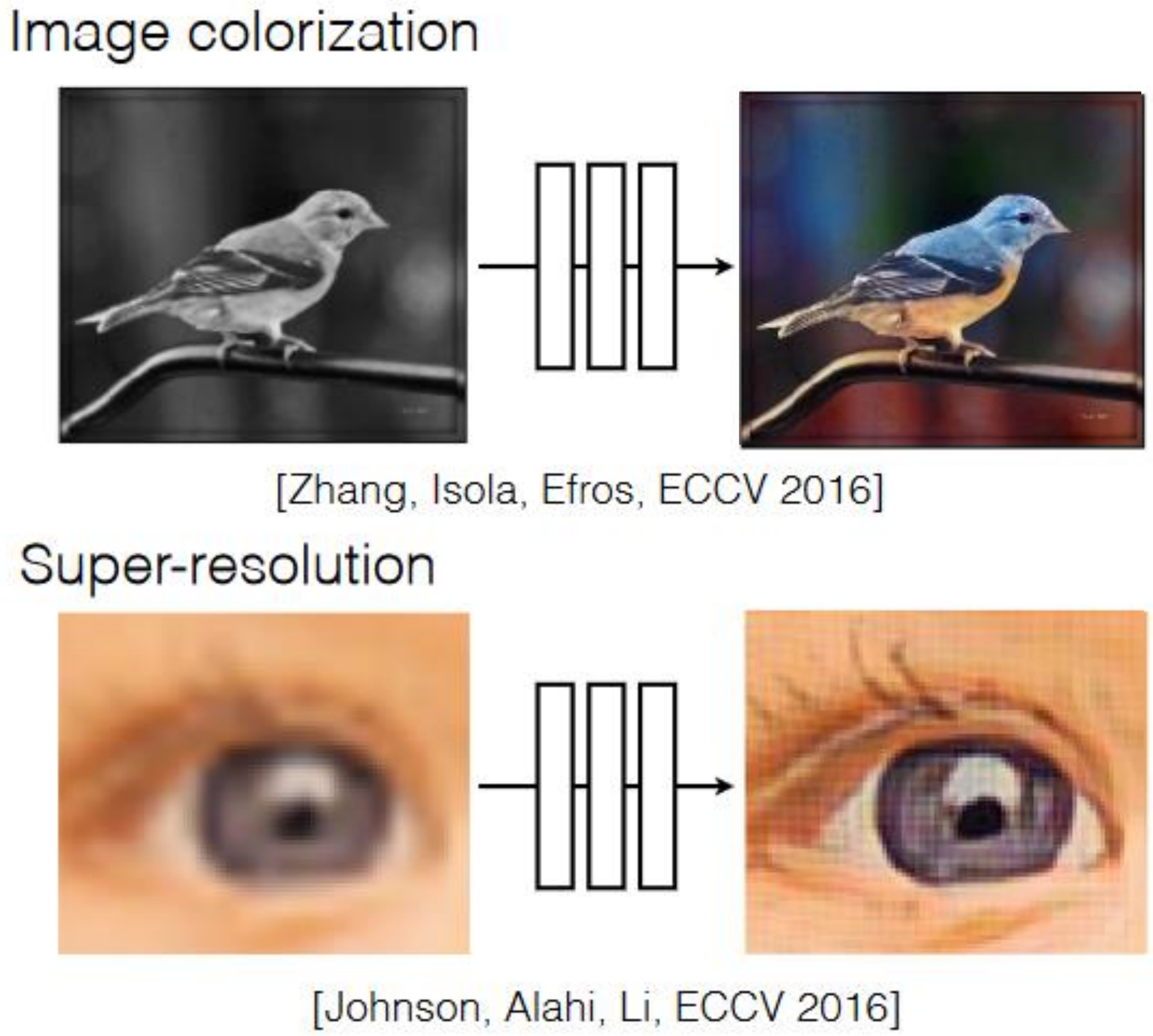
Super-resolution



L2 regression

[Johnson, Alahi, Li, ECCV 2016]

Automate Design of the Loss?

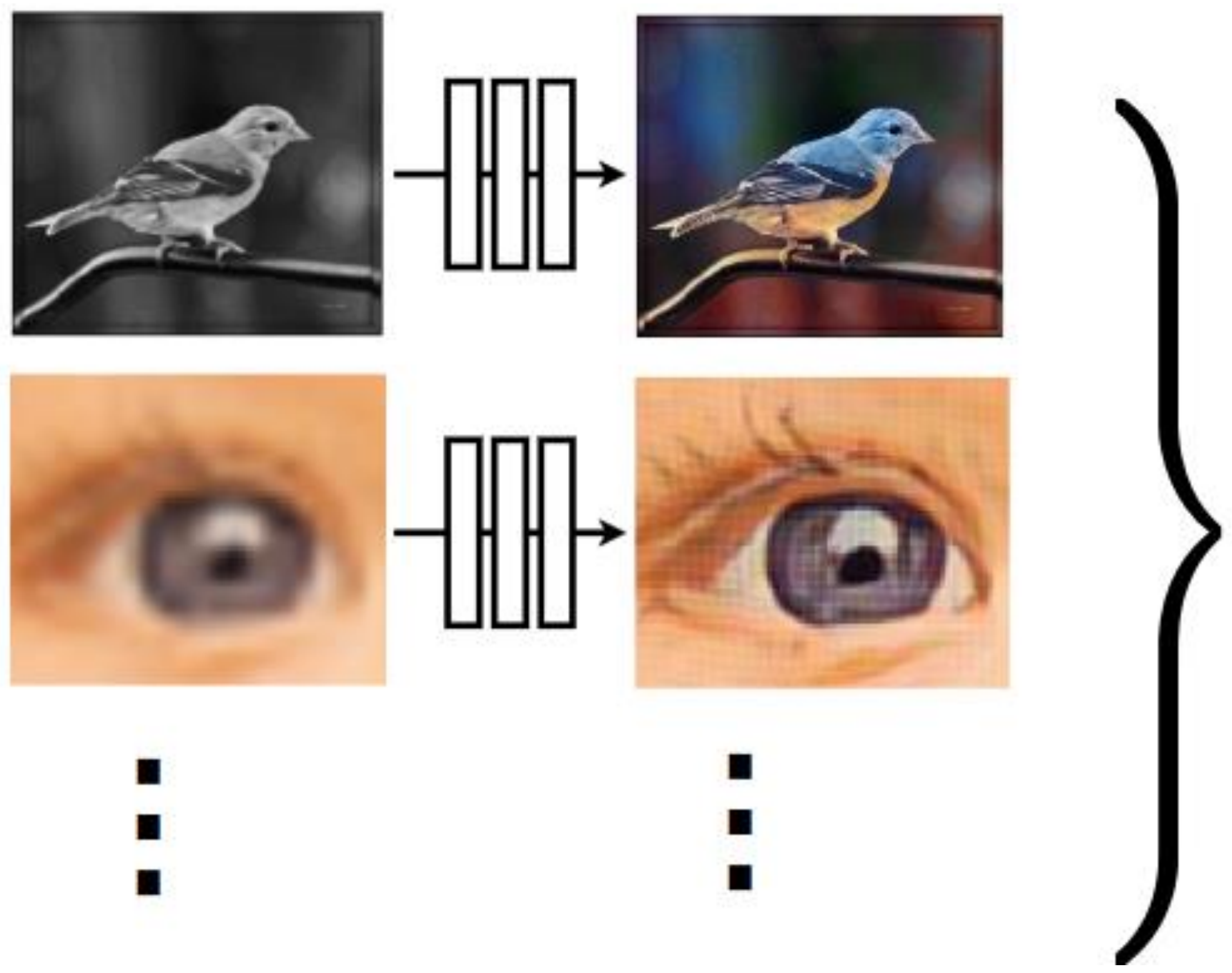


Cross entropy objective,
with colorfulness term

Deep feature covariance
matching objective

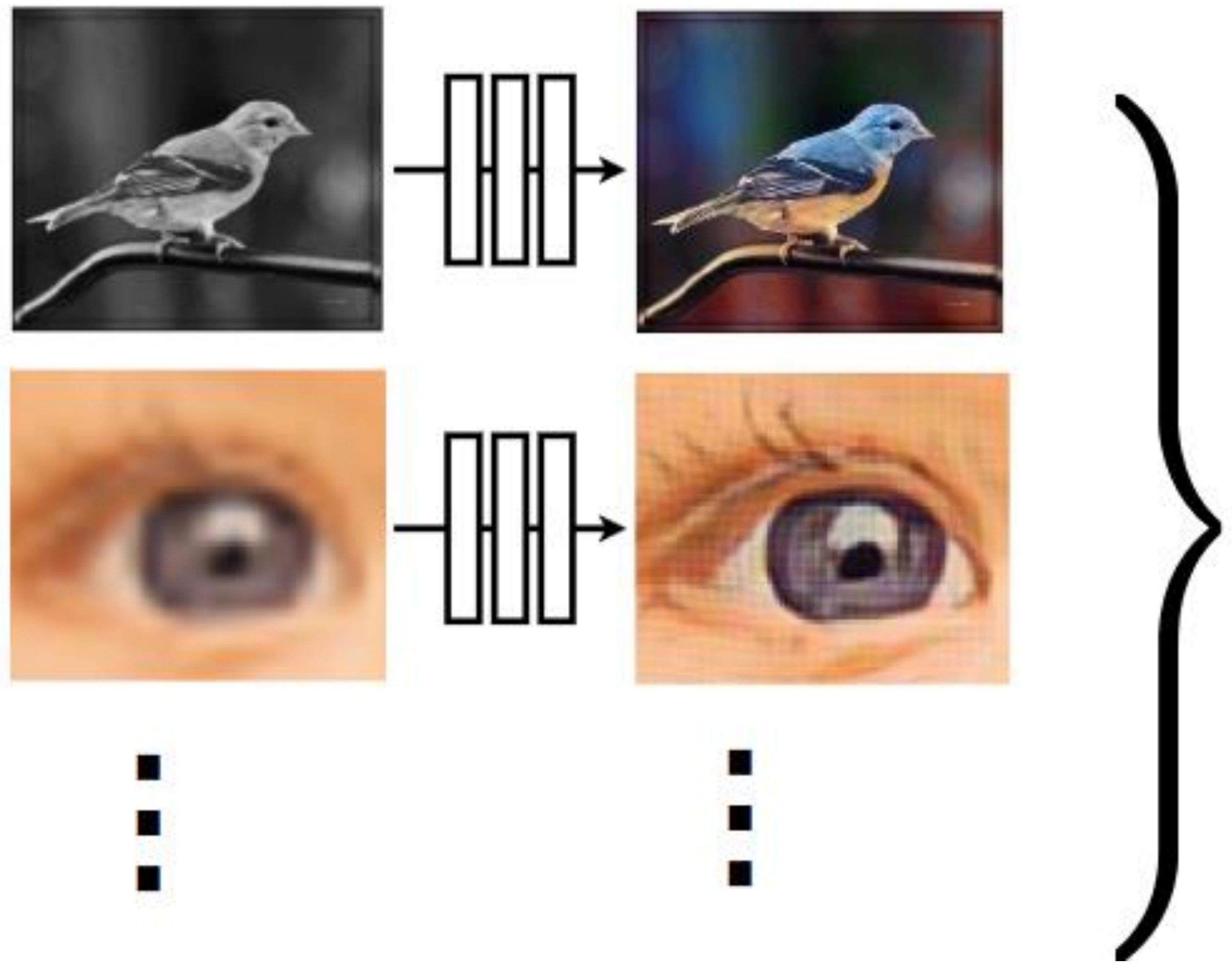
Automate Design of the Loss?

Deep learning got rid of handcrafted features. Can we also get rid of handcrafting the loss function?



Automate Design of the Loss?

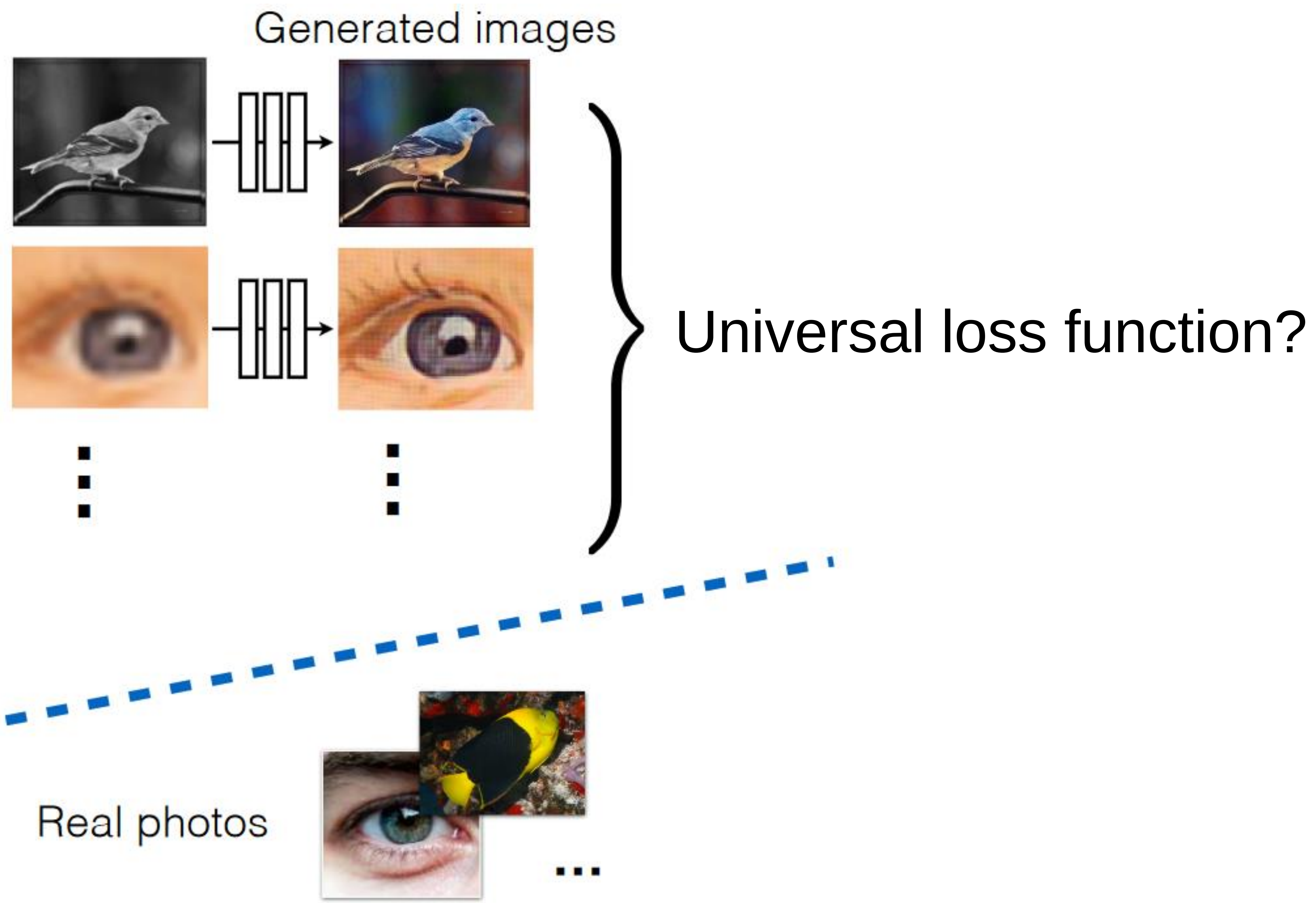
Deep learning got rid of handcrafted features. Can we also get rid of handcrafting the loss function?



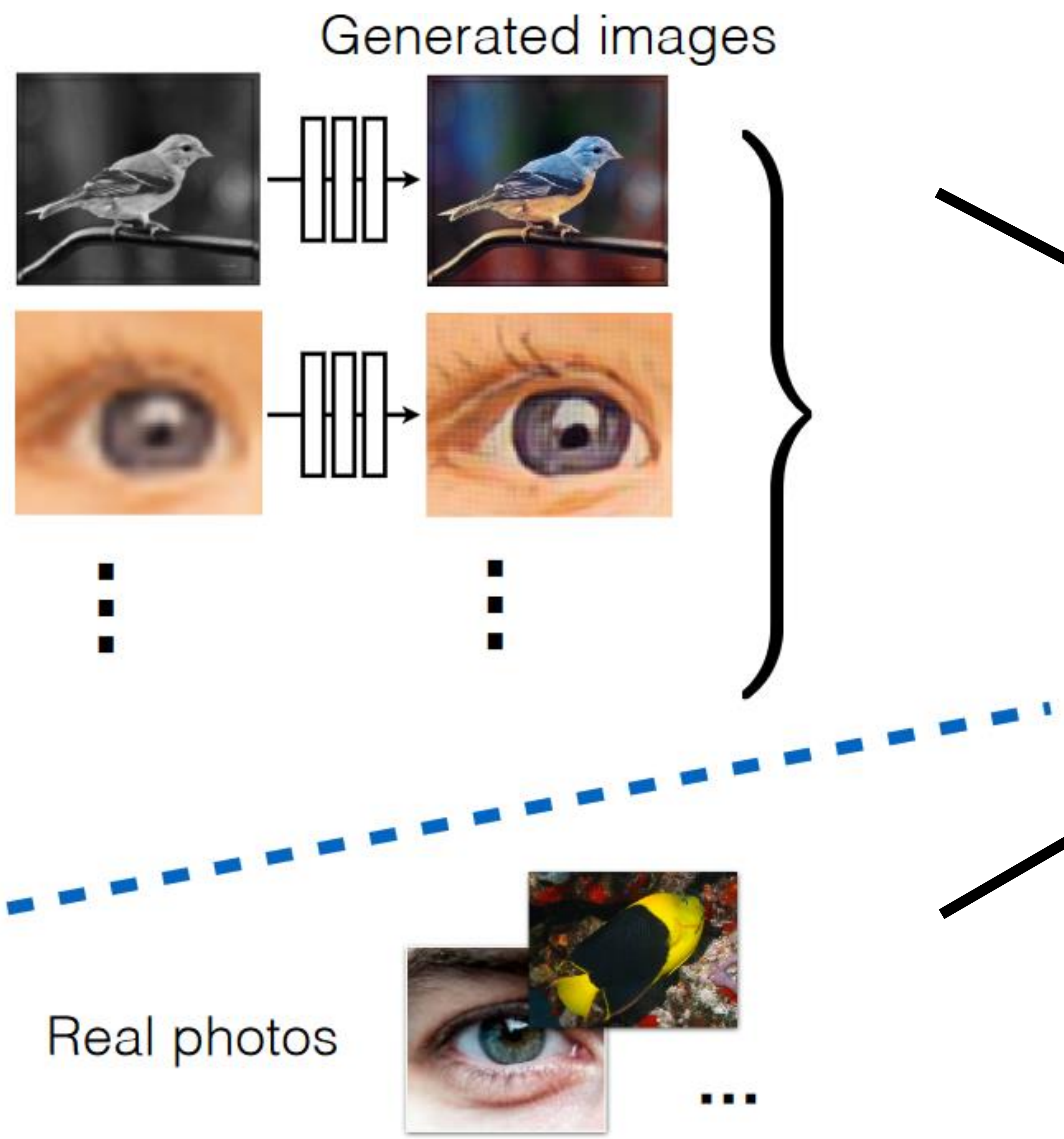
Universal loss function?

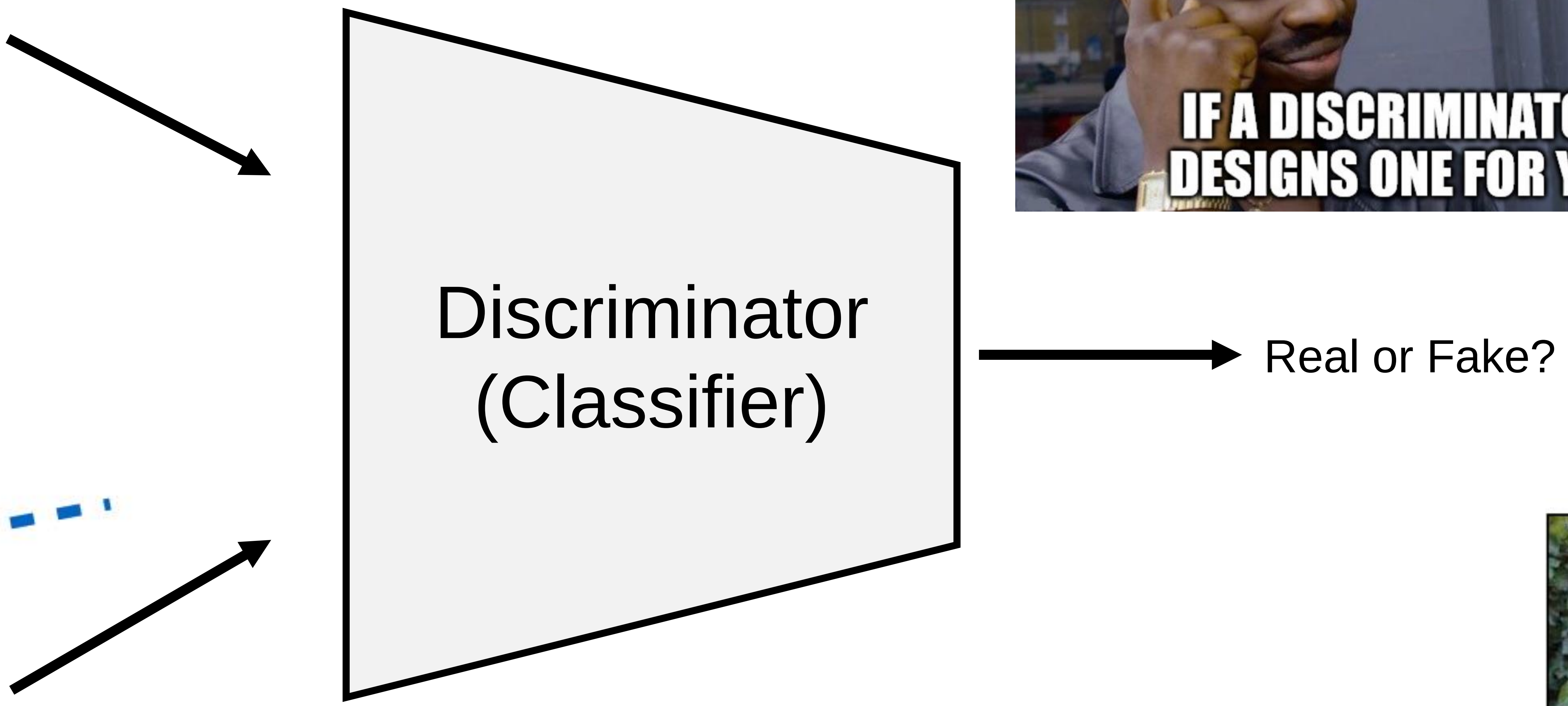
Automate Design of the Loss?

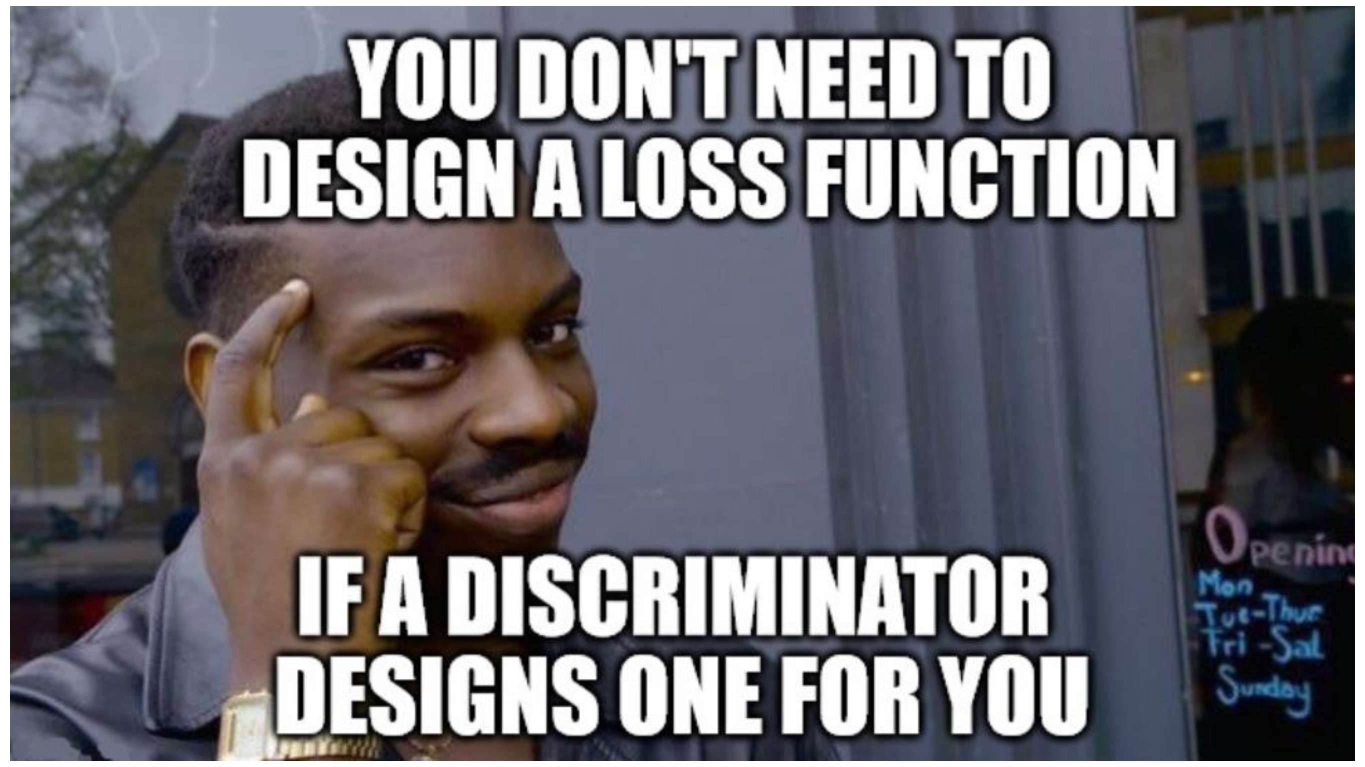
Deep learning got rid of handcrafted features. Can we also get rid of handcrafting the loss function?



Discriminator as a Loss Function

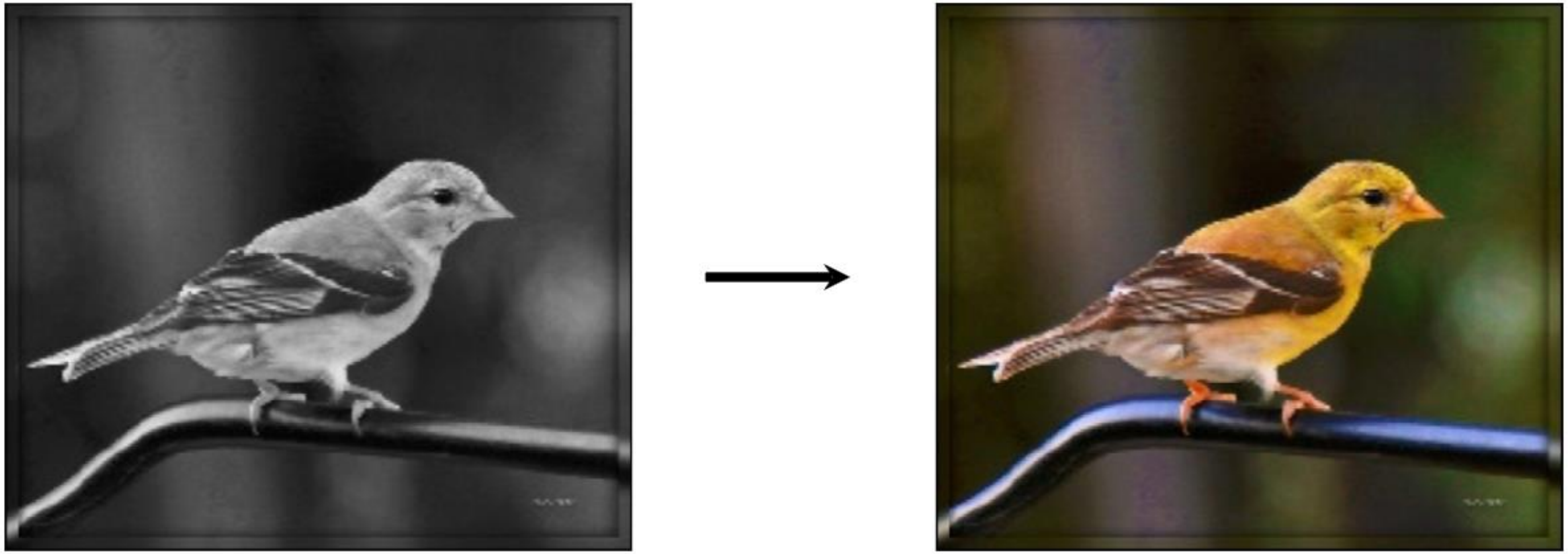




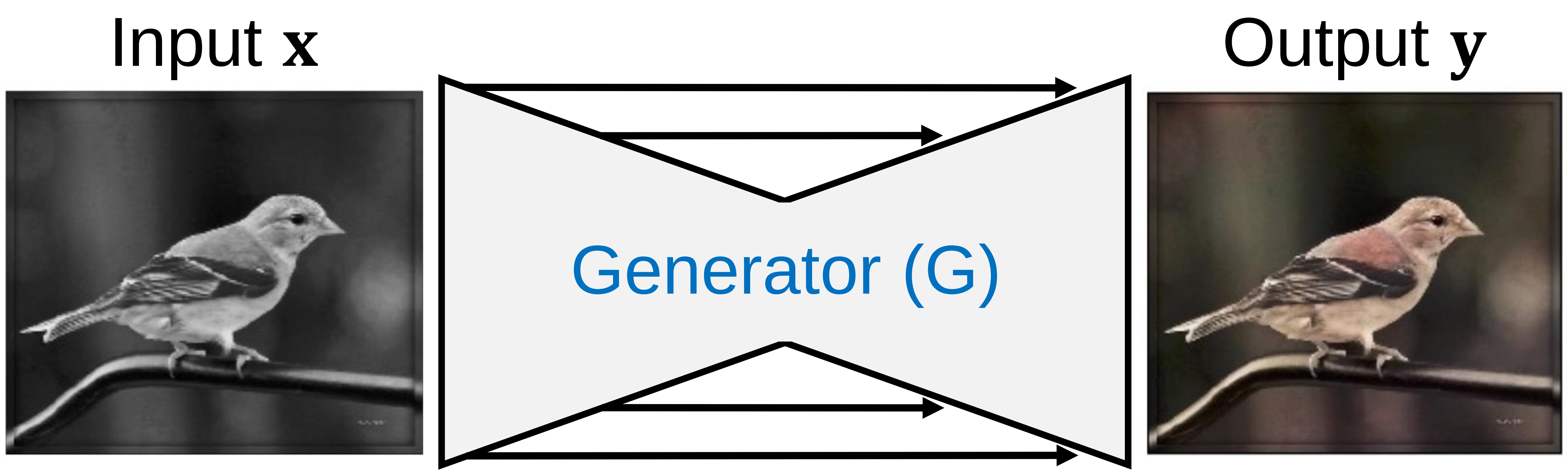


[Goodfellow, Pouget-Abadie, Mirza, Xu, Warde-Farley, Ozair, Courville, Bengio 2014]

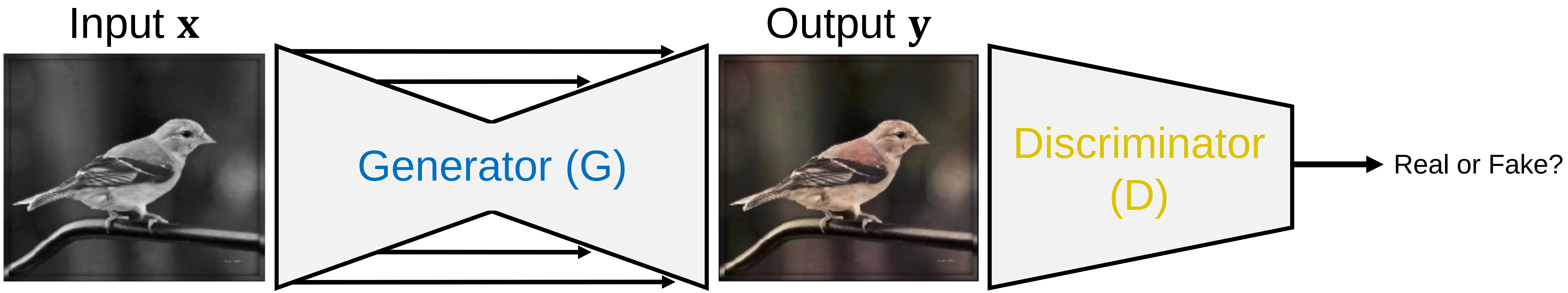
Conditional GAN



Conditional GAN



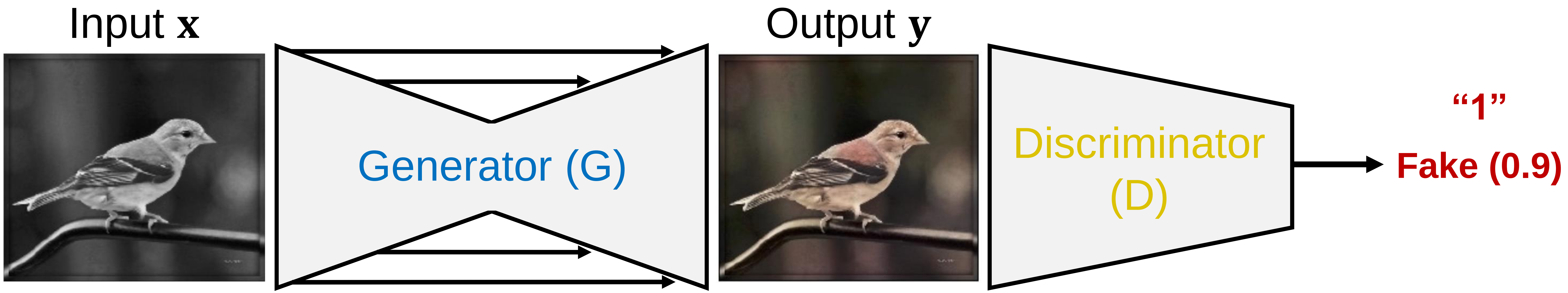
Conditional GAN

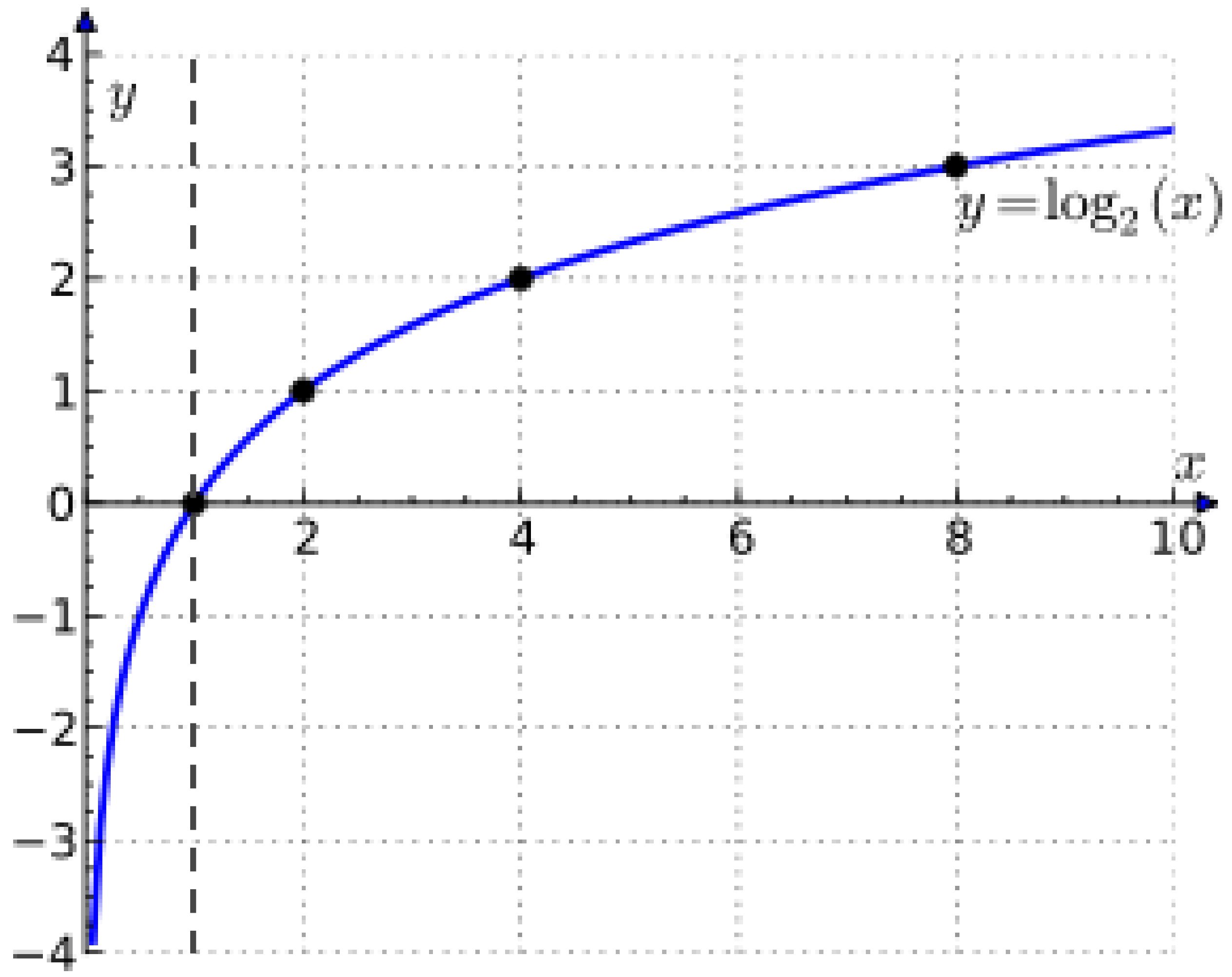


G tries to synthesize fake images that fool D

D tries to tell real from fake

Conditional GAN (Discriminator)

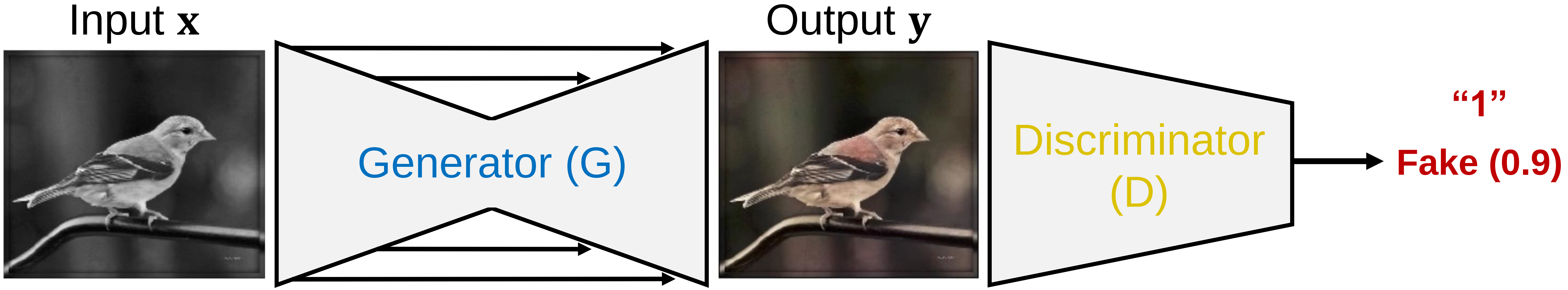


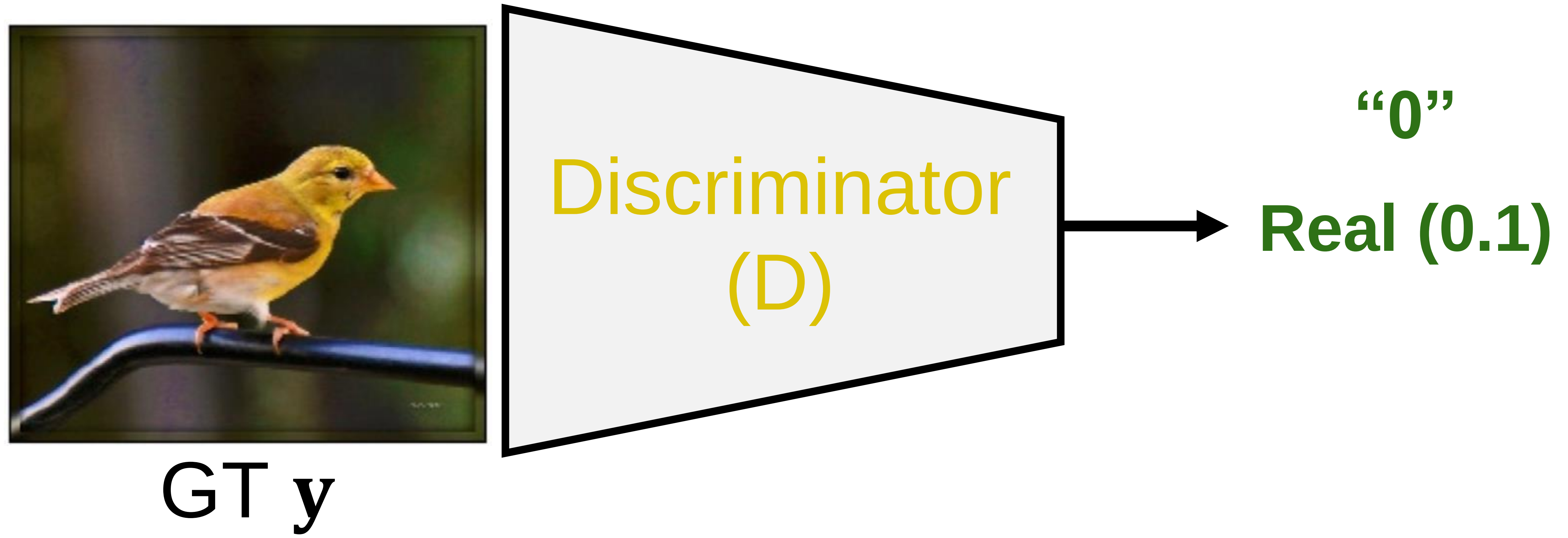


D tries to identify the fakes

$$\arg\max_{D} \mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\boxed{\log D(G(\mathbf{x}))} + \qquad\qquad\qquad\qquad \right]$$

Conditional GAN (Discriminator)



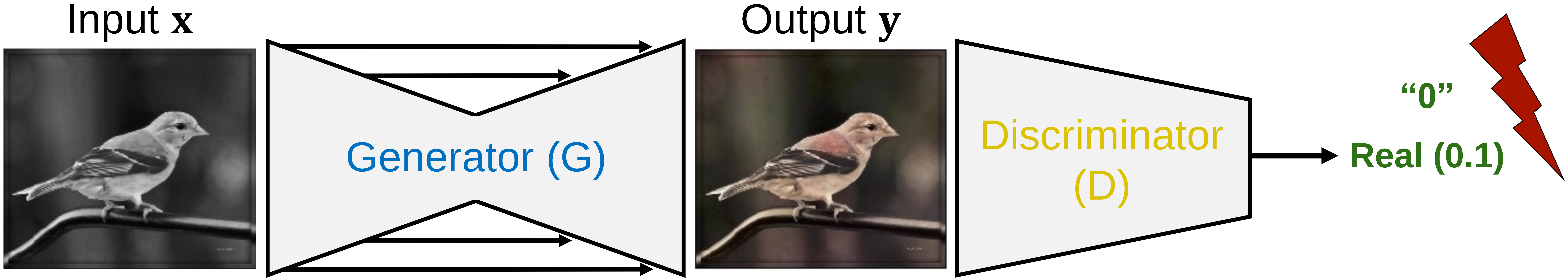


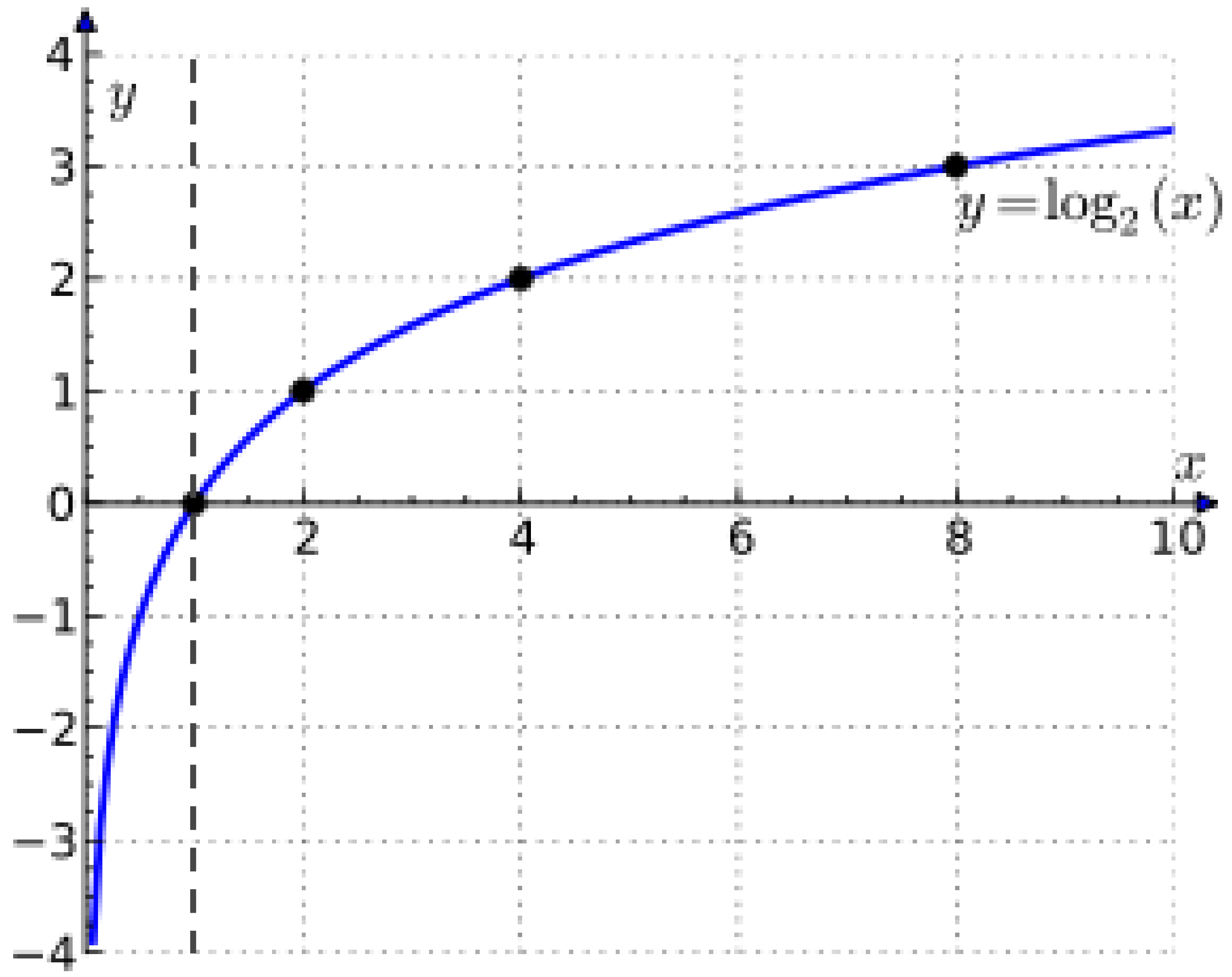
D tries to identify the fakes

D tries to identify the real images

$$\arg\max_{D} \mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\boxed{\log D(G(\mathbf{x}))} + \boxed{\log(1 - D(\mathbf{y}))}\right]$$

Conditional GAN (Generator)

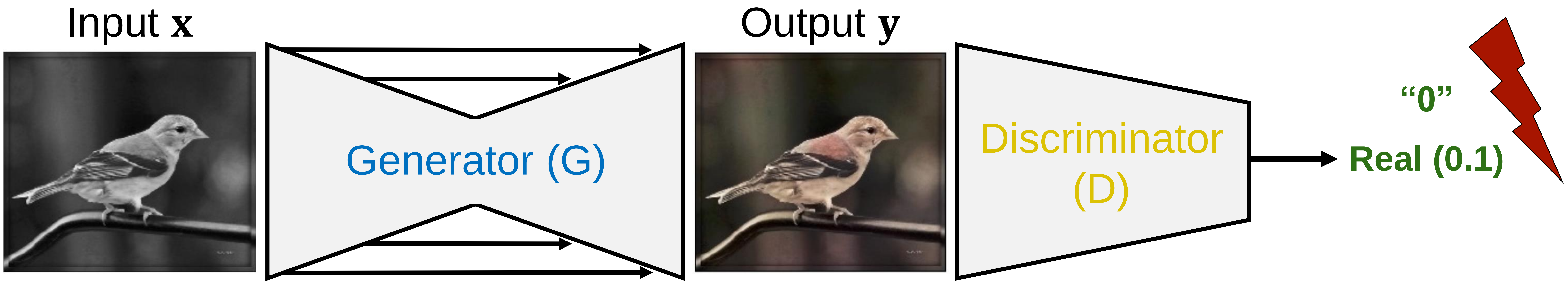


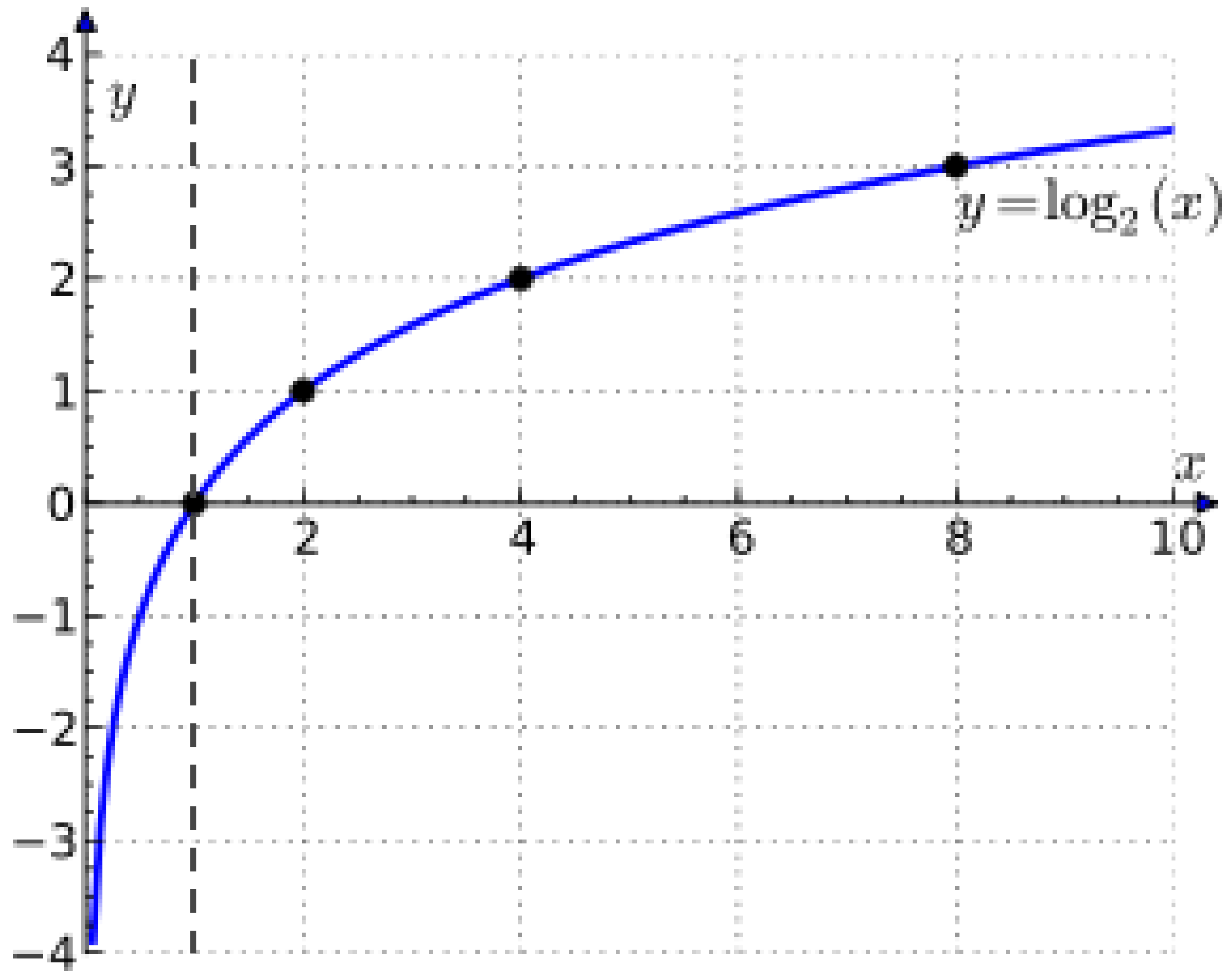


G tries to synthesize fake images that fool D.

$$\arg\min_{G} \mathbb{E}_{\mathbf{x},\mathbf{y}}[\, \log D\big(G(\mathbf{x})\big) + \log\big(1 - D(\mathbf{y})\big)\,]$$

Conditional GAN (Generator)

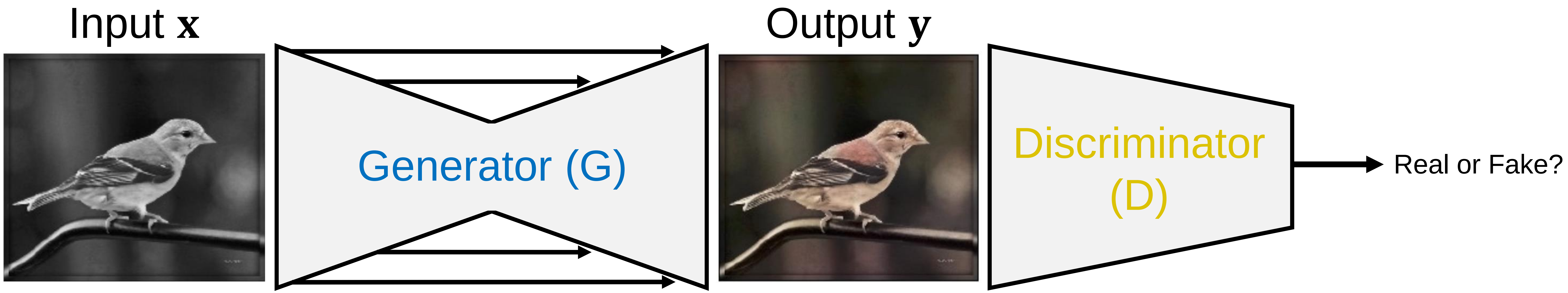




G tries to synthesize fake images that fool D.

$$\arg\min_{G} \mathbb{E}_{\mathbf{x},\mathbf{y}}\left[\boxed{\log D(G(\mathbf{x}))} + \cancel{\log(1 - D(\mathbf{y}))}\right]$$

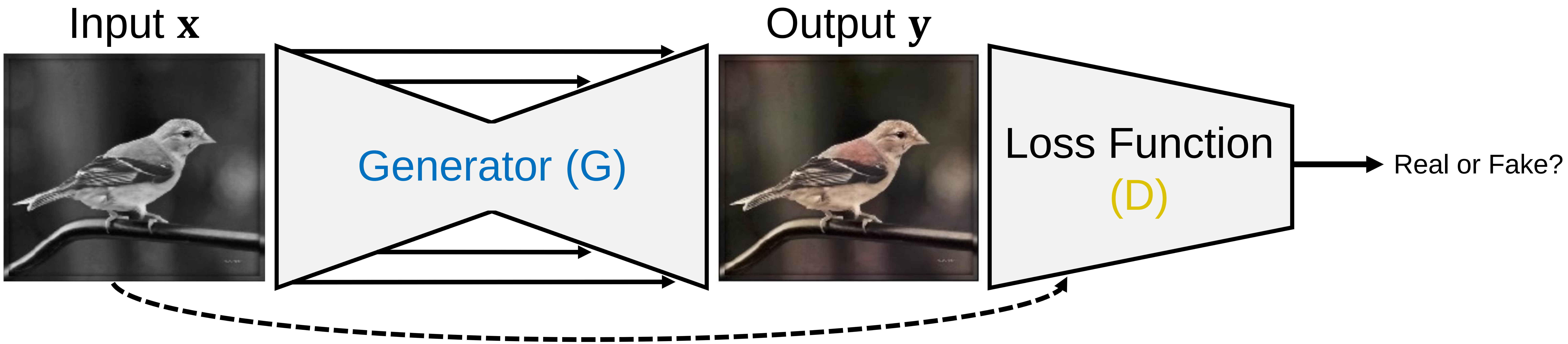
Conditional GAN



G tries to synthesize fake images that fool the best D.

$$\arg\min_{G}\max_{D}\mathbb{E}_{\mathbf{x},\mathbf{y}}[\,\log D\big(G(\mathbf{x})\big) + \log\big(1 - D(\mathbf{y})\big)\,]$$

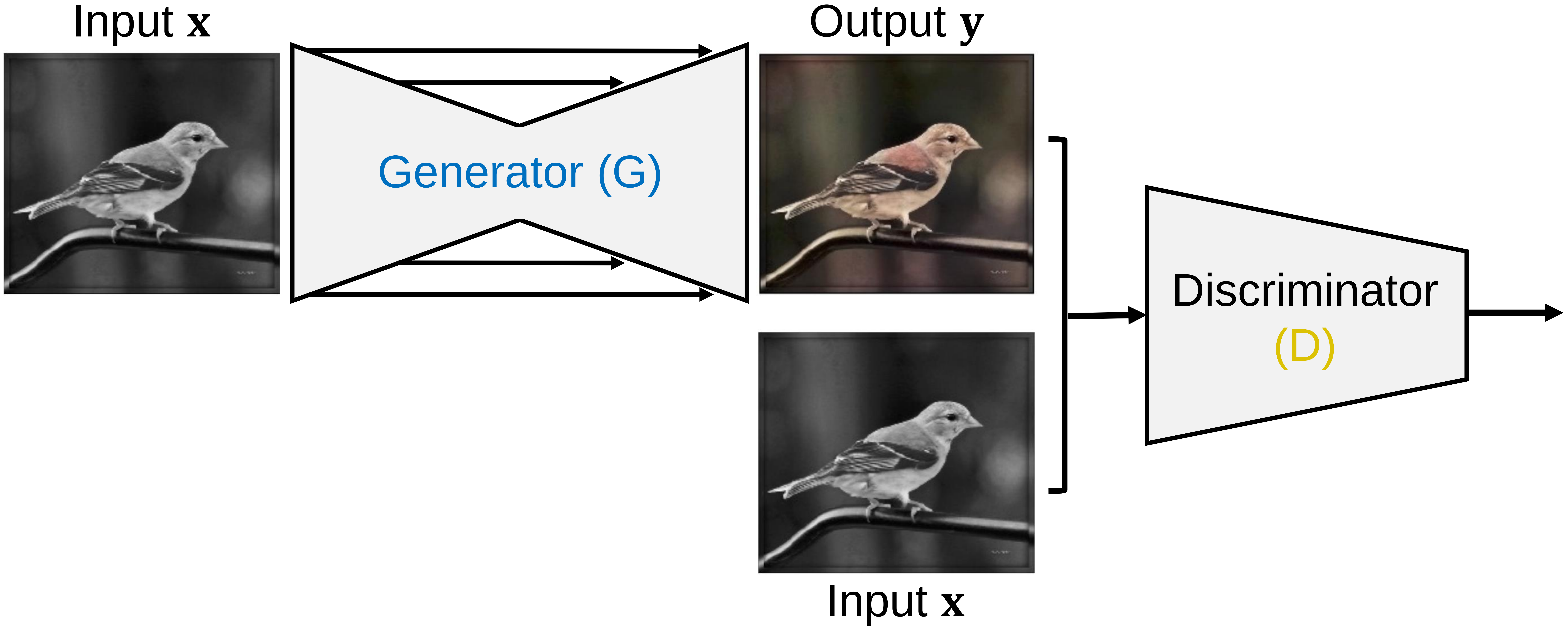
Conditional GAN



G's perspective: D is a loss function

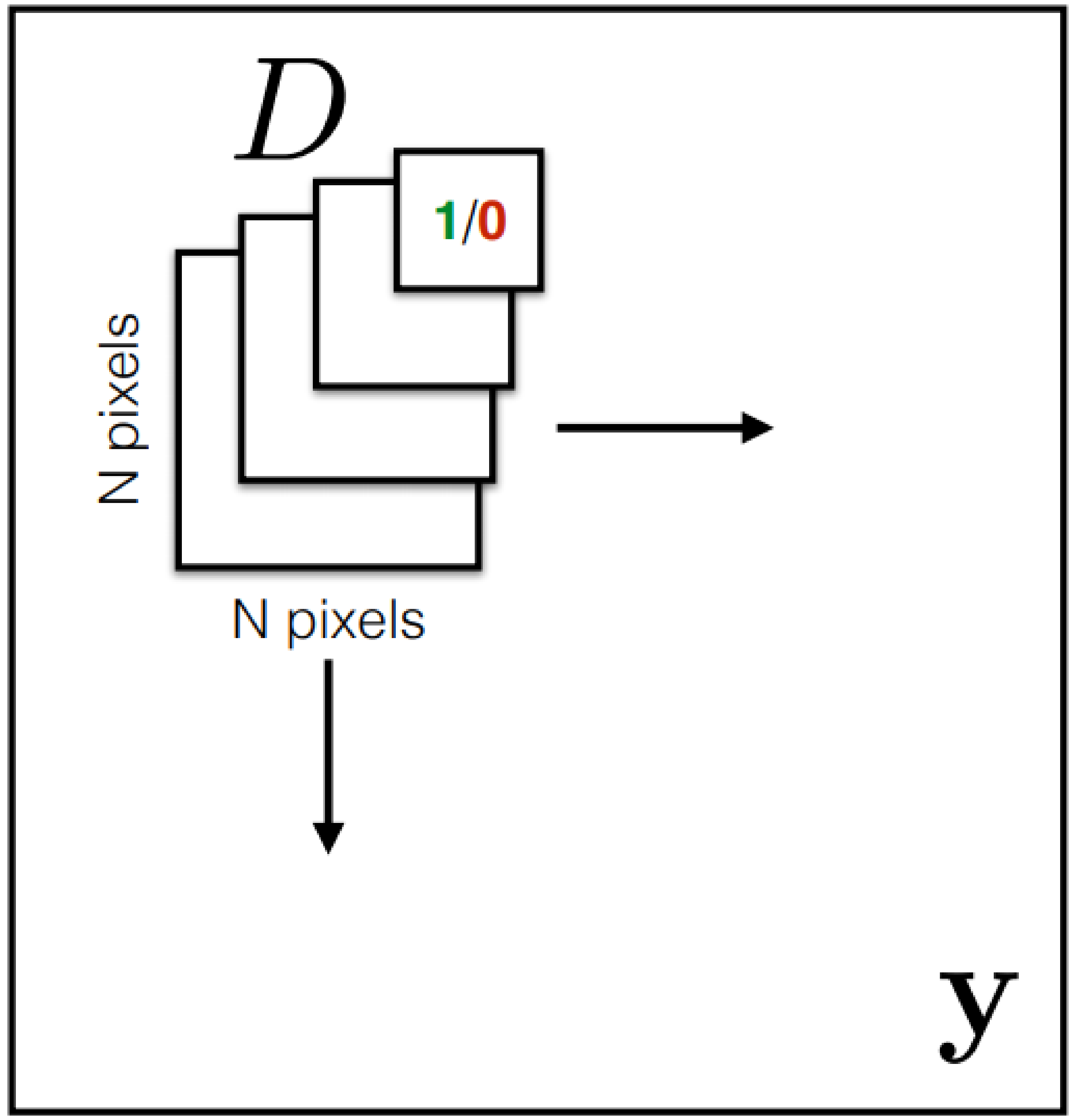
Rather than being hand-designed, it is *learned jointly*!

Conditional Discriminator



$$\arg\min_{G}\max_{D}\mathbb{E}_{\mathbf{x},\mathbf{y}}[\,\log D\big(\mathbf{x}, G(\mathbf{x})\big) + \log\big(1 - D(\mathbf{x},\mathbf{y})\big)\,]$$

Patch Discriminator



"Rather than penalizing if the output image looks fake, penalize if each overlapping patch in the output looks fake"

[Li & Wand 2016]
[Shrivastava et al. 2017]
[Isola et al. 2017]

1x1 Pixel Discriminator

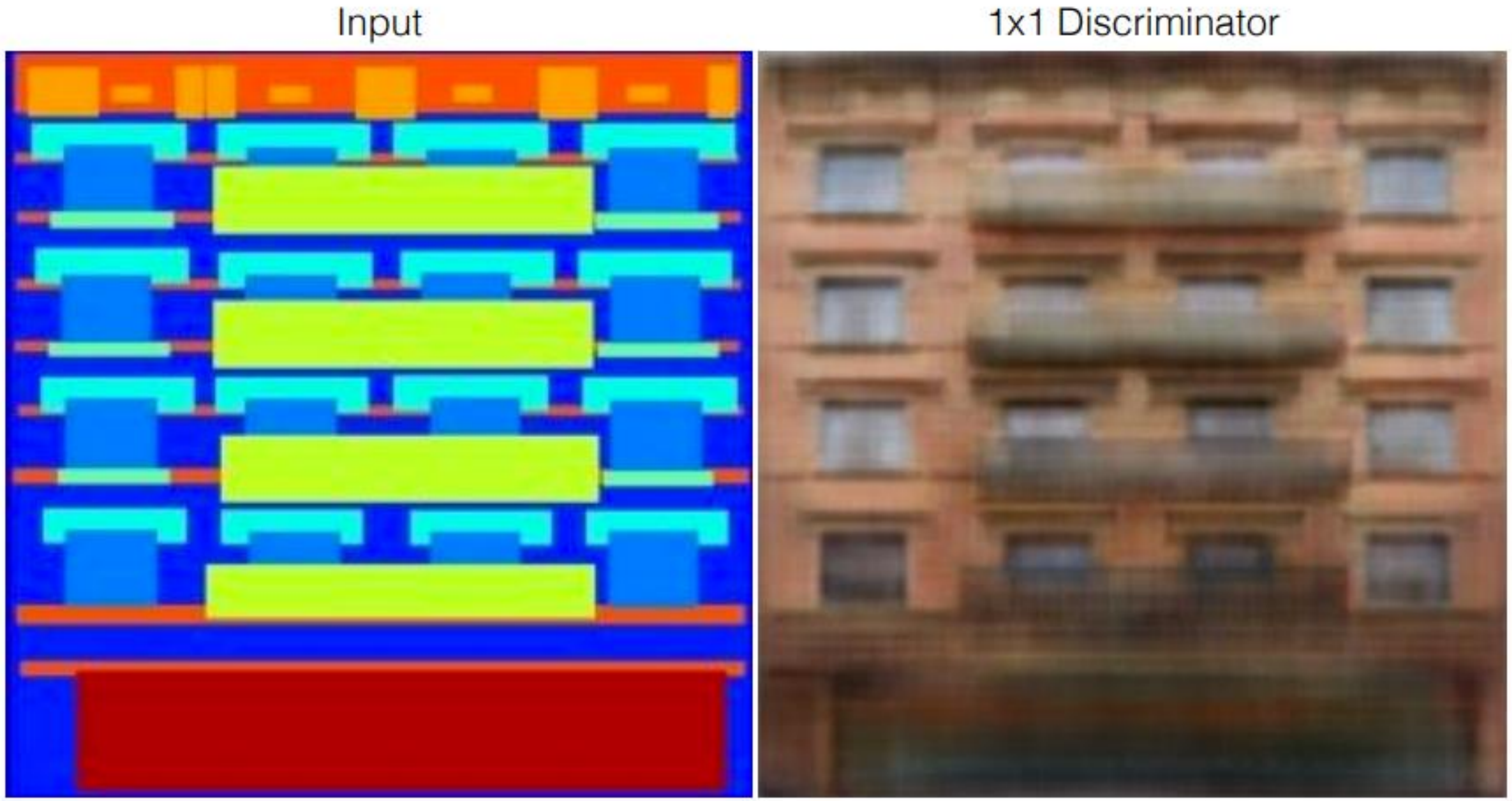
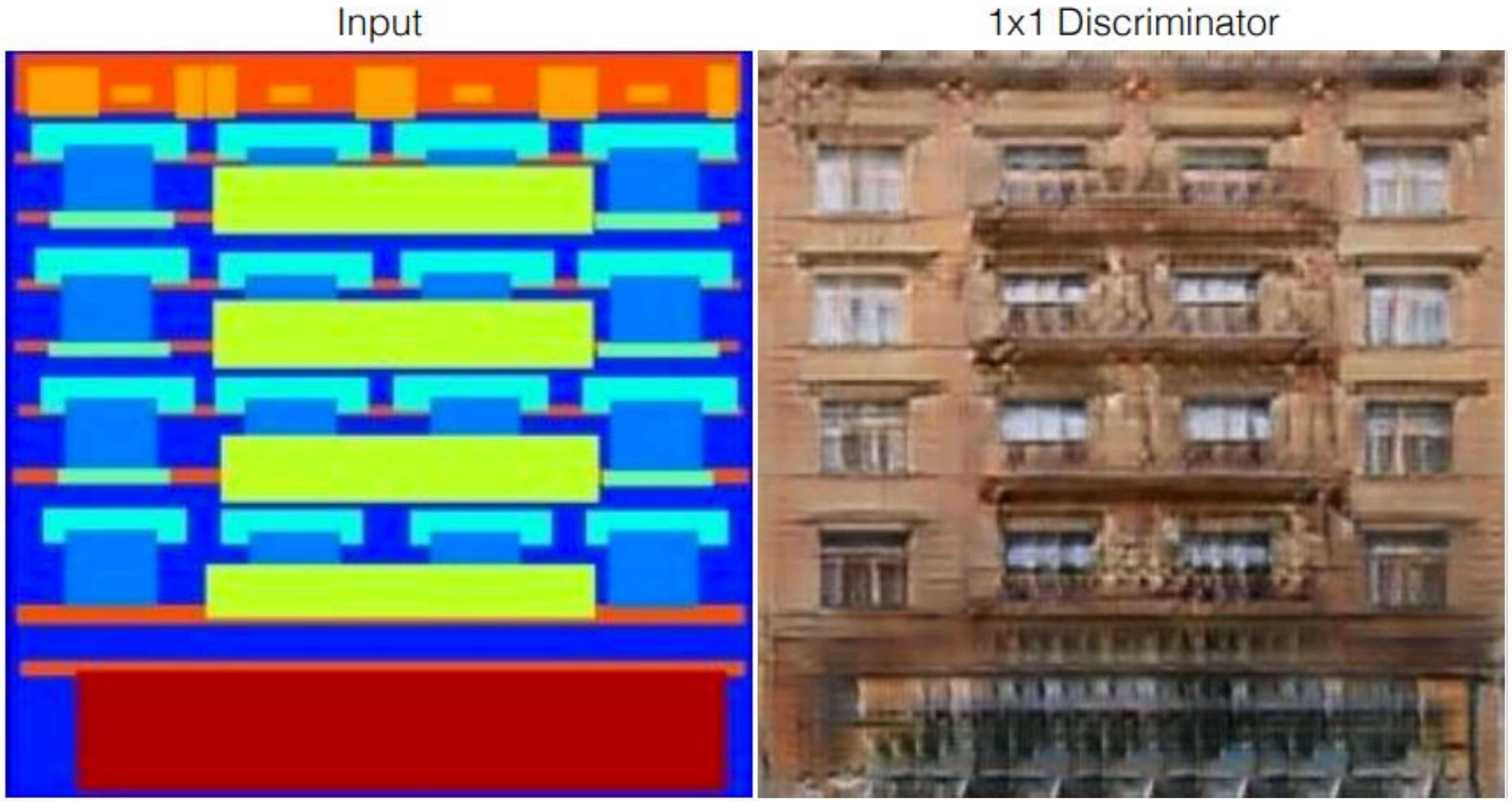
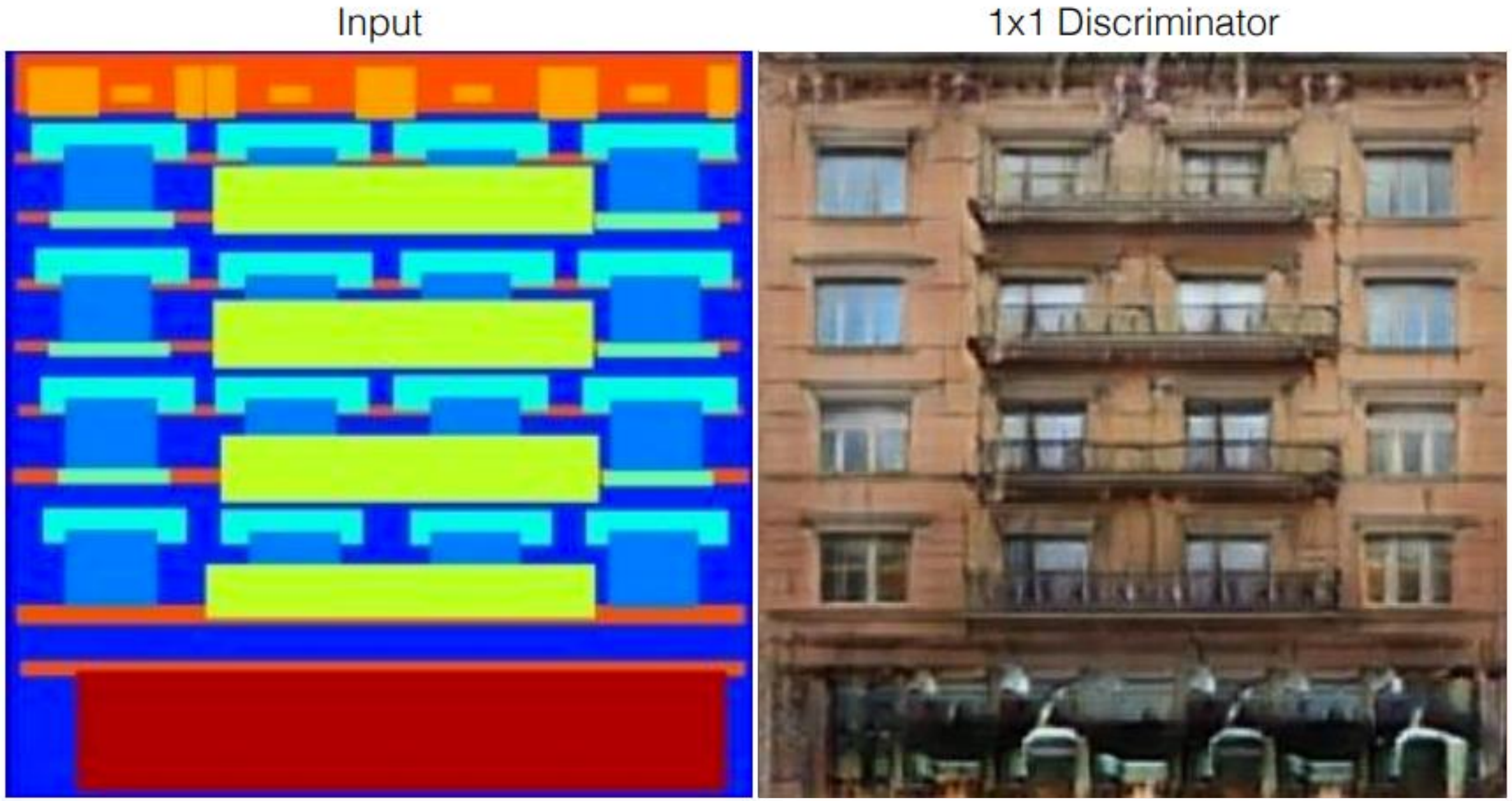


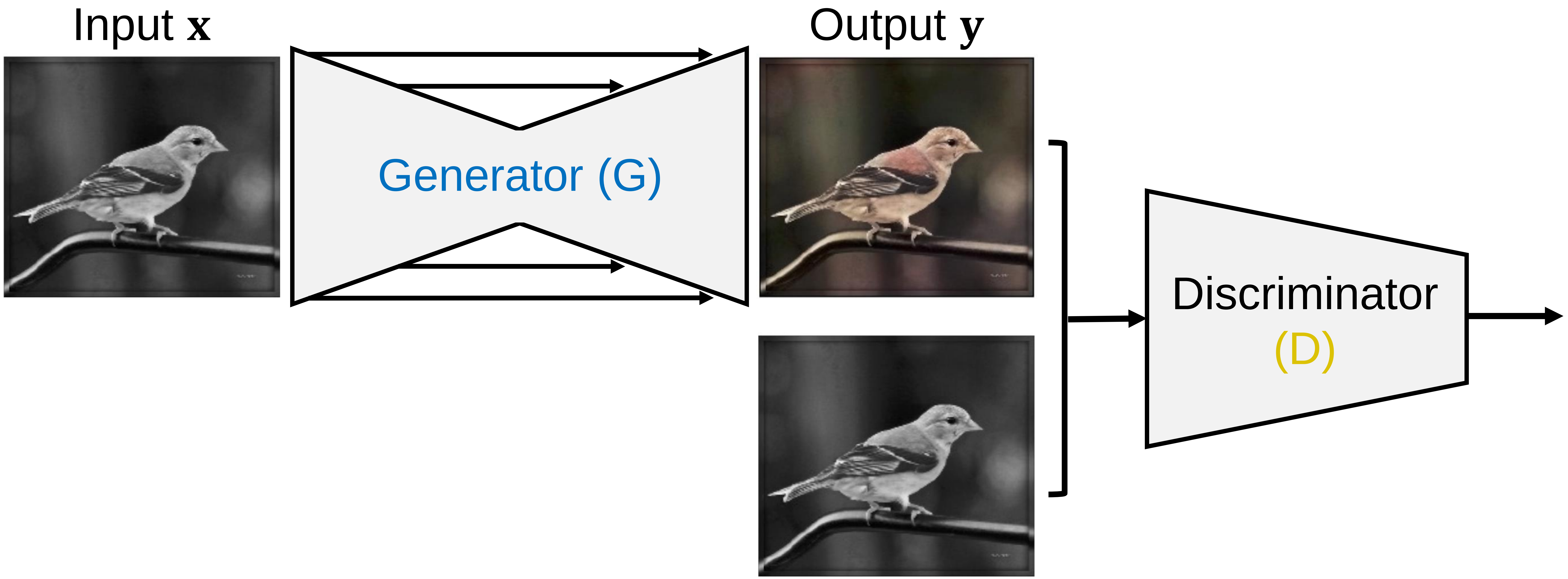
Image Discriminator



70x70 Patch Discriminator

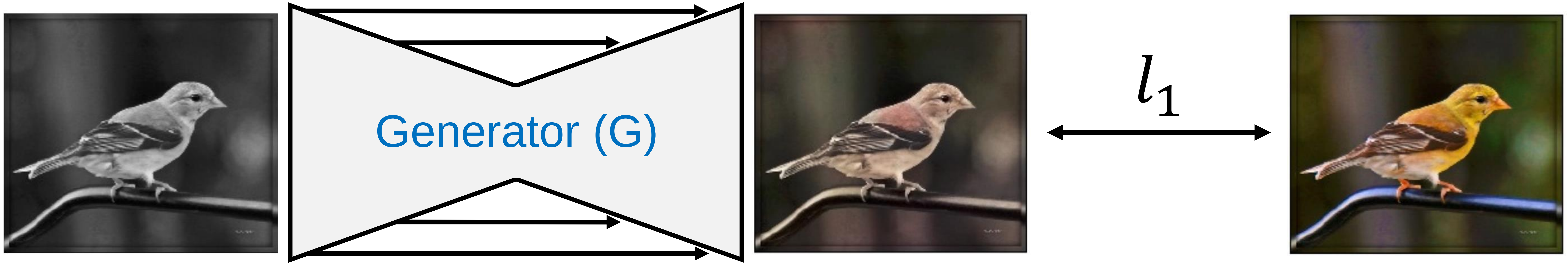


Conditional Discriminator



$$L_{cGAN}(G, D) = \mathbb{E}_{\mathbf{x},\mathbf{y}}[\log D\big(\mathbf{x}, G(\mathbf{x})\big) + \log\big(1 - D(\mathbf{x}, \mathbf{y})\big)]$$

Reconstruction Loss



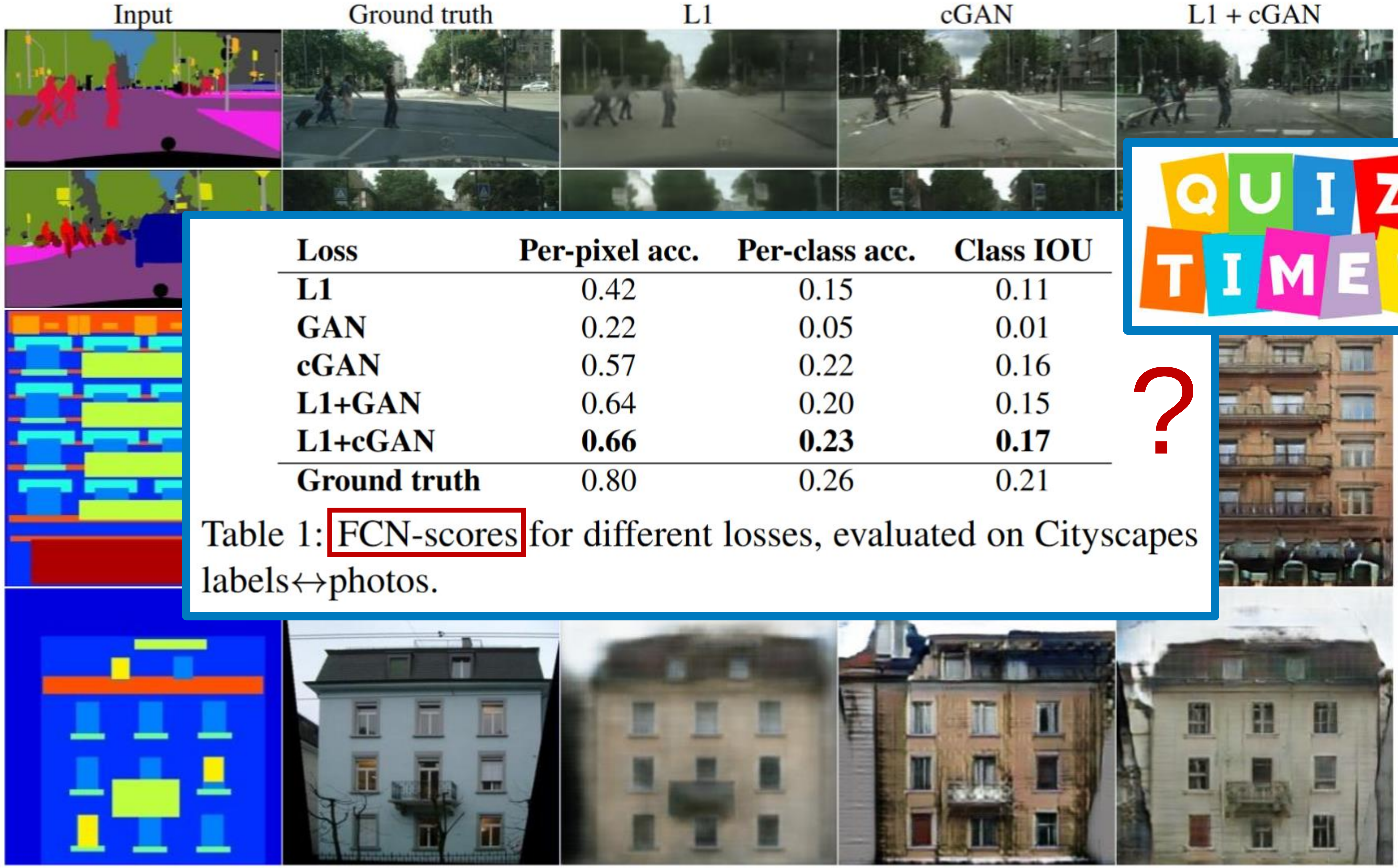
$$L_{l_1}(\mathrm{G}) = \mathbb{E}_{\mathbf{x},\mathbf{y}} \, \|\mathrm{G}(\mathrm{x}) - \mathrm{y}\|_1$$

"Stable training + fast convergence"

$$G^* = \arg\min_{\mathrm{G}} \max_{\mathrm{D}} L_{cGAN}(\mathrm{G}, \mathrm{D}) + \lambda \, L_{l_1}(\mathrm{G})$$

λ: 100

Ablation Study

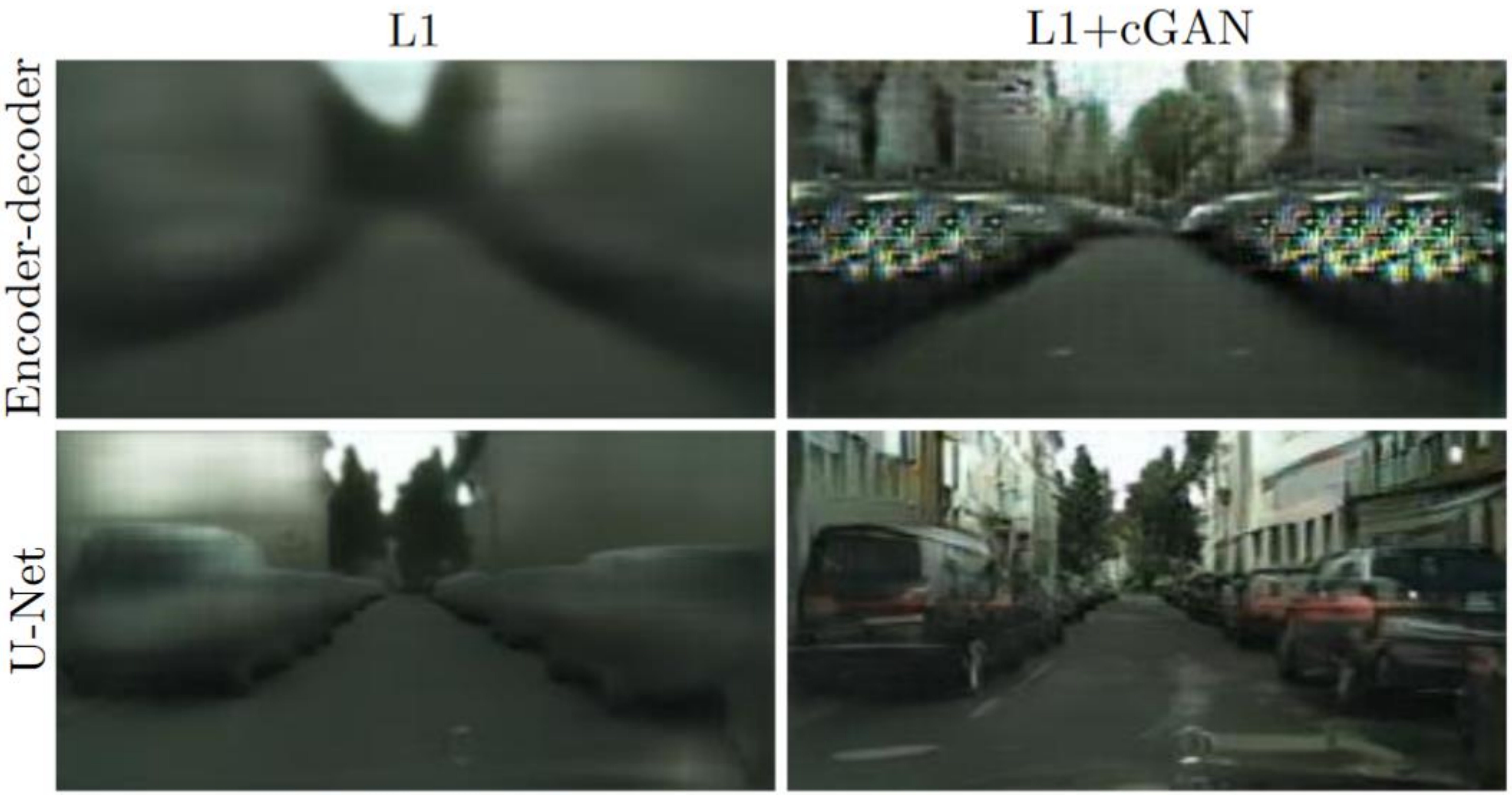


Loss	Per-pixel acc.	Per-class acc.	Class IOU
L1	0.42	0.15	0.11
GAN	0.22	0.05	0.01
cGAN	0.57	0.22	0.16
L1+GAN	0.64	0.20	0.15
L1+cGAN	**0.66**	**0.23**	**0.17**
Ground truth	0.80	0.26	0.21

Table 1: FCN-scores for different losses, evaluated on Cityscapes labels↔photos.

?

Ablation Study

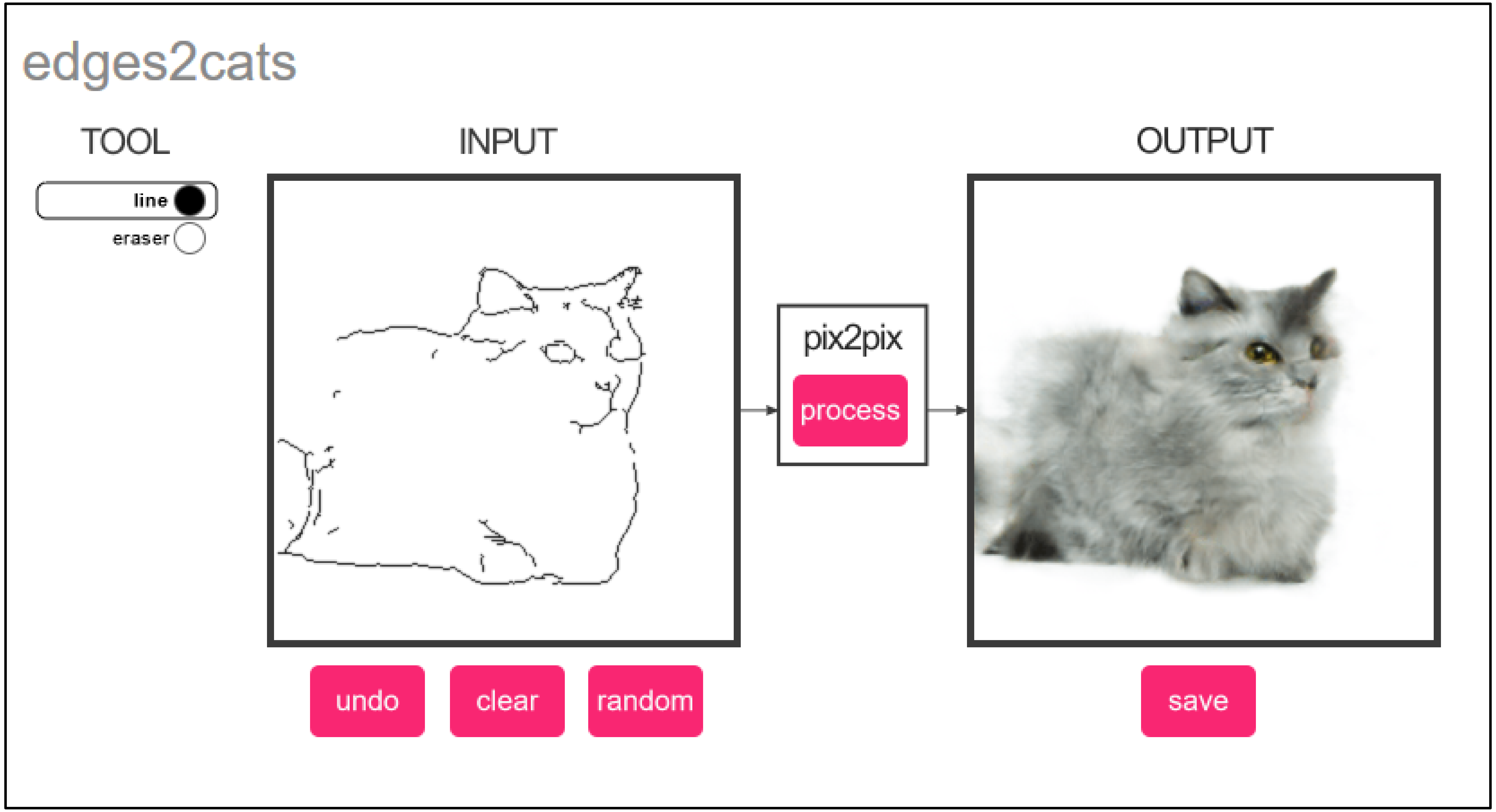


Results on the Test Split

Results for Hand Drawings

?

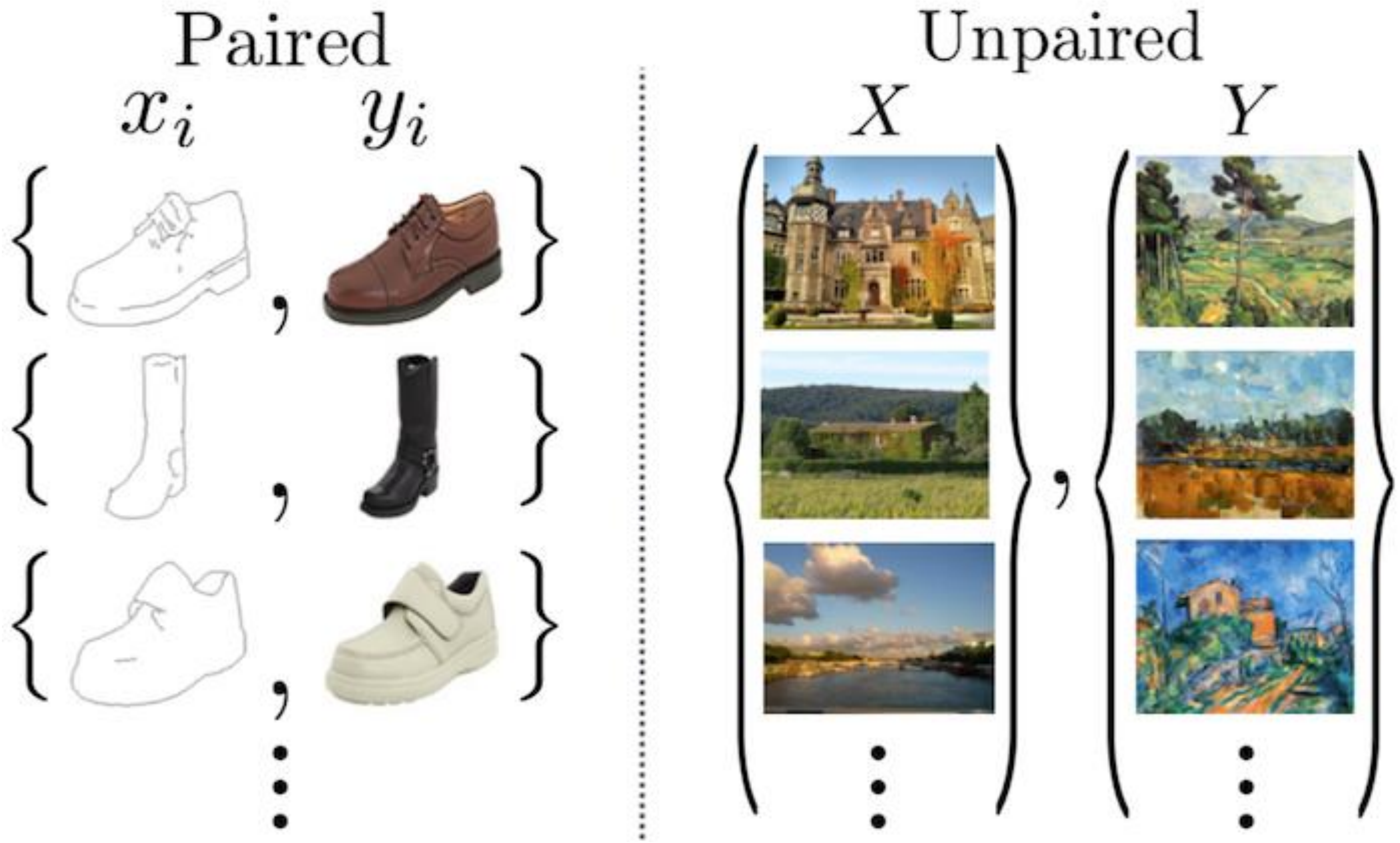
Demo: Pix2Pix



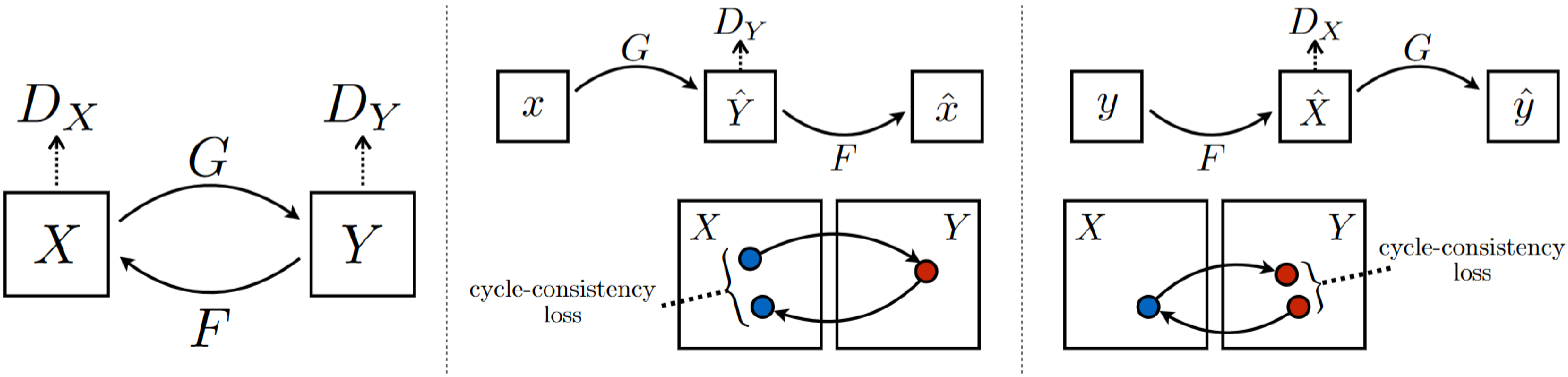
Limitations

1. Paired data is required

CycleGAN



Cycle Consistency



CycleGAN

Recycle-GAN

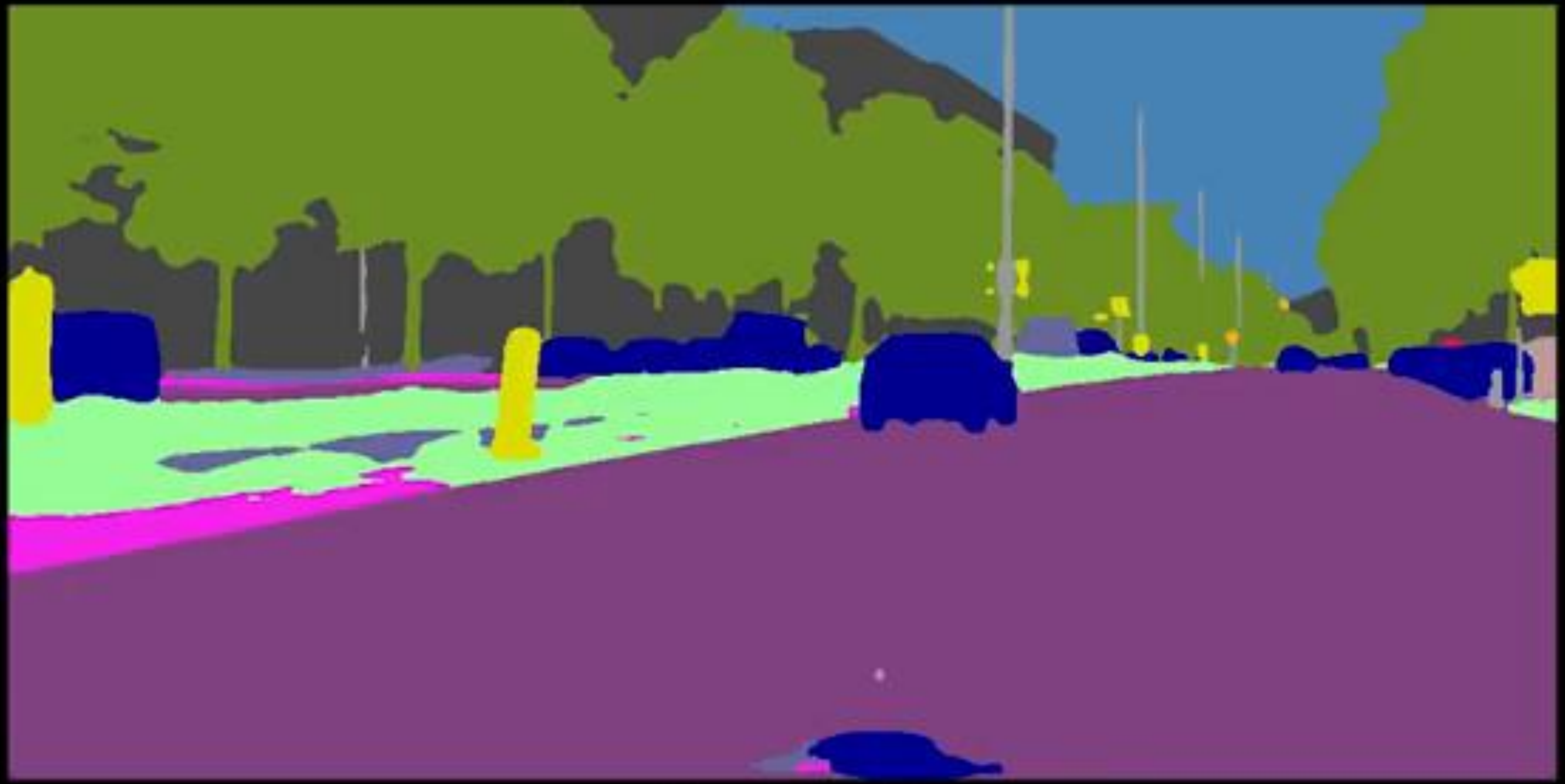
Limitations

1. Paired data is required

Limitations

1. Paired data is required

2. Temporally instable if applied per-frame to a video sequence


Labels


pix2pixHD


COVST


Ours

Video-to-Video Synthesis

Ting-Chun Wang[1], Ming-Yu Liu[1], Jun-Yan Zhu[2], Guilin Liu[1], Andrew Tao[1], Jan Kautz[1], Bryan Catanzaro[1]

[1]NVIDIA Corporation [2]MIT

Limitations

1. Paired data is required

2. Temporally instable if applied per-frame to a video sequence

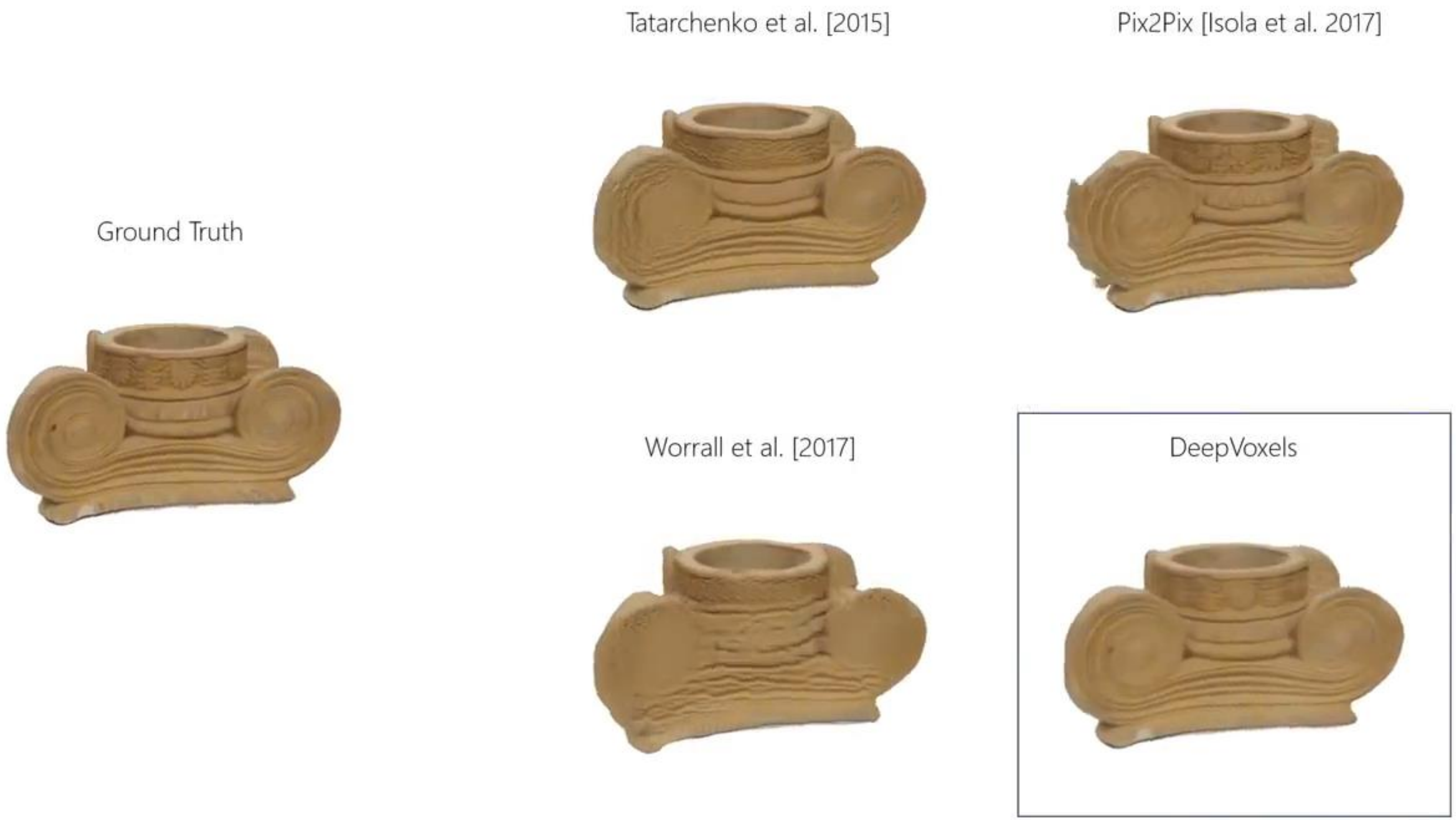
Limitations

1. Paired data is required

2. Temporally instable if applied per-frame to a video sequence

3. Does not generalize to 3D transformations

DeepVoxels


Tatarchenko et al. [2015]
Pix2Pix [Isola et al. 2017]
Ground Truth
Worrall et al. [2017]
DeepVoxels

Summary

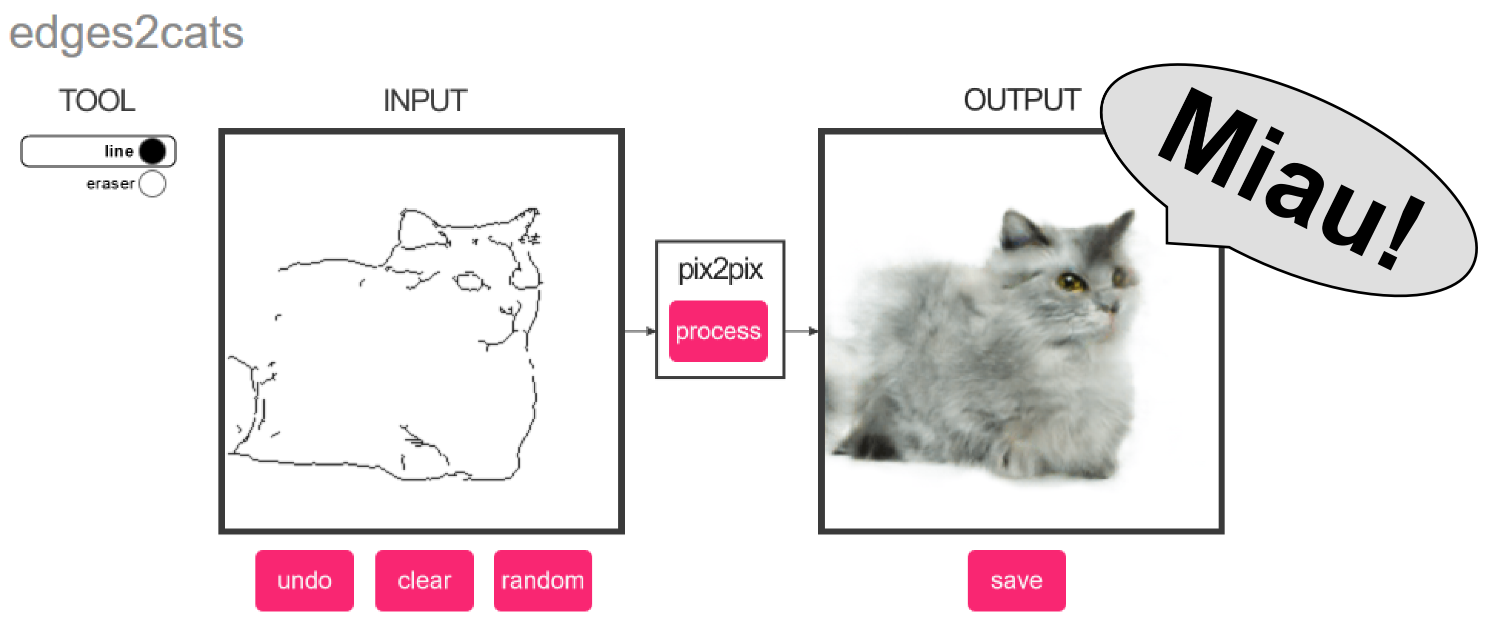
- Convolutional Neural Networks

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau.$$

- Generative Modeling

(Brundage et al, 2018)

- Pix2Pix (“mapping from A to B”)



References

- CVPR GAN Tutorial
  - https://sites.google.com/view/cvpr2018tutorialongans
- CS231n
  - http://cs231n.stanford.edu/slides/2016/winter1516_lecture7.pdf
- …