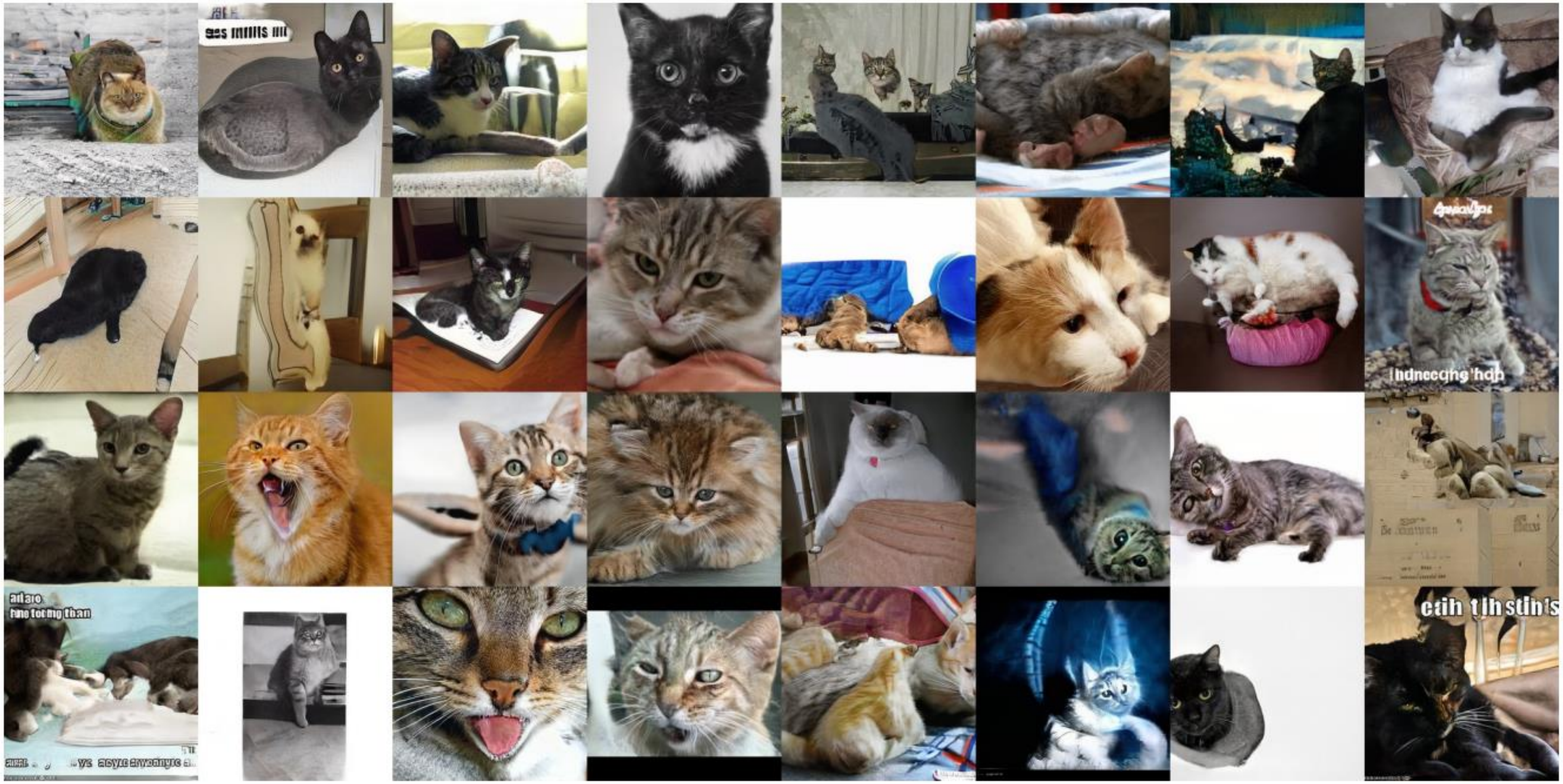


Perceptual Loss, GANs (part I)

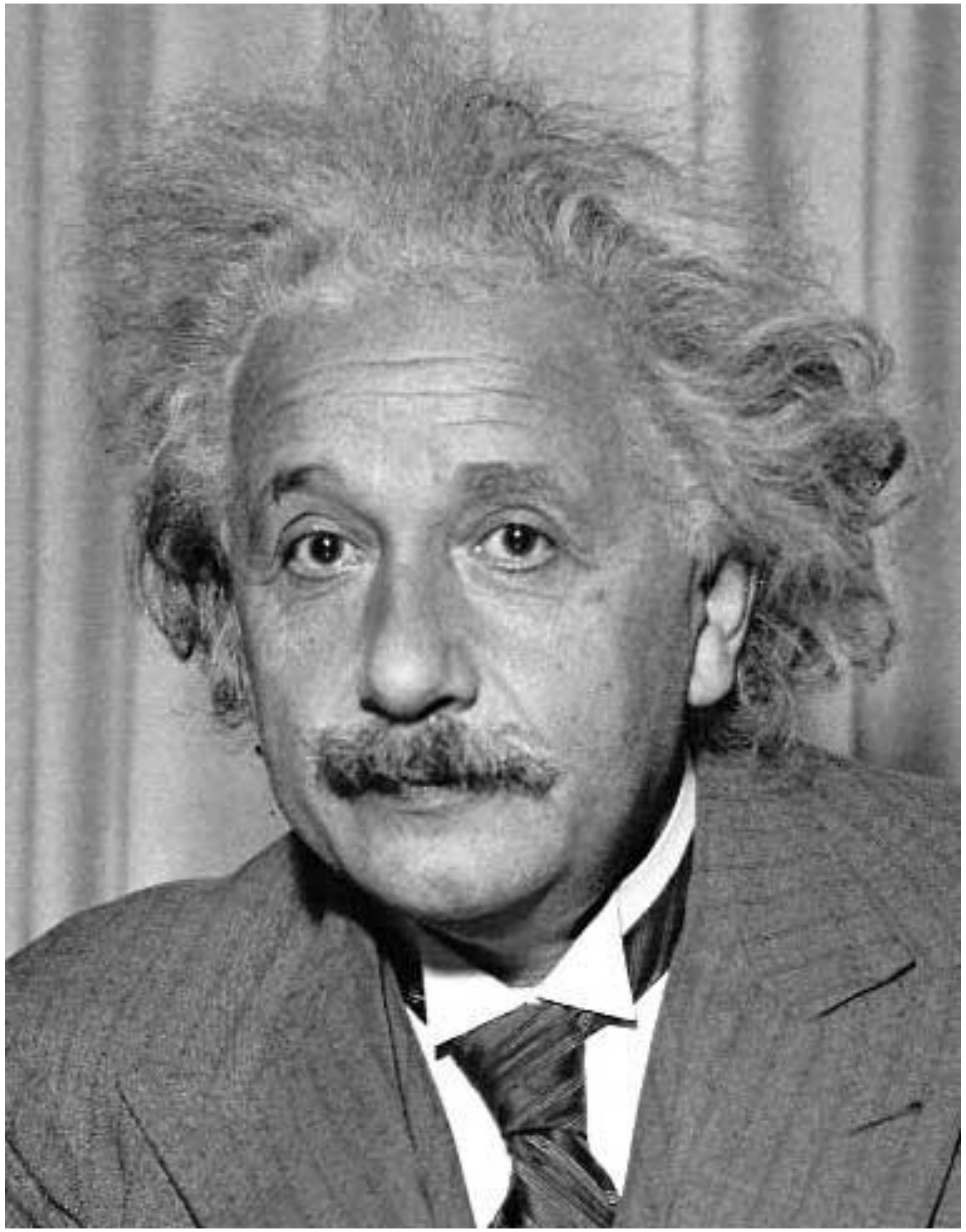
Jun-Yan Zhu

16-726 Learning-based Image Synthesis, Spring 2025

many slides from Alyosha Efros, Phillip Isola, Richard Zhang, James Hays, and Andrea Vedaldi, Jitendra Malik.

HW1 (hints)

Template matching



- Goal: find  in image
- Main challenge: What is a good similarity or distance measure between two patches?
  - Correlation
  - Zero-mean correlation
  - Sum Square Difference
  - Normalized Cross Correlation

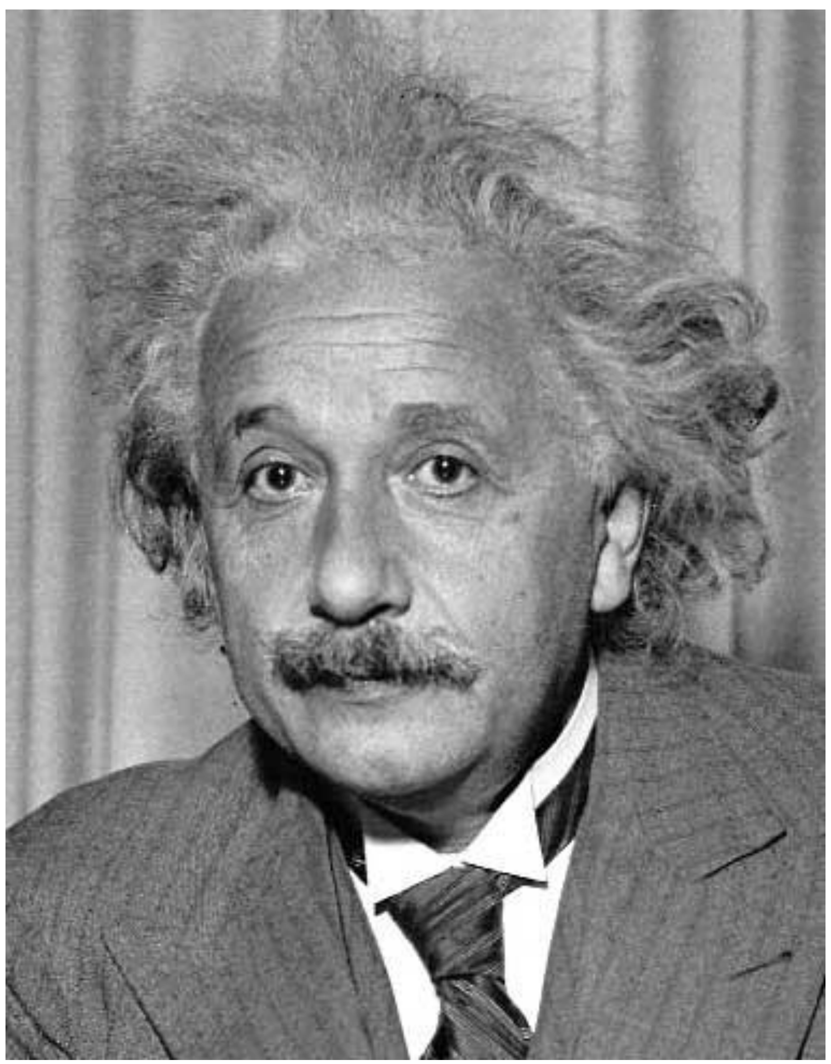
Side by Derek Hoiem

Matching with filters

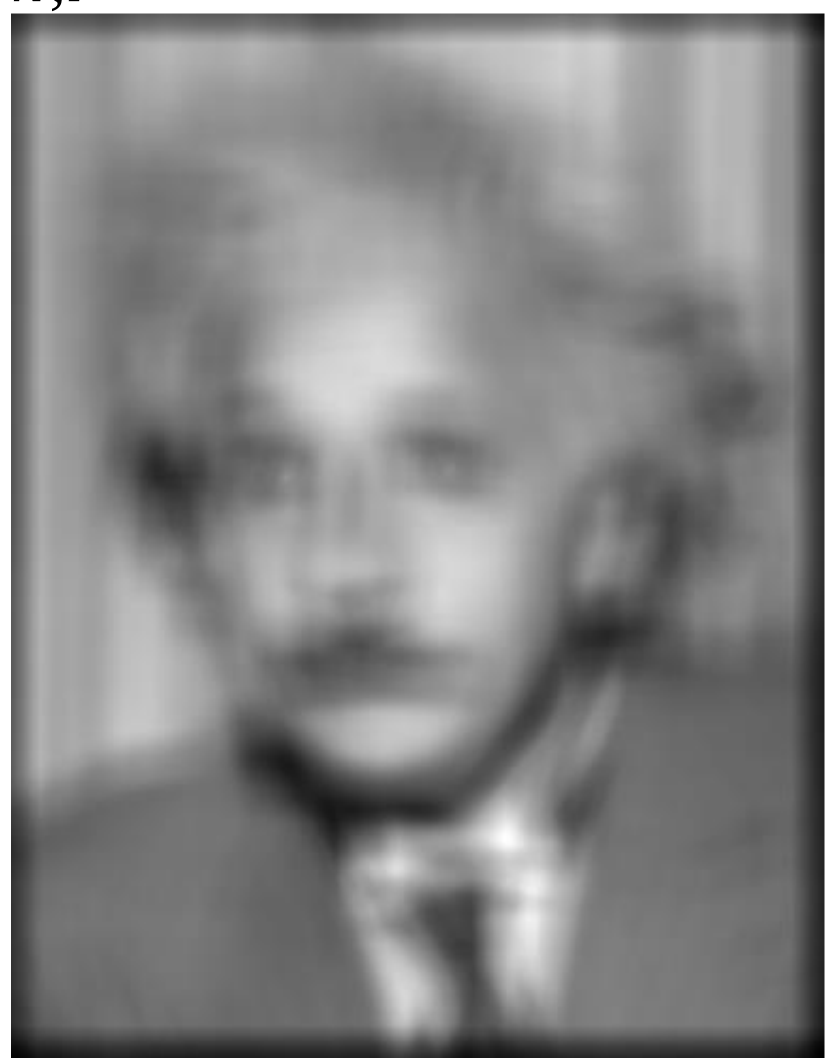
- Goal: find  in image
- Method 0: filter the image with eye patch

$$h[m,n]=\sum_{k,l} g[k,l]\ f[m+k,n+l]$$

f = image
g = filter



Input



Filtered Image

What went wrong?

Side by Derek Hoiem

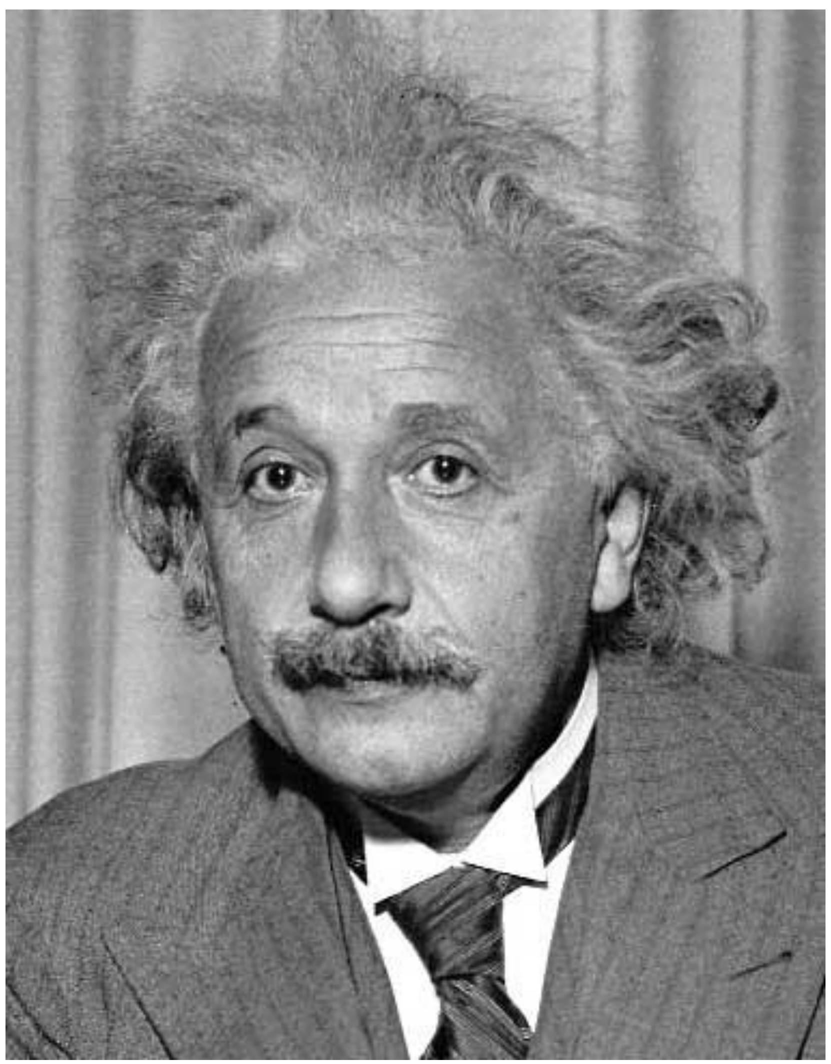
Matching with filters

- Goal: find  in image
- Method 1: filter the image with zero-mean eye

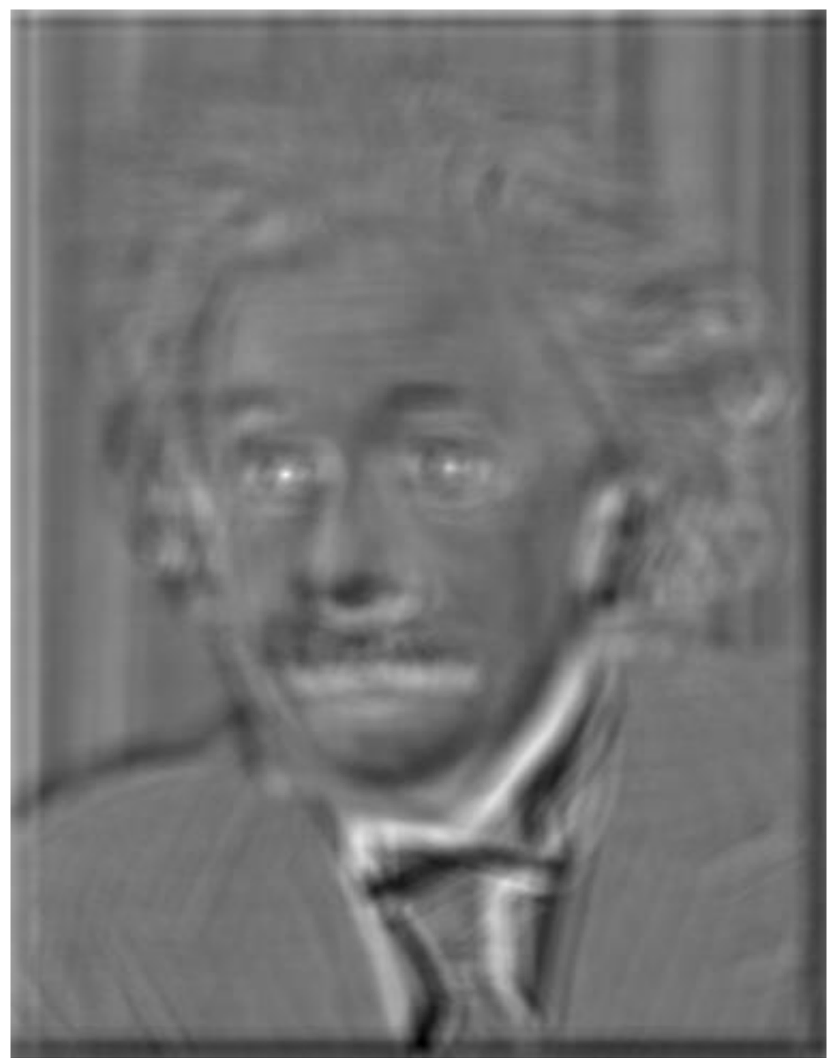
$$h[m,n] = \sum_{k,l} (g[k,l] - \bar{g})(f[m+k, n+l])$$

mean of g

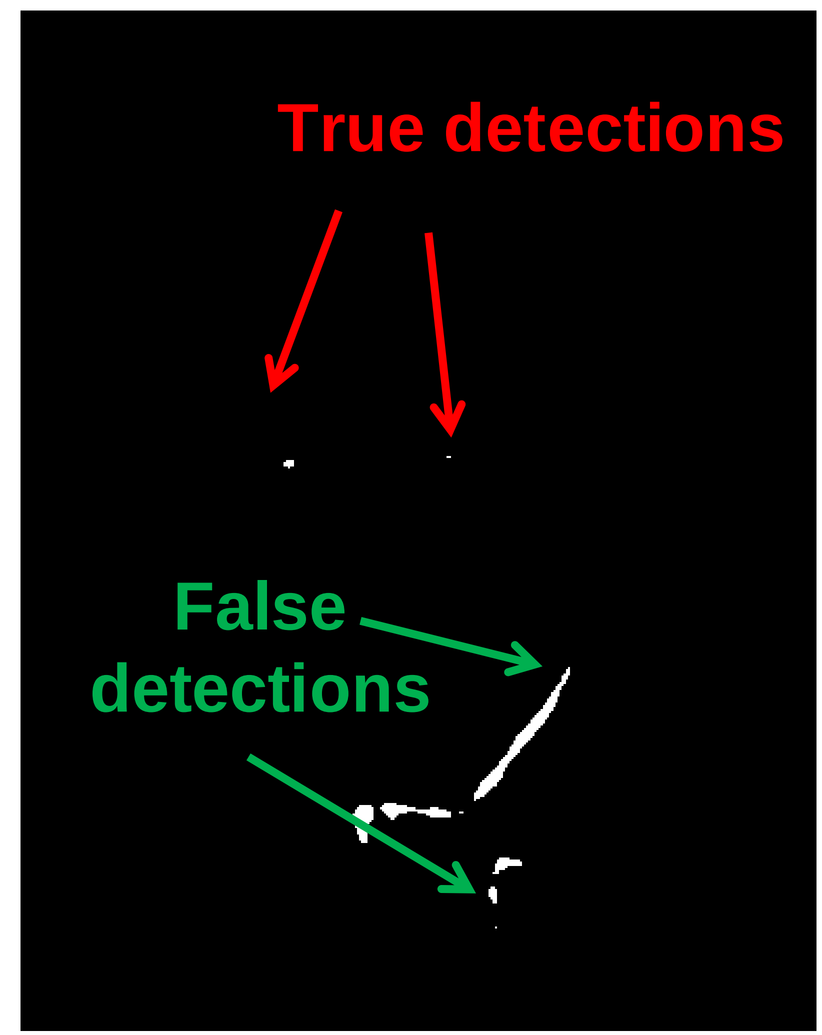
f = image
g = filter



Input



Filtered Image (scaled)



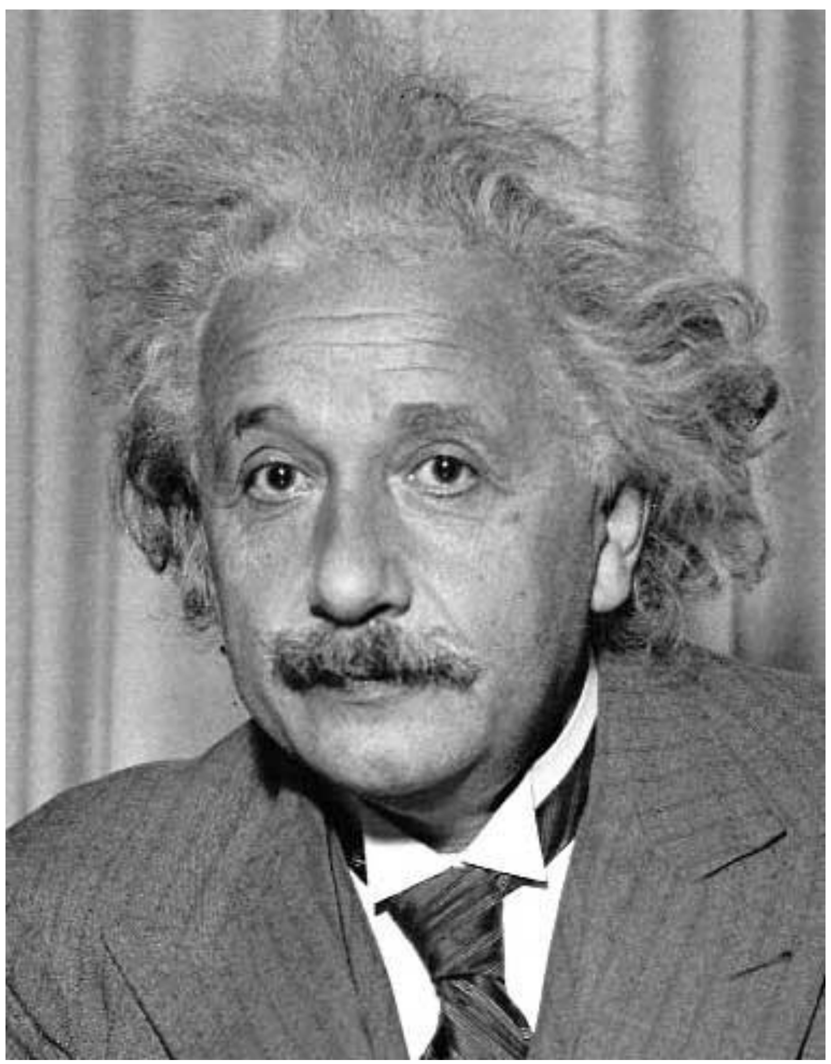
Thresholded Image

Matching with filters

- Goal: find  in image
- Method 2: SSD (Sum Square Difference)

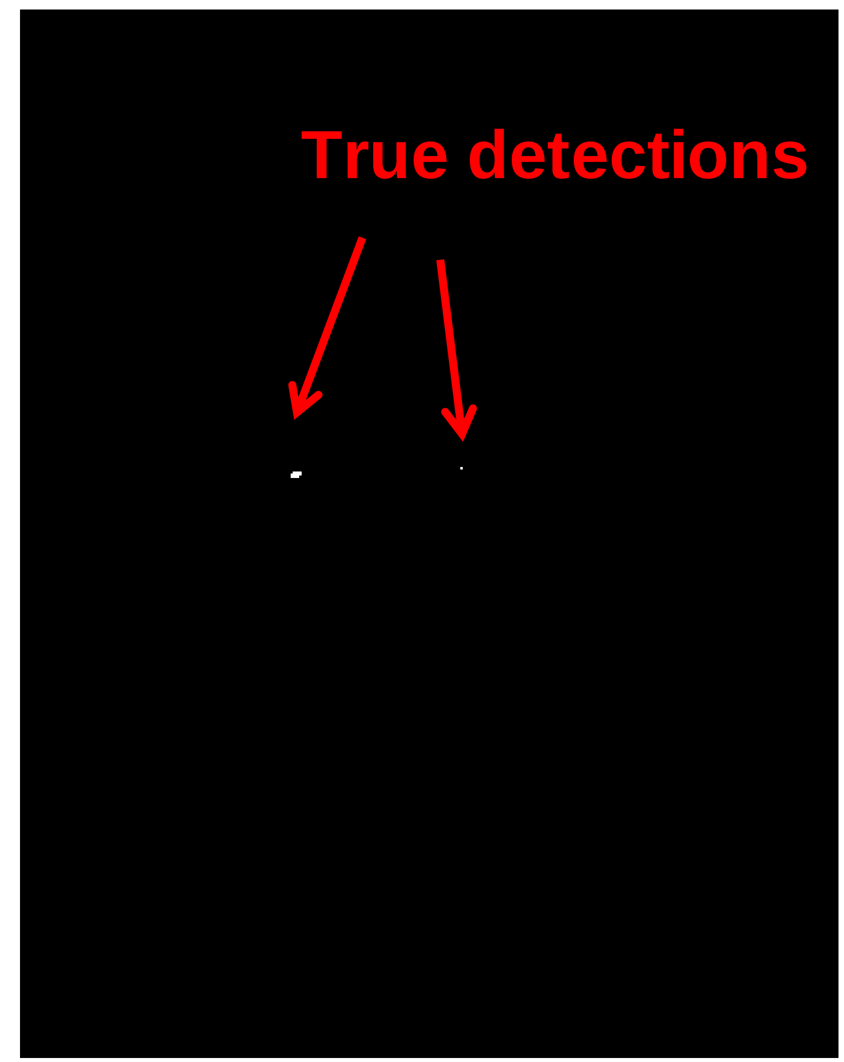
$$h[m,n]=\sum_{k,l}(g[k,l]-f[m+k,n+l])^2$$

f = image
g = filter



Input

1- sqrt(SSD)



Thresholded Image

Matching with filters

$$h[m,n] = \sum_{k,l} (g[k,l] - f[m+k, n+l])^2$$

f = image
g = filter

- Can SSD be implemented with linear filters?

Side by Derek Hoiem

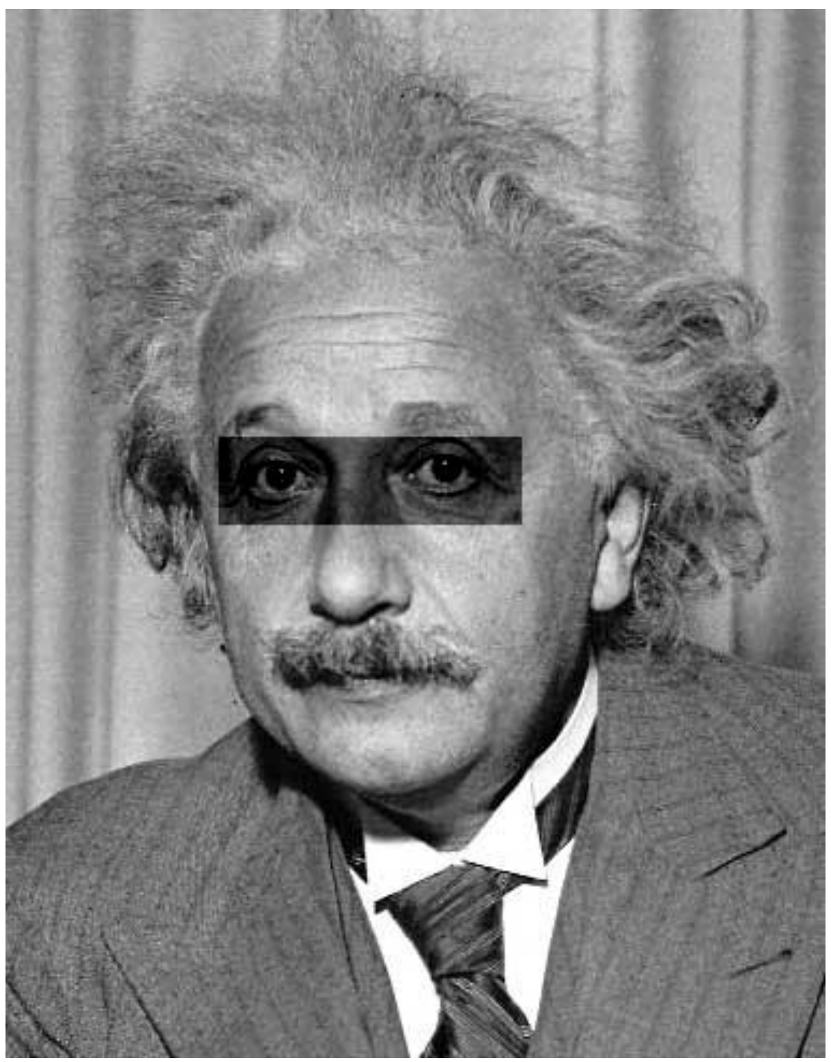
Matching with filters

- Goal: find  in image
- Method 2: SSD (Sum Square Difference)

What's the potential downside of SSD?

$$h[m,n]=\sum_{k,l}(g[k,l]-f[m+k,n+l])^2$$

f = image
g = filter



Input

1- sqrt(SSD)

Matching with filters

- Goal: find  in image
- Method 2: Normalized Cross-Correlation

f = image
g = filter

mean template

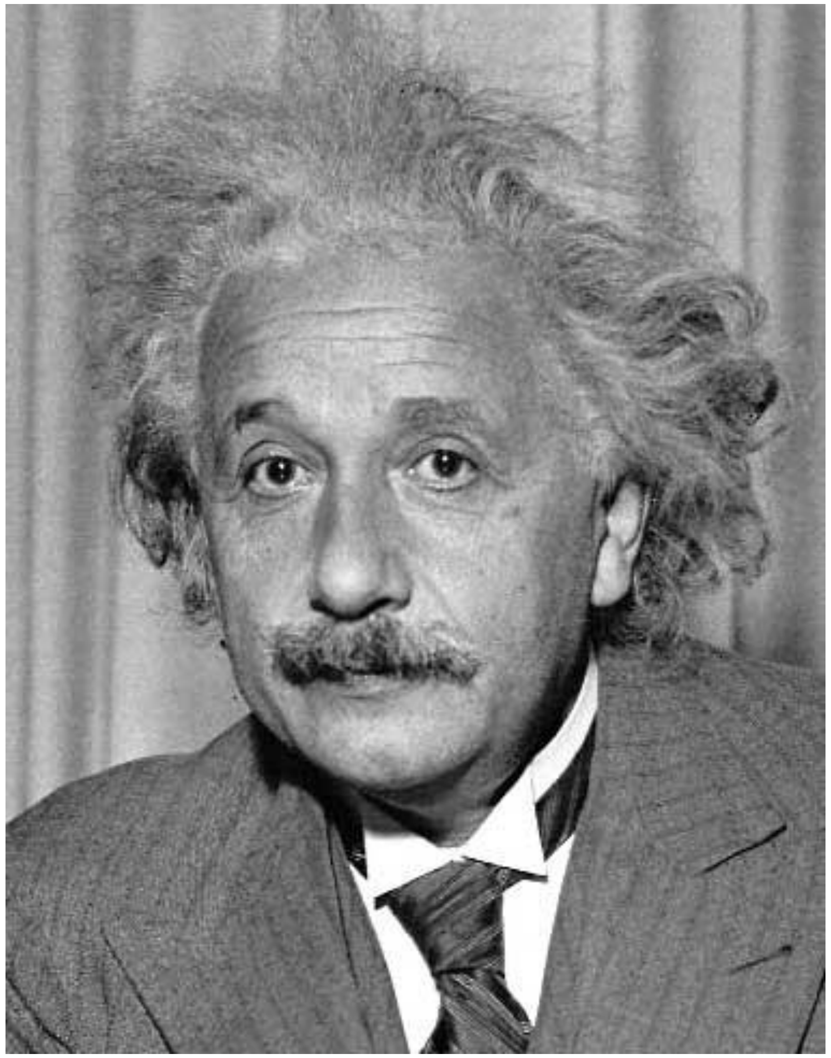
mean image patch

$$h[m,n] = \frac{\sum_{k,l} (g[k,l] - \bar{g})(f[m+k,n+l] - \bar{f}_{m,n})}{\left(\sum_{k,l} (g[k,l] - \bar{g})^2 \sum_{k,l} (f[m+k,n+l] - \bar{f}_{m,n})^2 \right)^{0.5}}$$

Side by Derek Hoiem

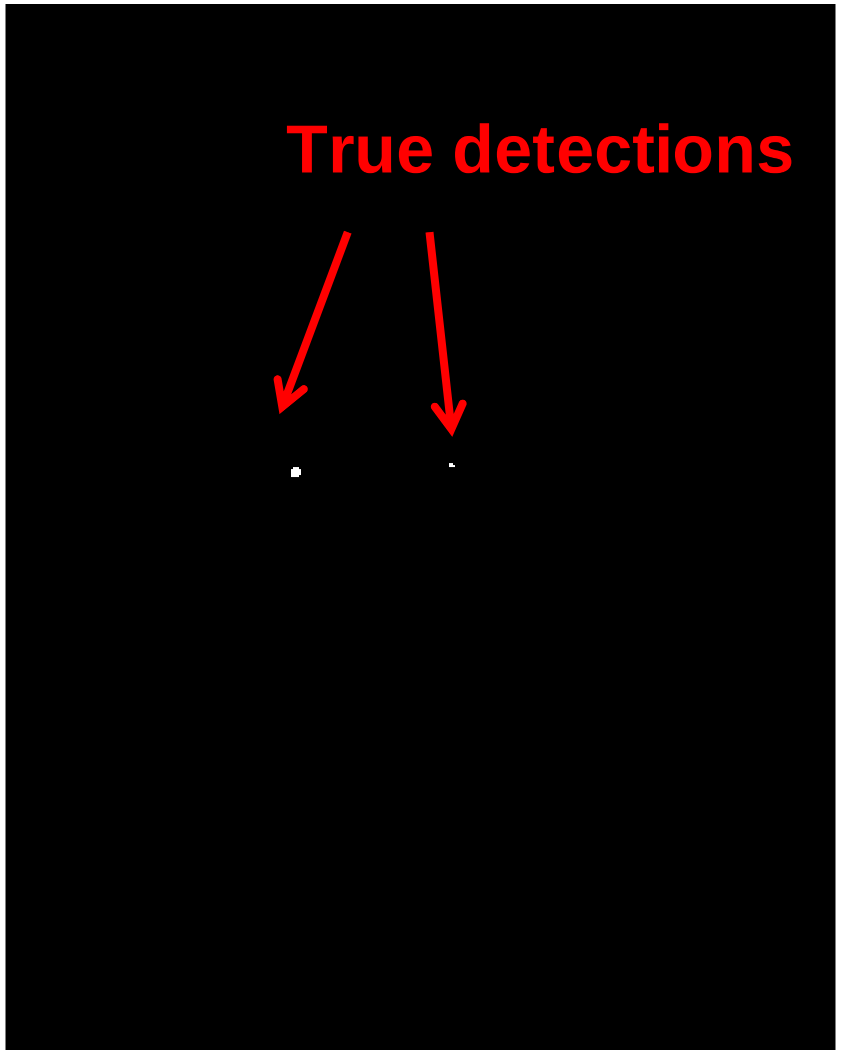
Matching with filters

- Goal: find  in image
- Method 2: Normalized Cross-Correlation



Input

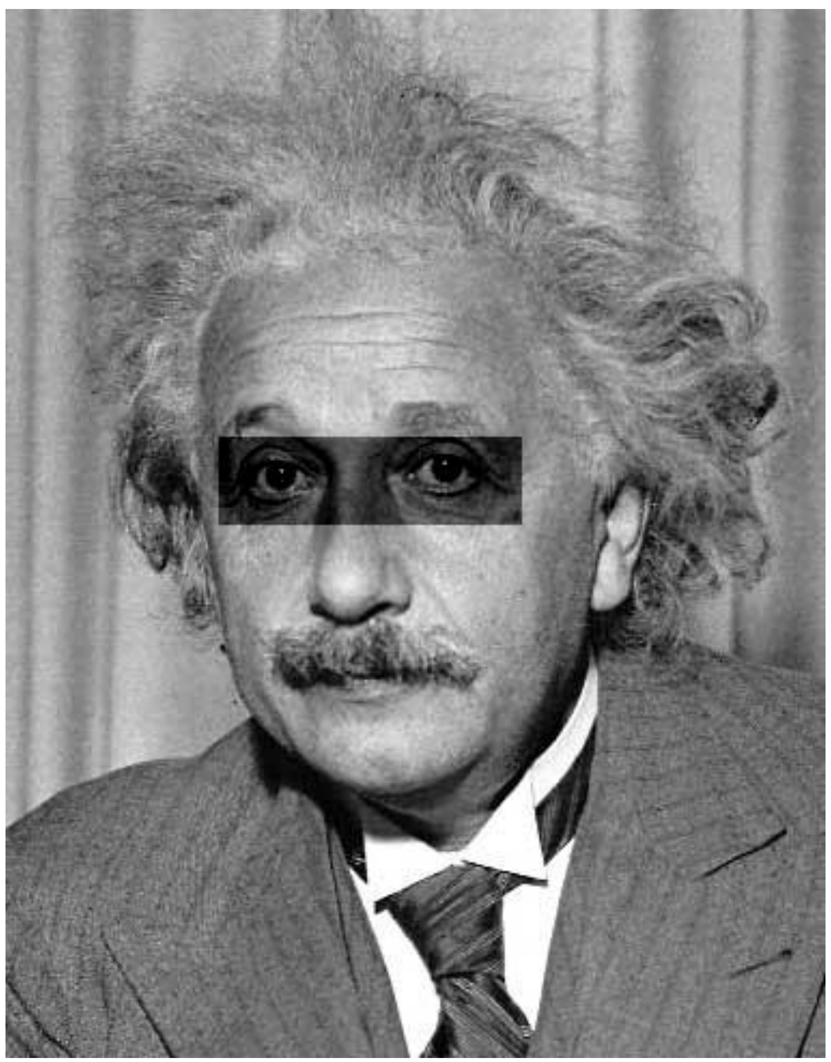
Normalized X-Correlation



Thresholded Image

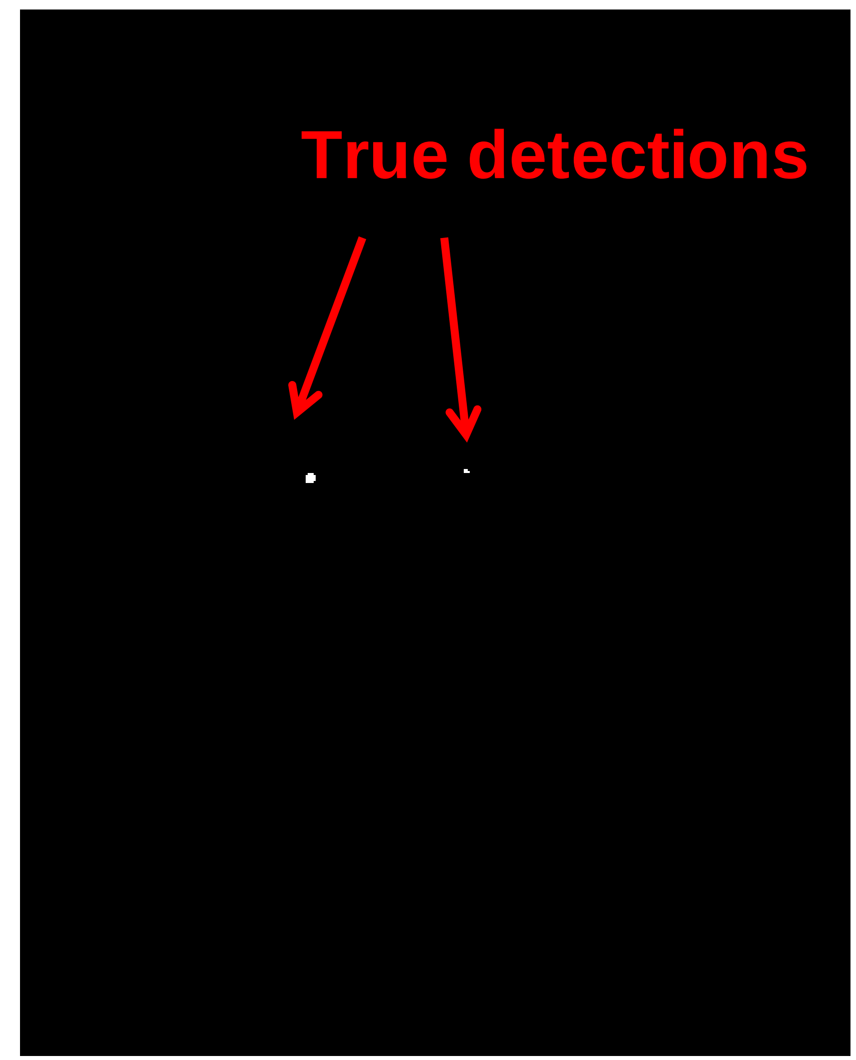
Matching with filters

- Goal: find  in image
- Method 2: Normalized Cross-Correlation



Input

Normalized X-Correlation



Thresholded Image

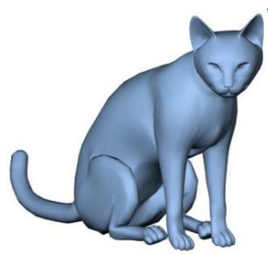
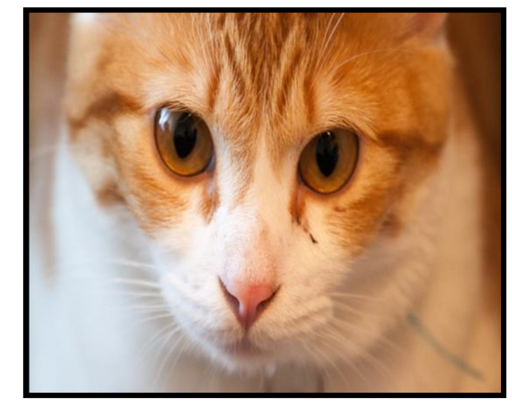
Q: What is the best method to use?

- Answer: Depends
- Zero-mean filter: fastest but not a great matcher
- SSD: next fastest, sensitive to overall intensity
- Normalized cross-correlation: slowest, invariant to local average intensity and contrast

Side by Derek Hoiem

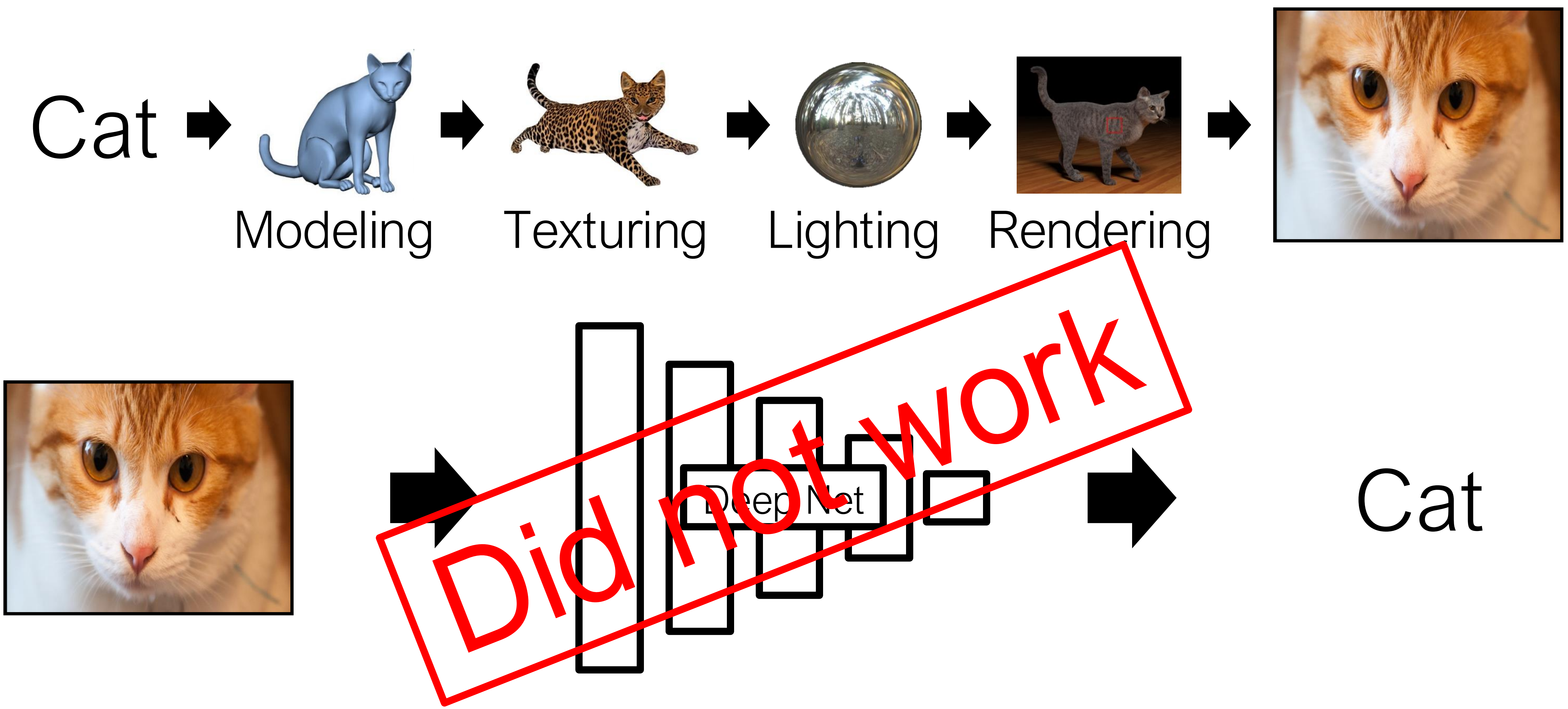
Review
(CNN for Image Synthesis)

Can Deep Learning **Help** Graphics?

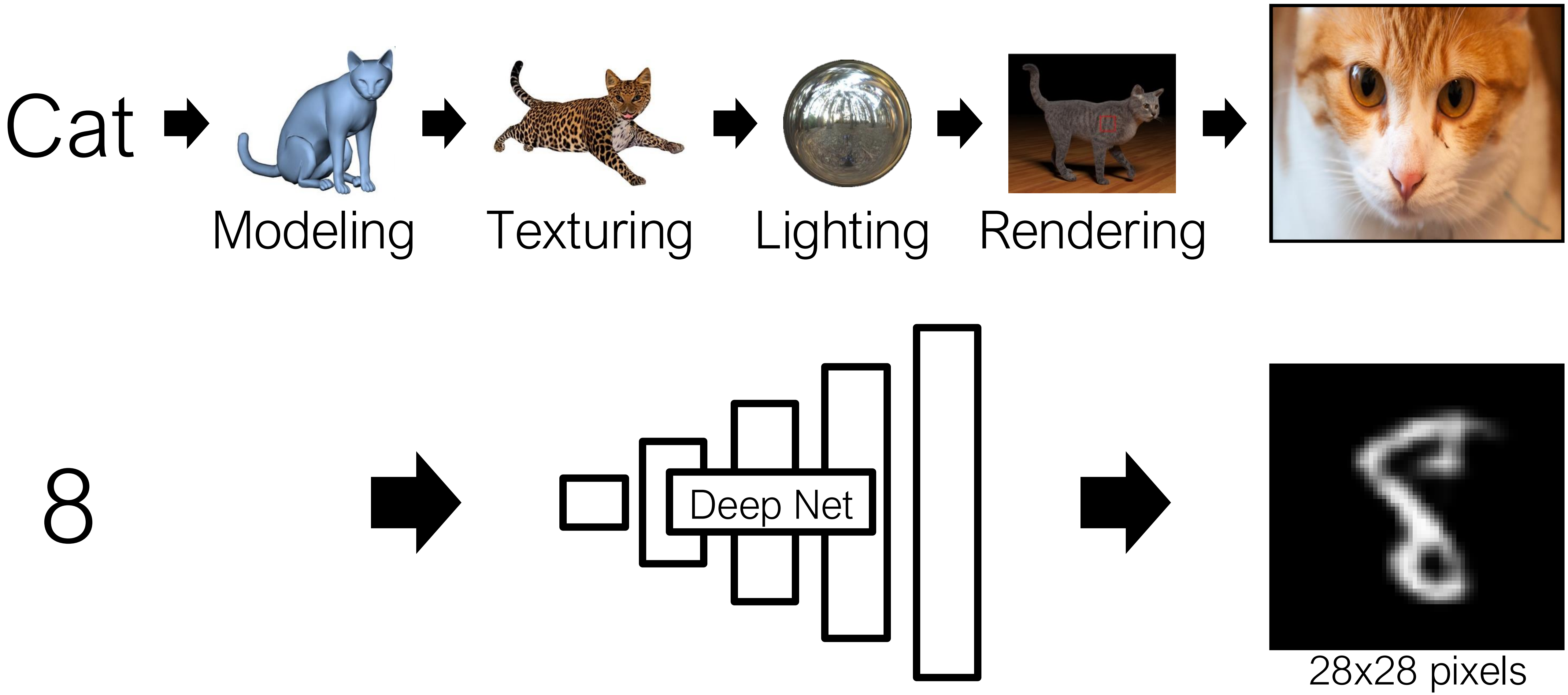
Cat ➡  ➡  ➡  ➡  ➡ 

Modeling Texturing Lighting Rendering

Can Deep Learning **Help** Graphics?

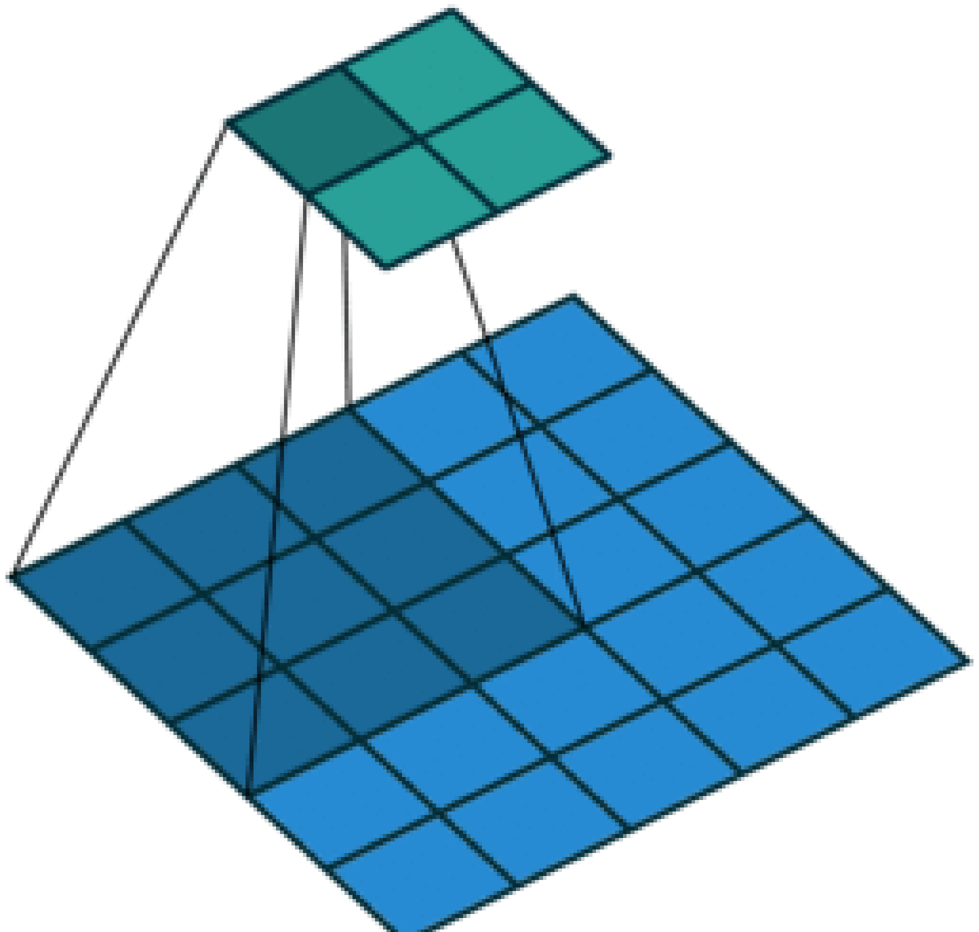


Generating images is hard!

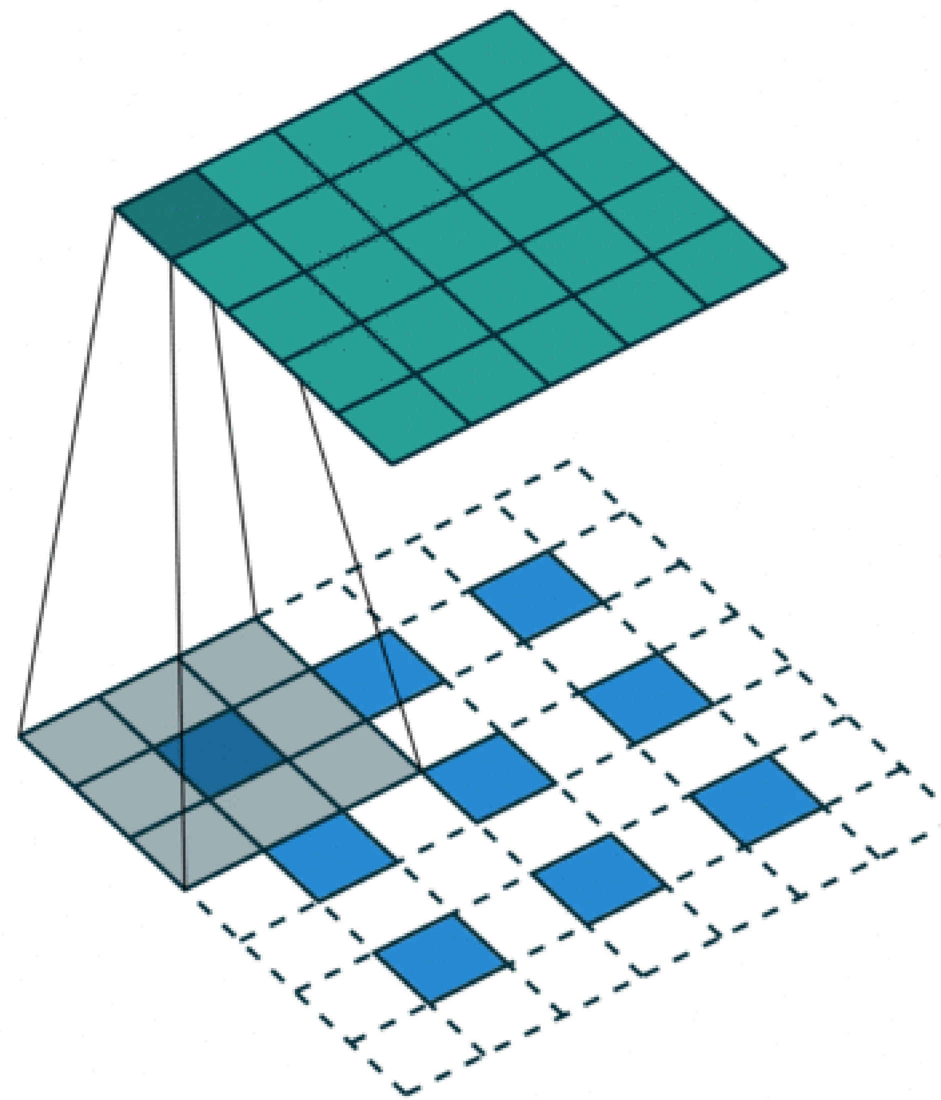


Better Architectures

Fractionally-strided Convolution



Regular conv (no padding)



Fractiaionally-strided conv

Better Loss Functions

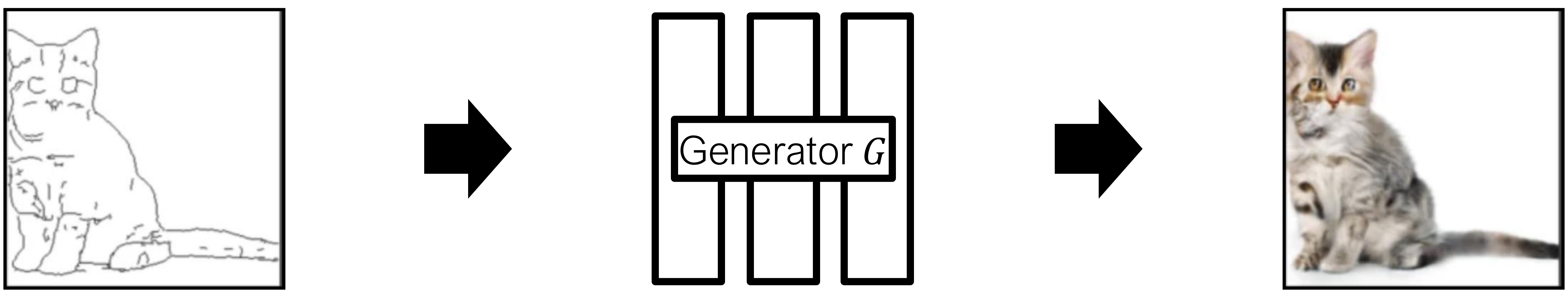
Simple L2 regression doesn't work ☹

Input

Output

Ground truth

Loss functions for Image Synthesis



Input $\mathbf{x}$ Learnable rendering Output Image $\mathbf{G}(x)$

What is a good objective $\mathcal{L}$?

- Capture realism
- Calculate image distance
- Adapt to new tasks/data.

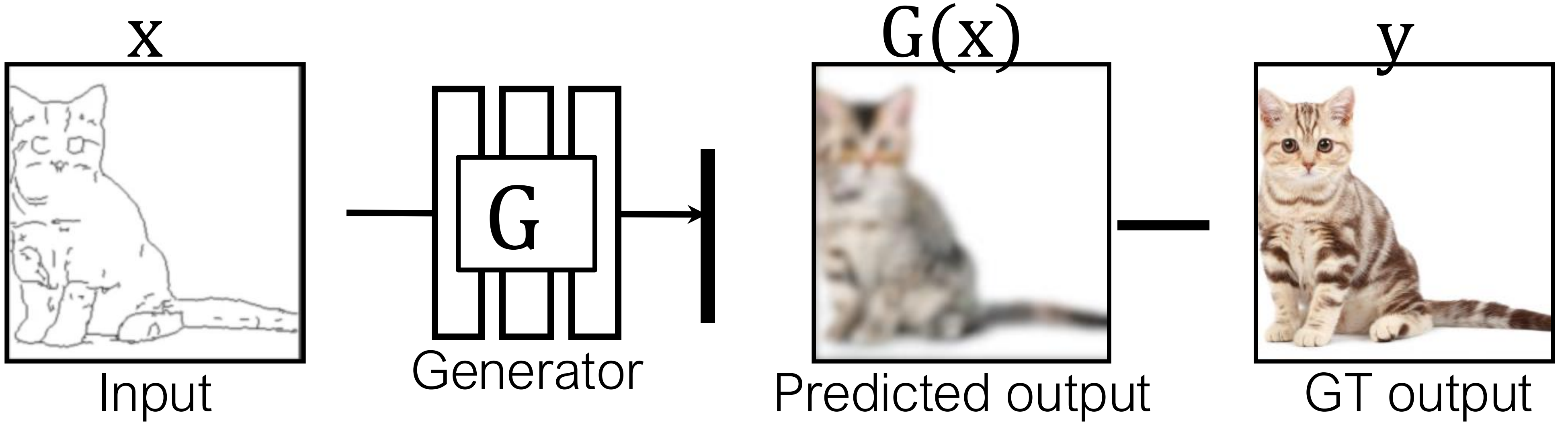
Problem Statement

Loss function

$$\arg\min_{G} \underline{\mathcal{L}}(\underline{G}(\underline{x}), \underline{y})$$

Generator Input Output image

Designing Loss Functions

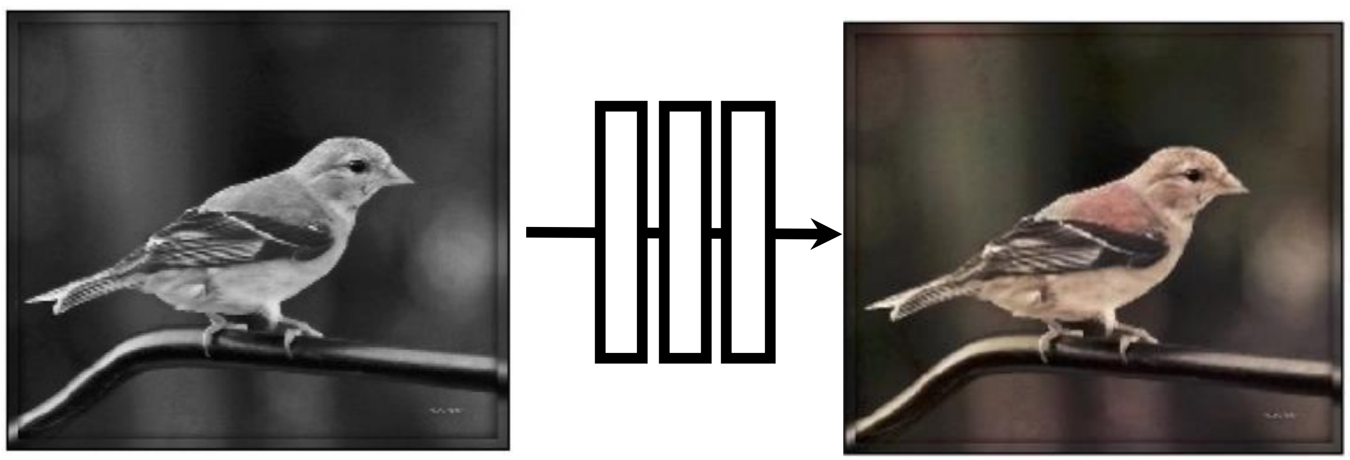


L2 regression

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

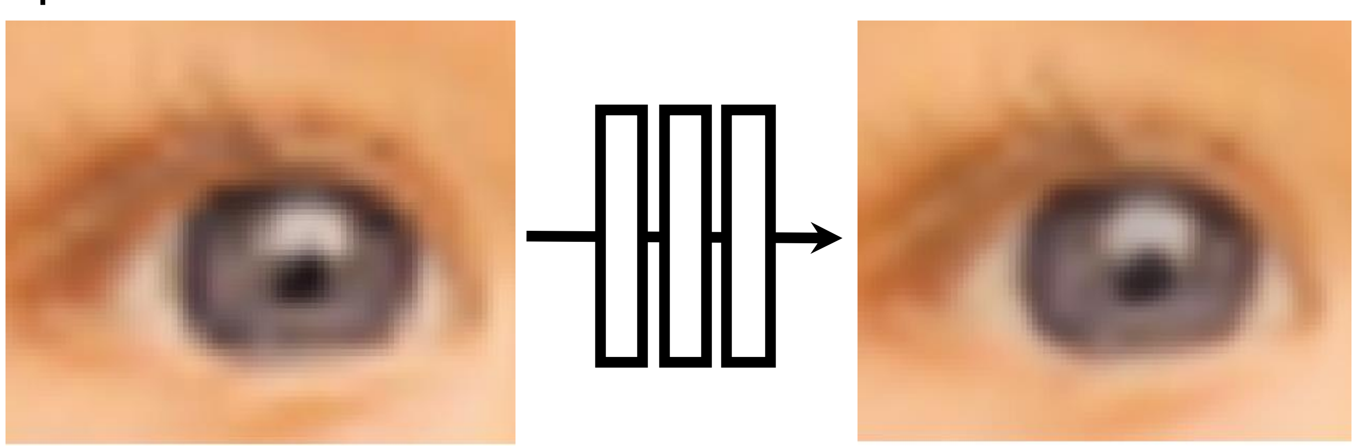
Designing Loss Functions

Image colorization



L2 regression

Super-resolution

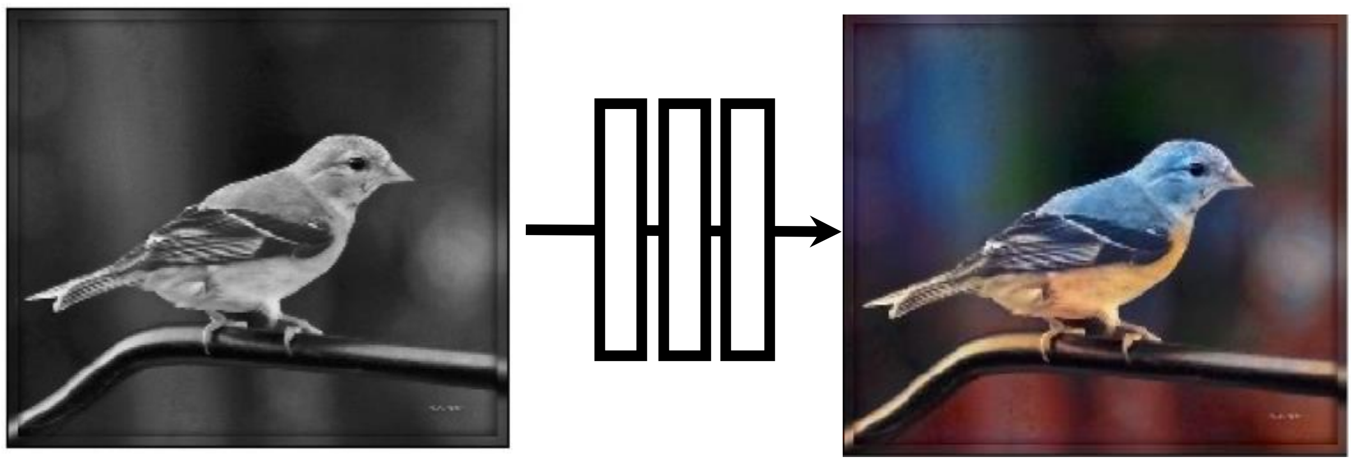


L2 regression

Slide credit: Phillip Isola

Designing Loss Functions

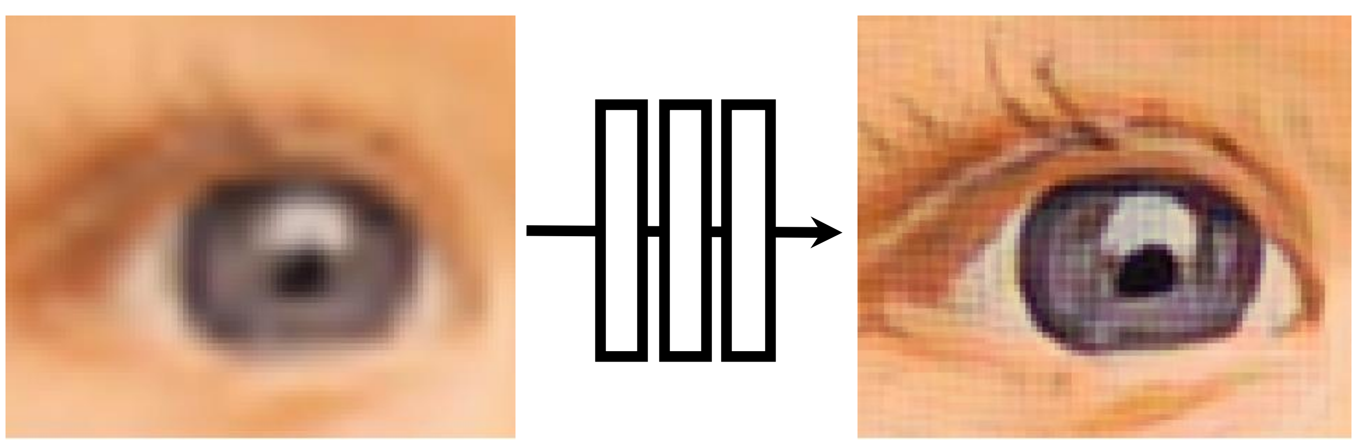
Image colorization



[Zhang et al. 2016]

Classification Loss:
Cross entropy objective,
with colorfulness term

Super-resolution



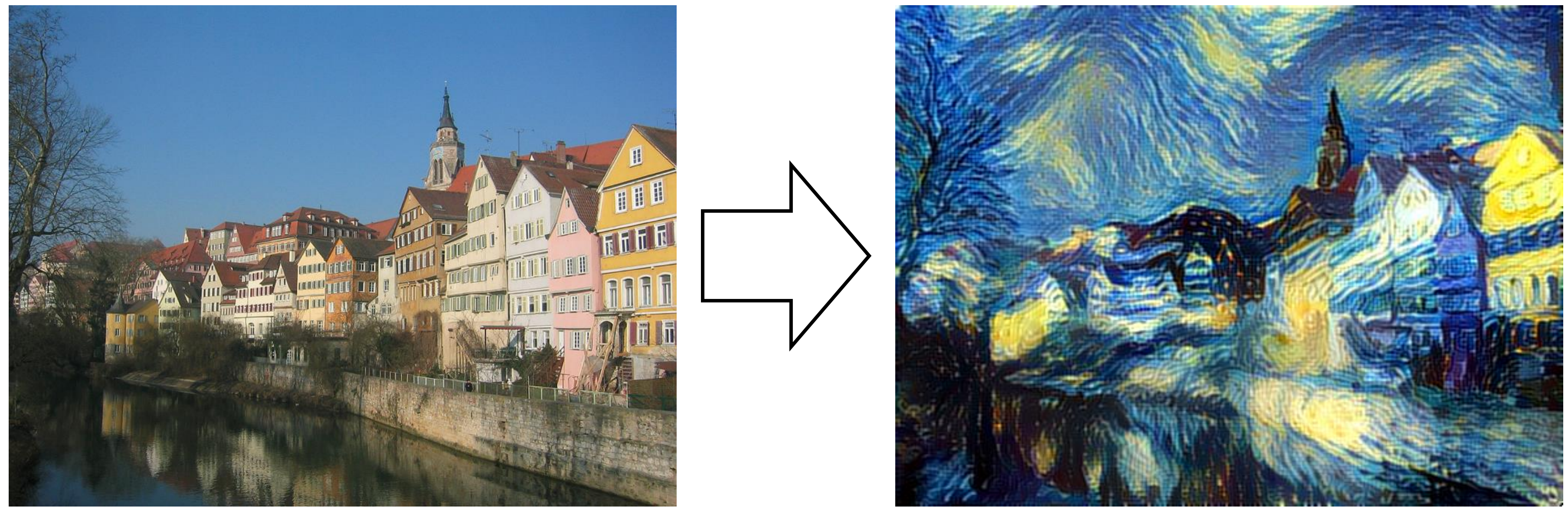
[Gatys et al., 2016], [Johnson et al. 2016]
[Dosovitskiy and Brox. 2016]

Feature/Perceptual loss
Deep feature matching
objective

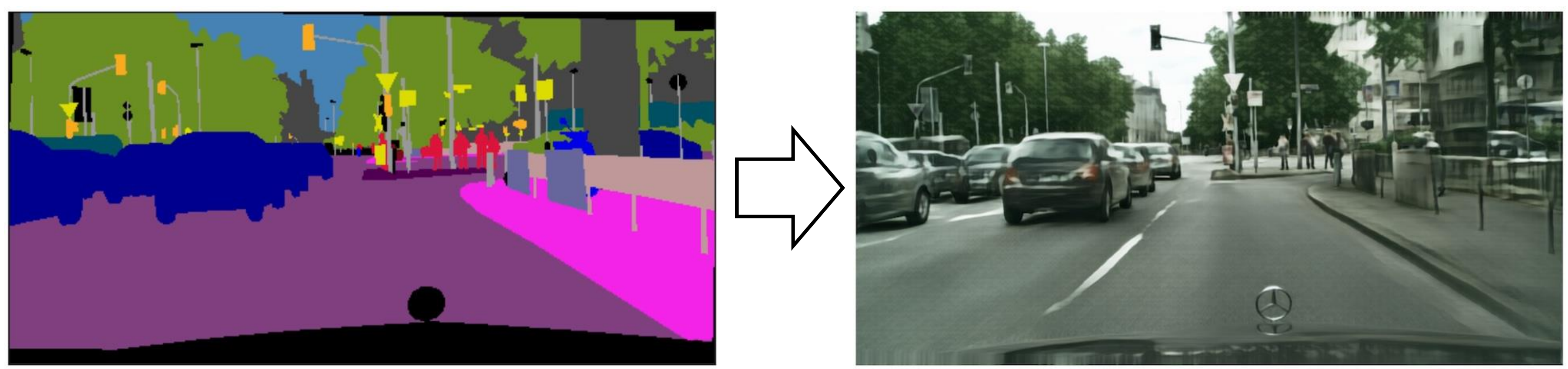
Slide credit: Phillip Isola

“Perceptual Loss”

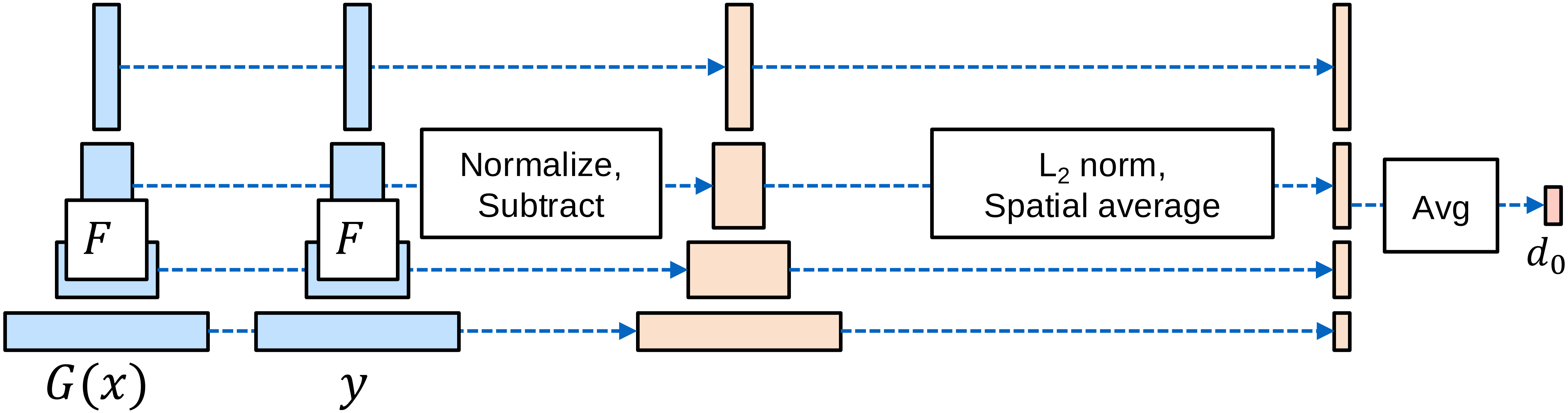
Gatys et al. In CVPR, 2016.
Johnson et al. In ECCV, 2016.
Dosovitskiy and Brox. In NIPS, 2016.



Chen and Koltun. In ICCV, 2017.



CNNs as a Perceptual Metric

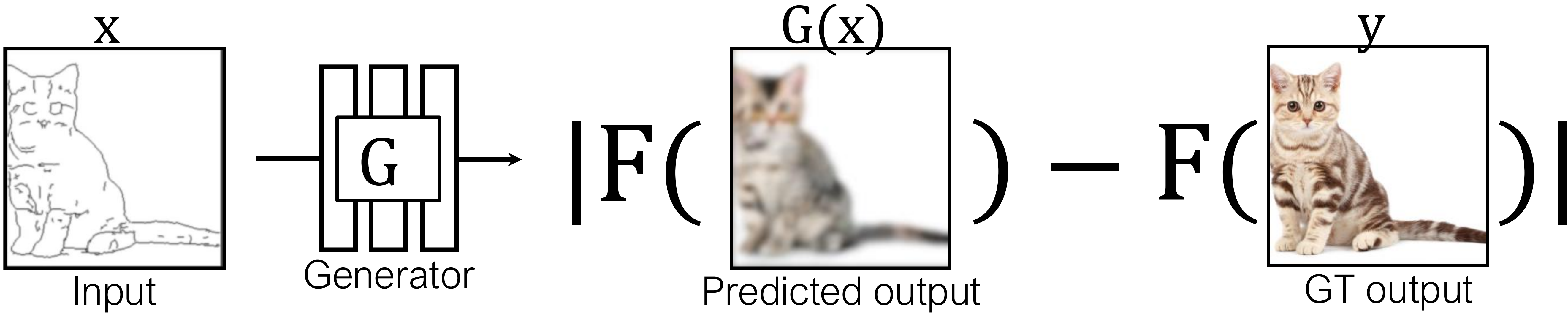


(1) How well do "perceptual losses" describe perception?

c.f. Gatys et al. CVPR 2016. Johnson et al. ECCV 2016. Dosovitskiy and Brox. NIPS 2016.

Slide credit: Richard Zhang

CNNs as a Perceptual Metric



F is a deep network (e.g., ImageNet classifier)

Perceptual Loss

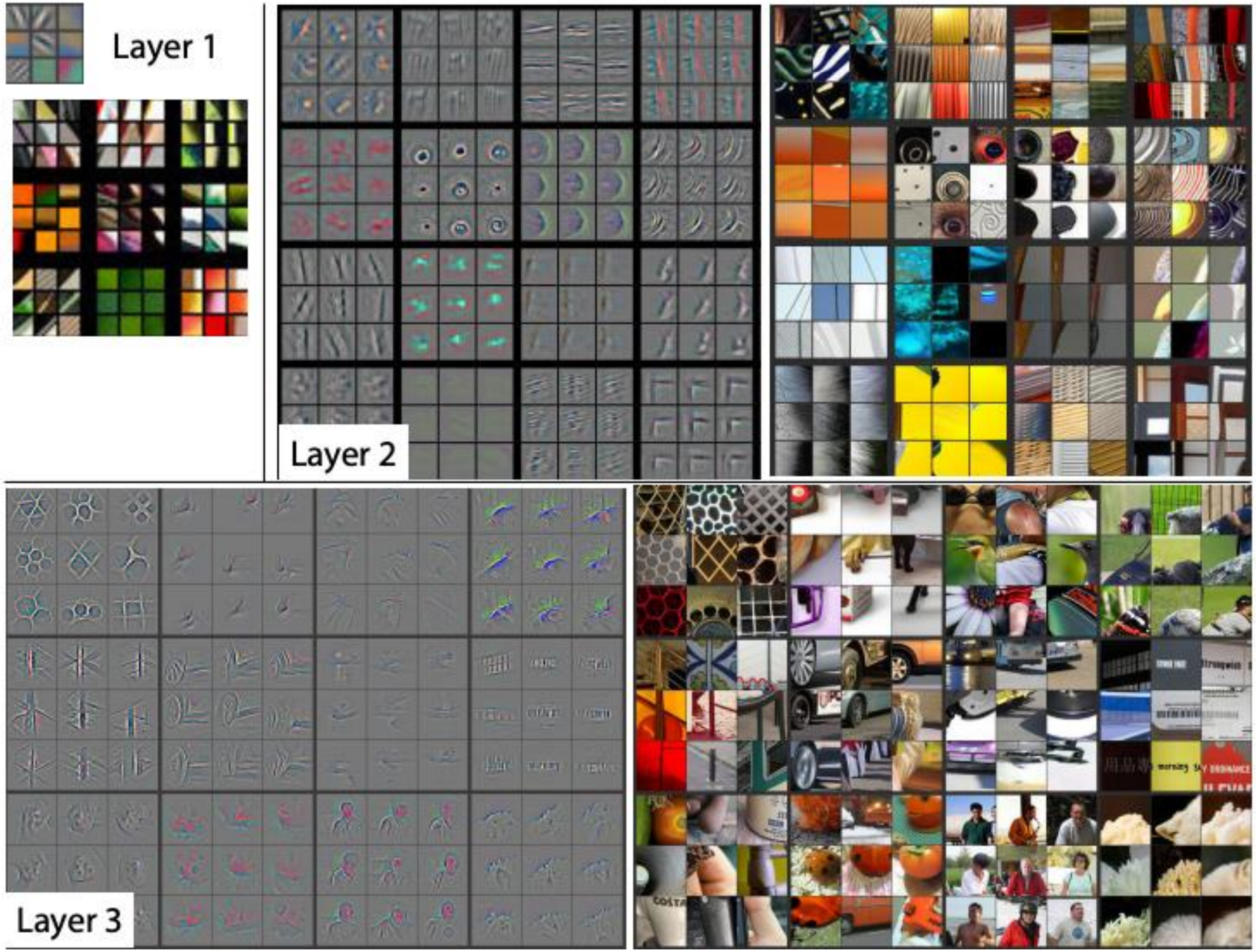
weight

(i)-th layer

$$\arg\min_G \mathbb{E}_{(x,y)} \sum_{i=1}^{N} \lambda_i \frac{1}{M_i} ||F^{(i)}(G(x)) - F^{(i)}(y)||_2^2$$

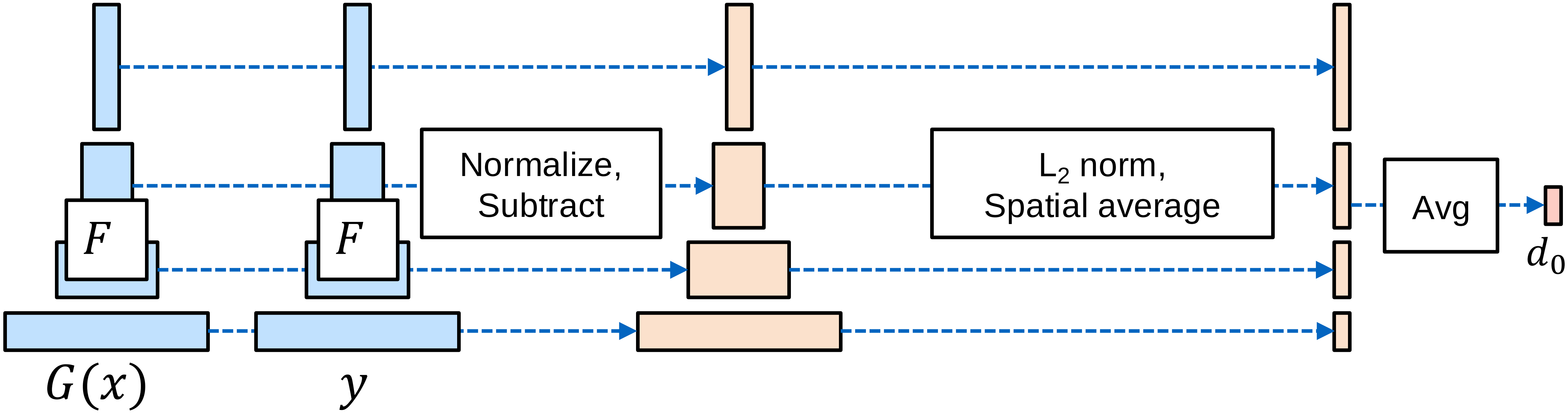
The number of elements in the (i)-th layer

What has a CNN Learned?



Zeiler and Fergus. In ECCV, 2014.

CNNs as a Perceptual Metric



Perceptual Loss

weight

(i)-th layer

$$\arg\min_G \mathbb{E}_{(x,y)} \sum_{i=1}^{N} \lambda_i \frac{1}{M_i} ||F^{(i)}(G(x)) - F^{(i)}(y)||_2^2$$

The number of elements in the (i)-th layer

Slide credit: Richard Zhang

How Different are these Patches?

D ( , )

Zhang, Isola, Efros, Shechtman, Wang.
The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*, 2018.

Slide credit: Richard Zhang

Which patch is more similar to the middle?

< Type 1 >

Humans

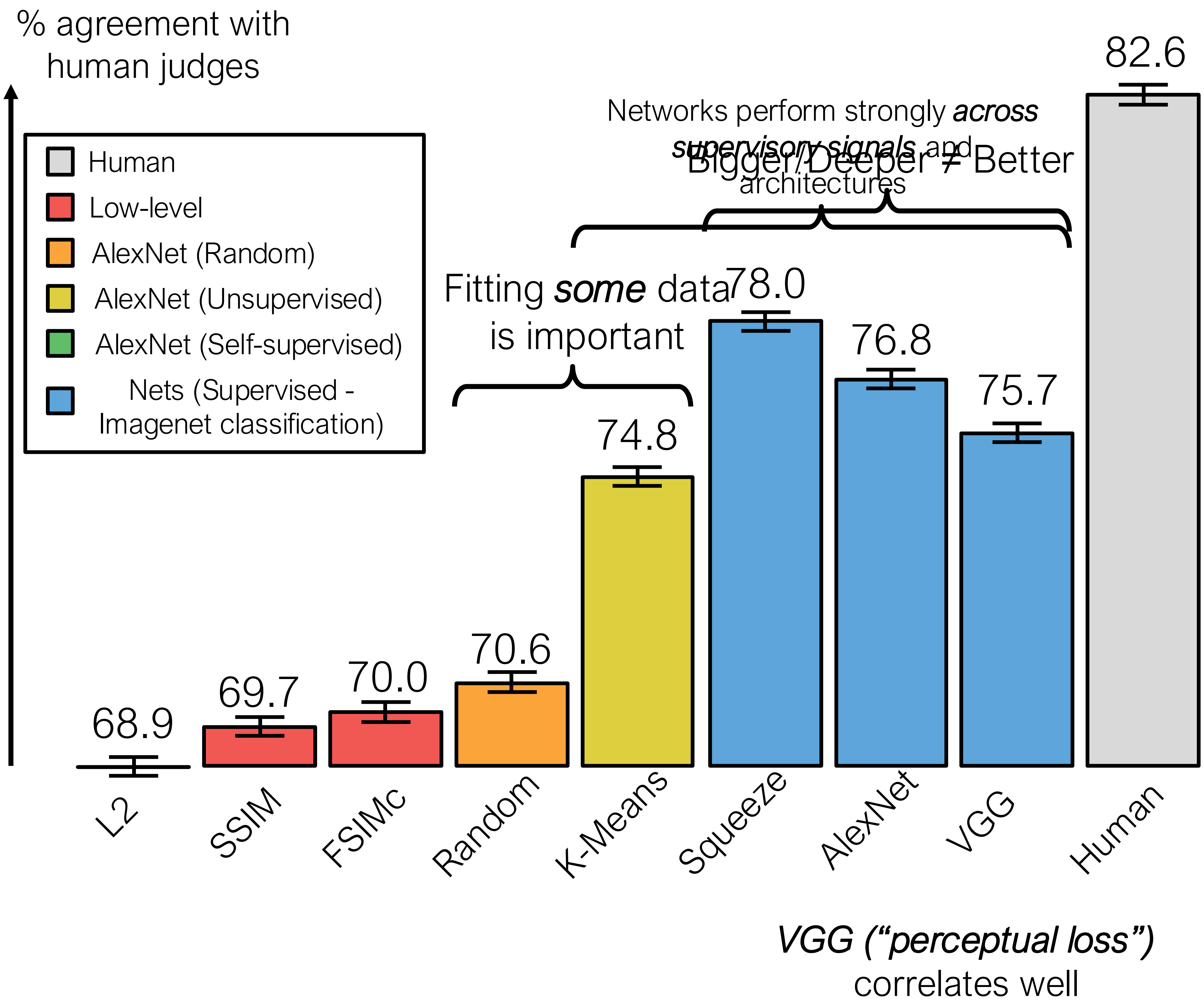
L2/PSNR

SSIM/FSIMc

Deep Networks?

< Type 2 >

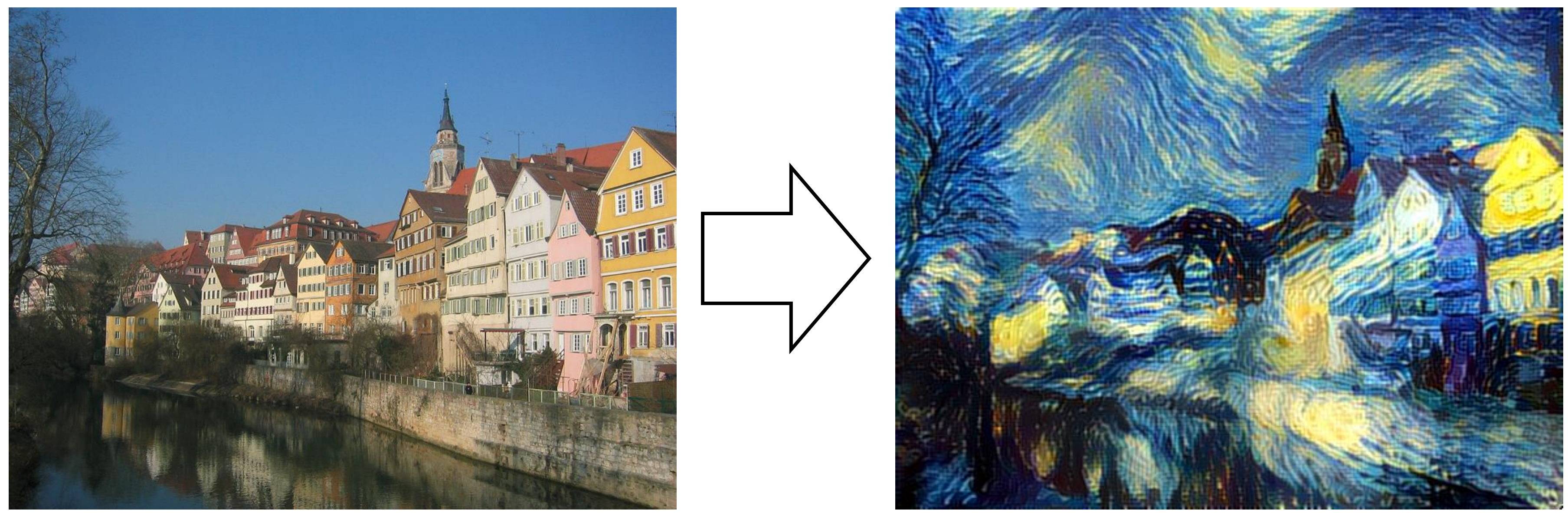
Slide credit: Richard Zhang



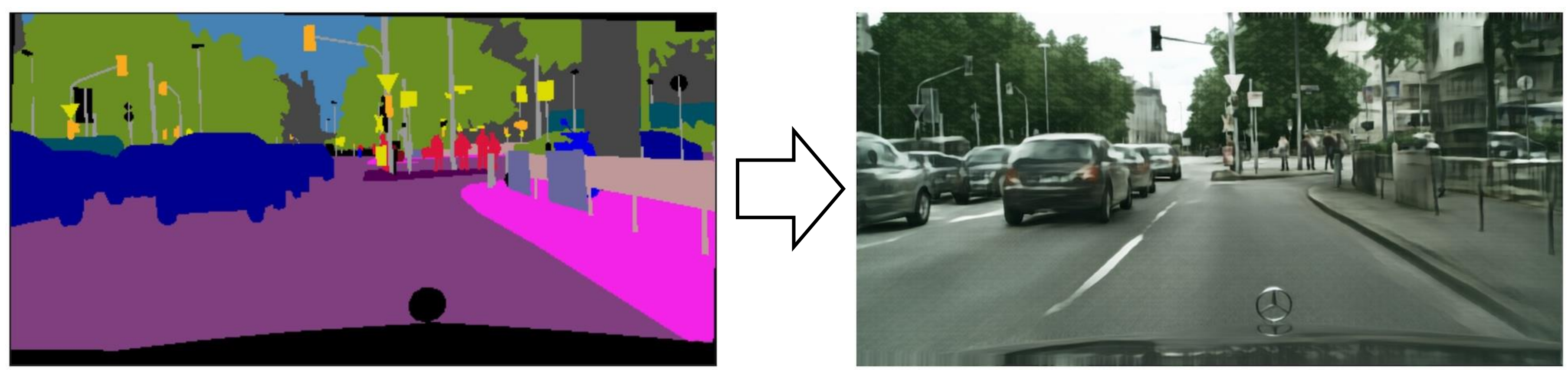
Slide credit: Richard Zhang

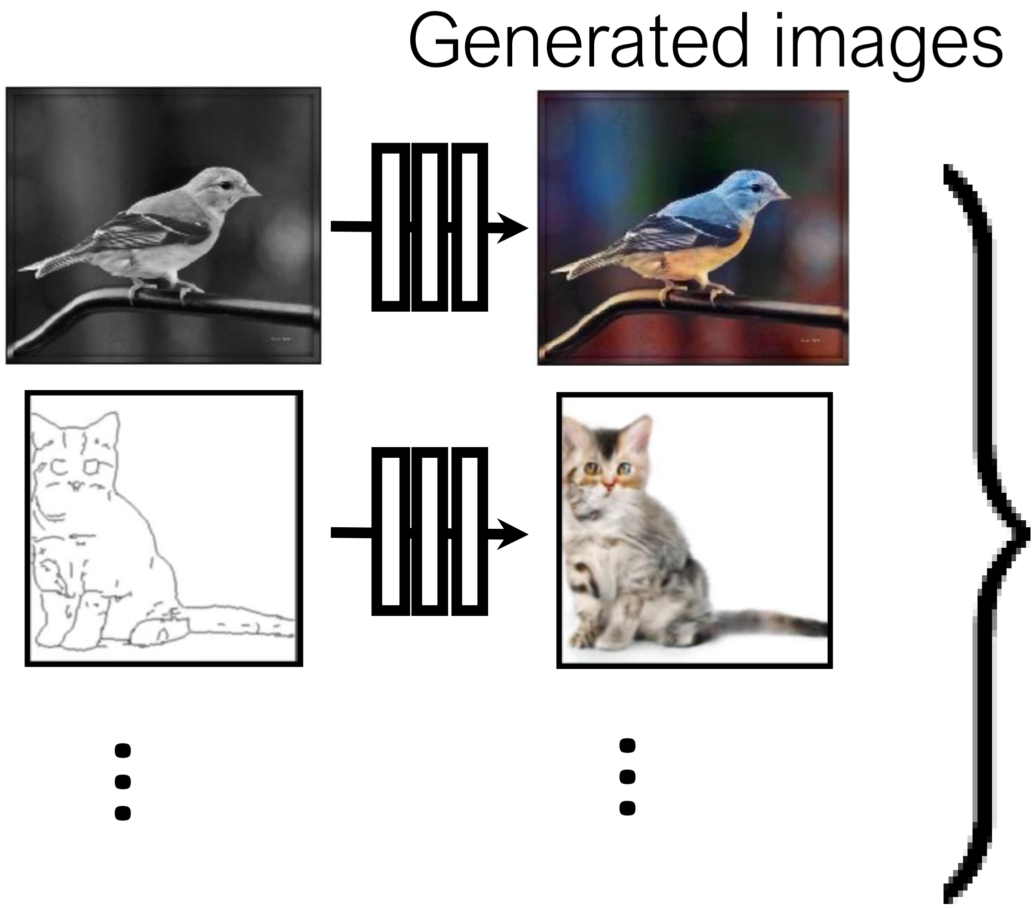
“Perceptual Loss”

Gatys et al. In CVPR, 2016.
Johnson et al. In ECCV, 2016.
Dosovitskiy and Brox. In NIPS, 2016.



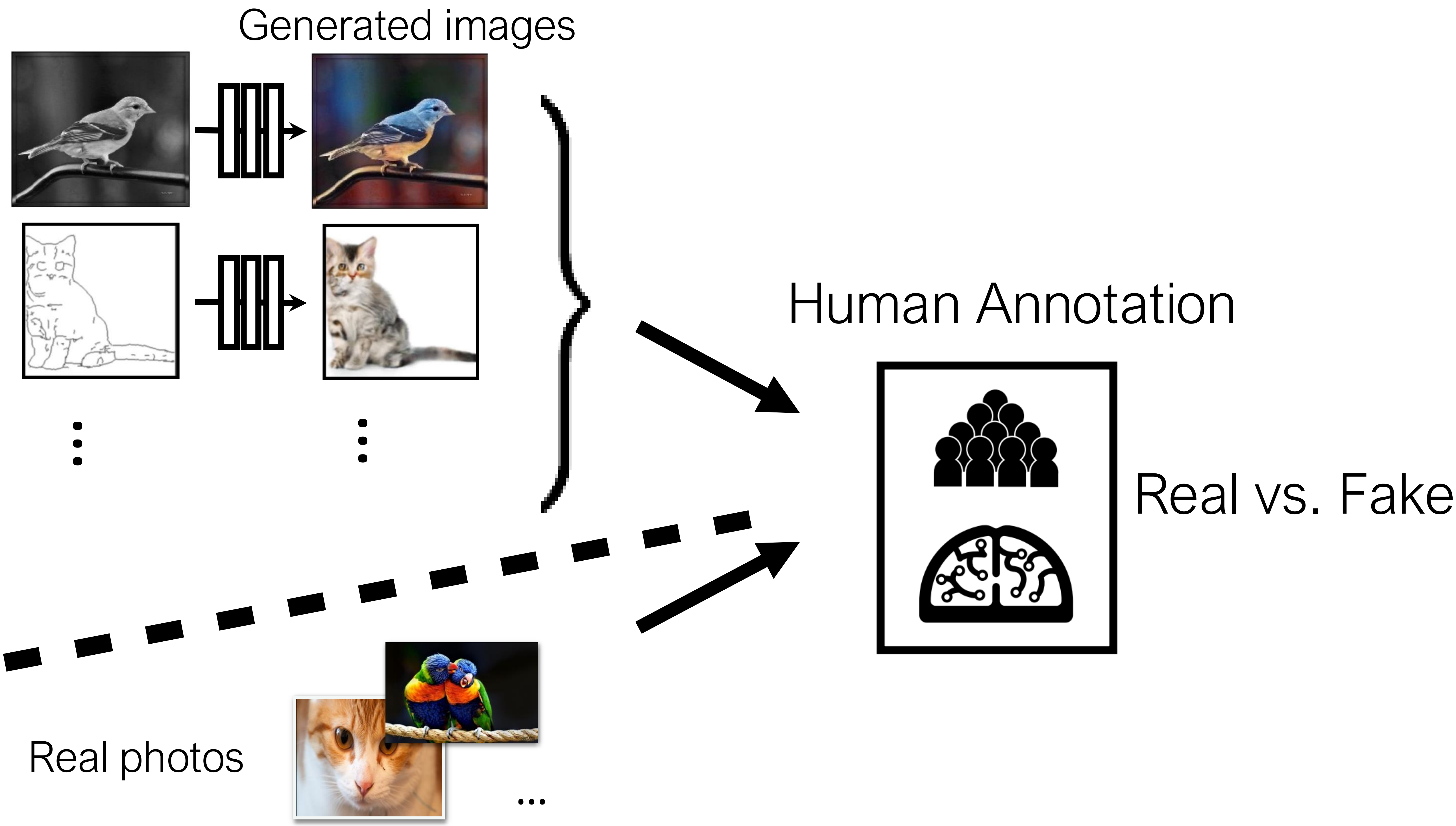
Chen and Koltun. In ICCV, 2017.





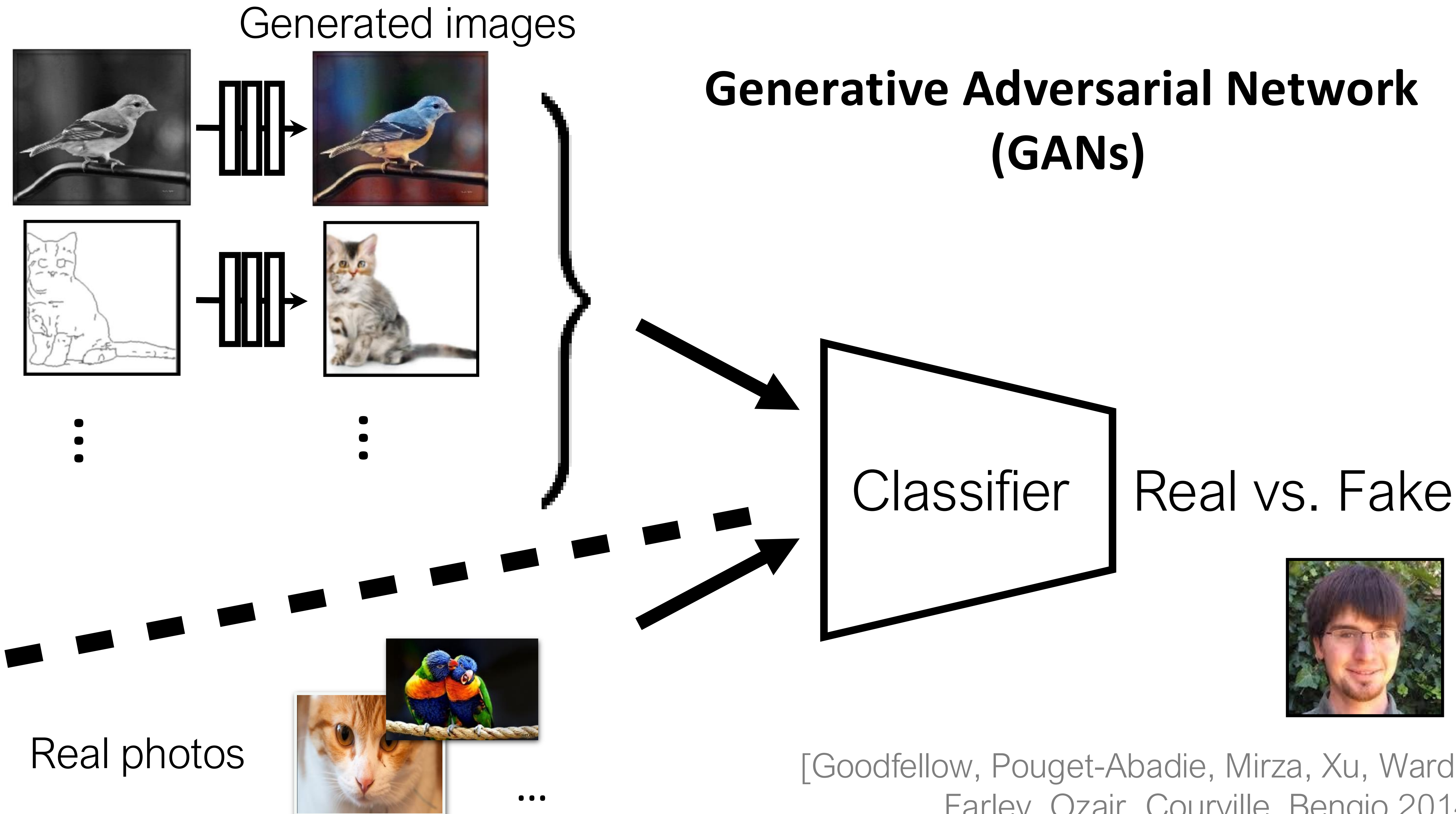
Universal loss?

Learning with Human Perception



[Zhu et al. 2014]

Generative Adversarial Network (GANs)



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

Image synthesis from “noise”

G

$z \sim p(z)$

Generator

$x = G(z)$

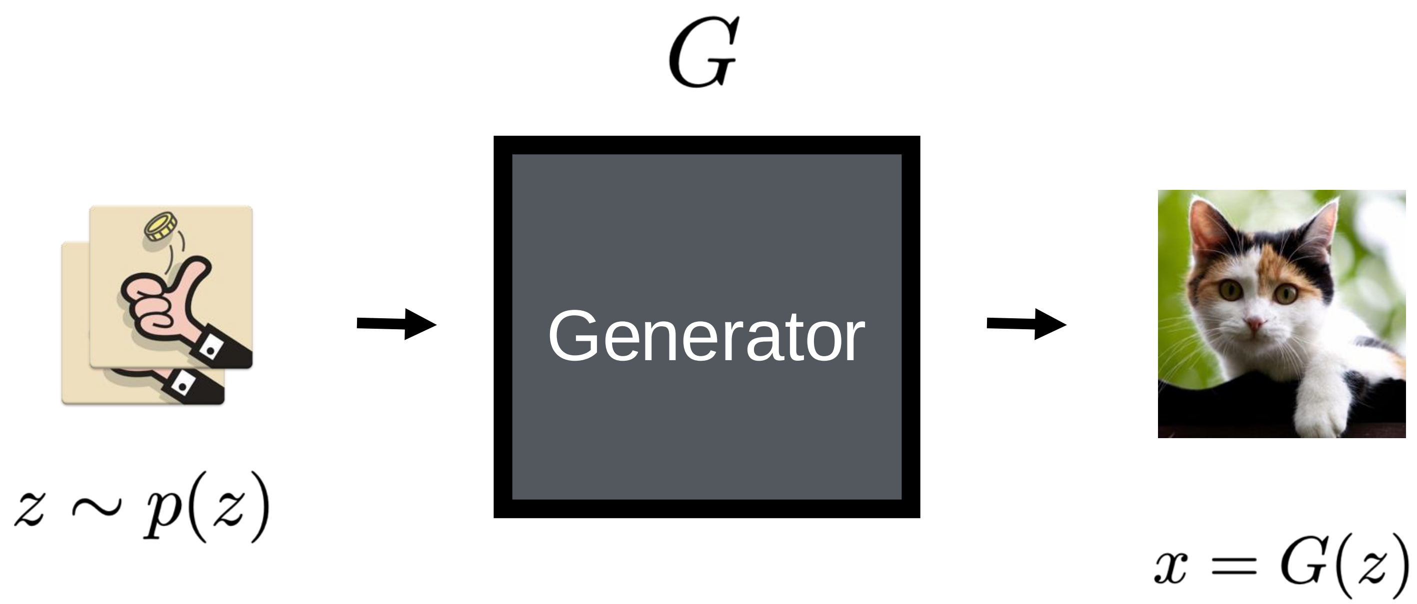
Sampler

$$G : \mathcal{Z} \to \mathcal{X}$$

$$z \sim p(z)$$

$$x = G(z)$$

Image synthesis from “noise”



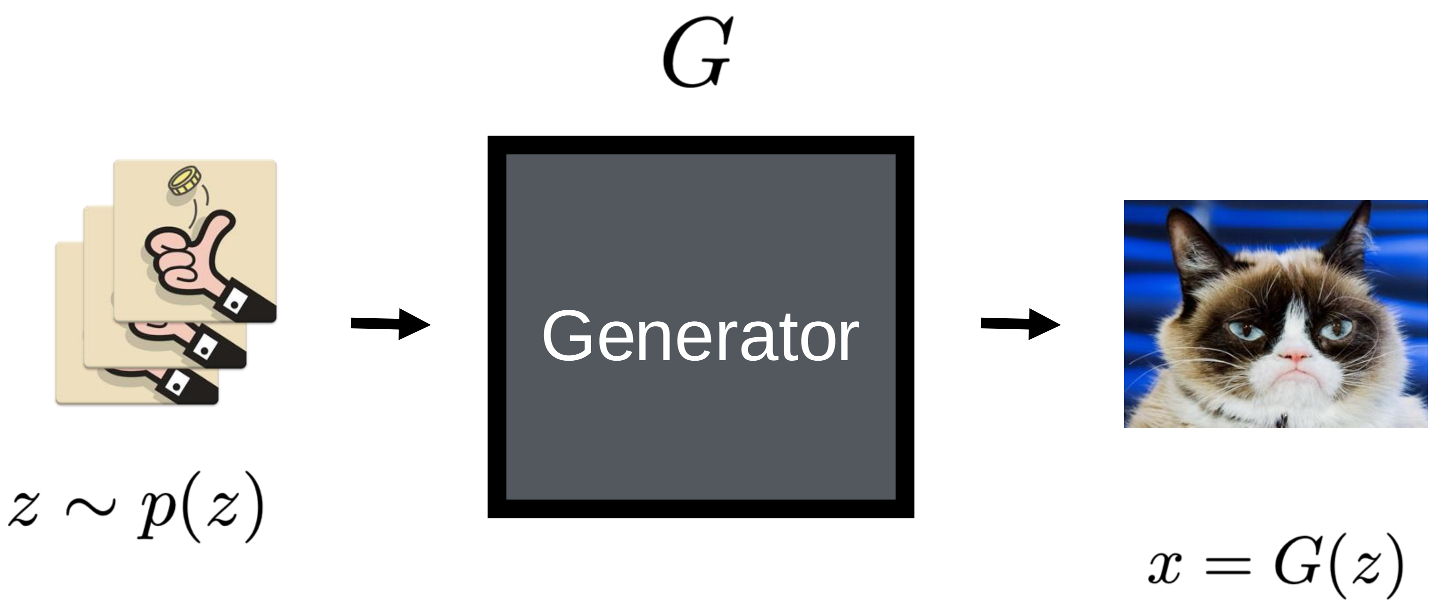
Sampler

$$G : \mathcal{Z} \rightarrow \mathcal{X}$$

$$z \sim p(z)$$

$$x = G(z)$$

Image synthesis from “noise”

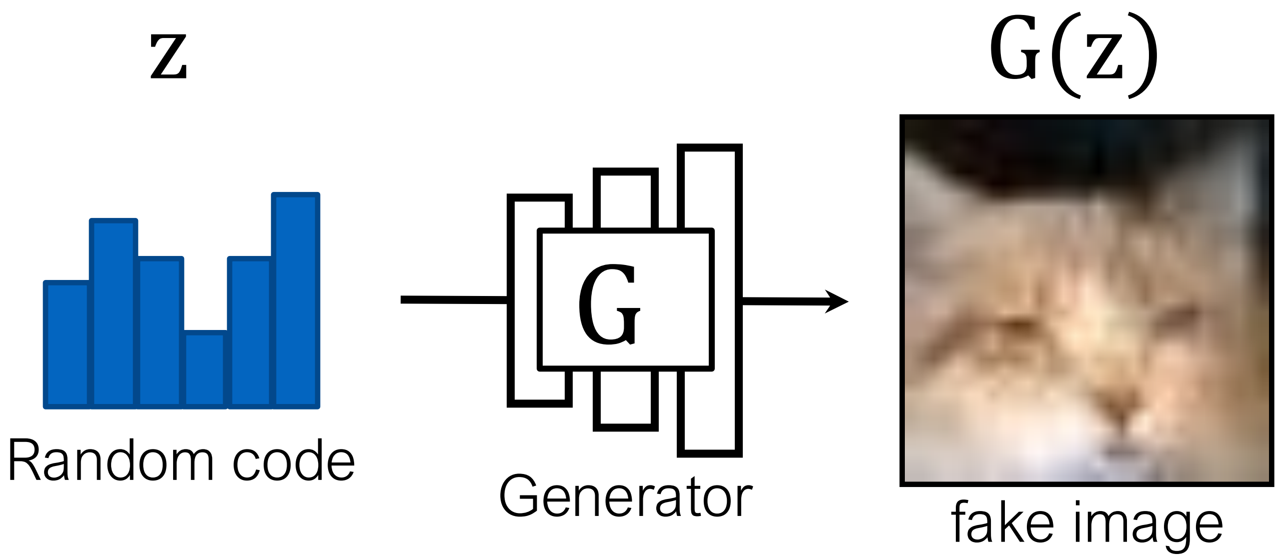


Sampler

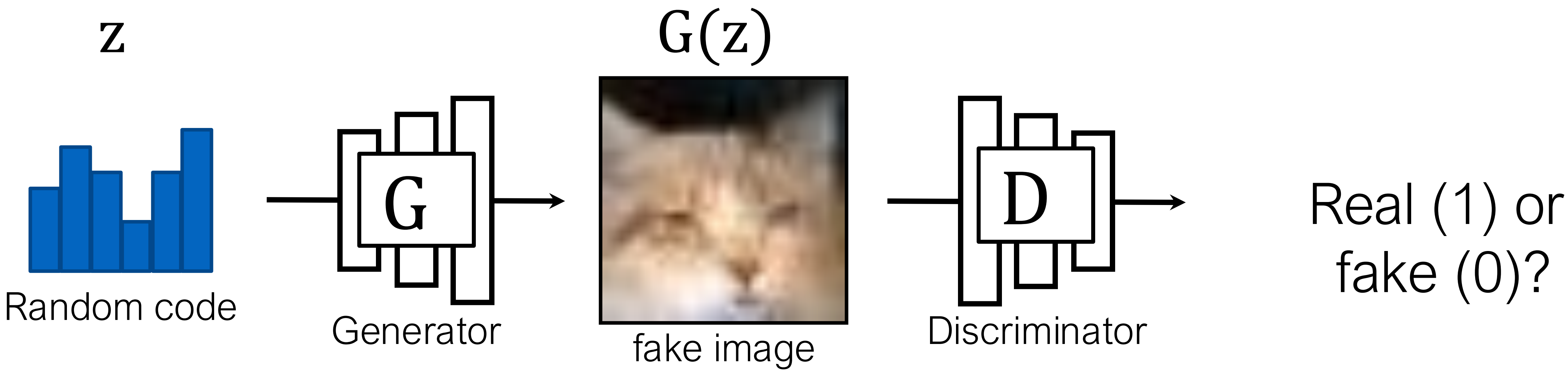
$$G : \mathcal{Z} \to \mathcal{X}$$

$$z \sim p(z)$$

$$x = G(z)$$



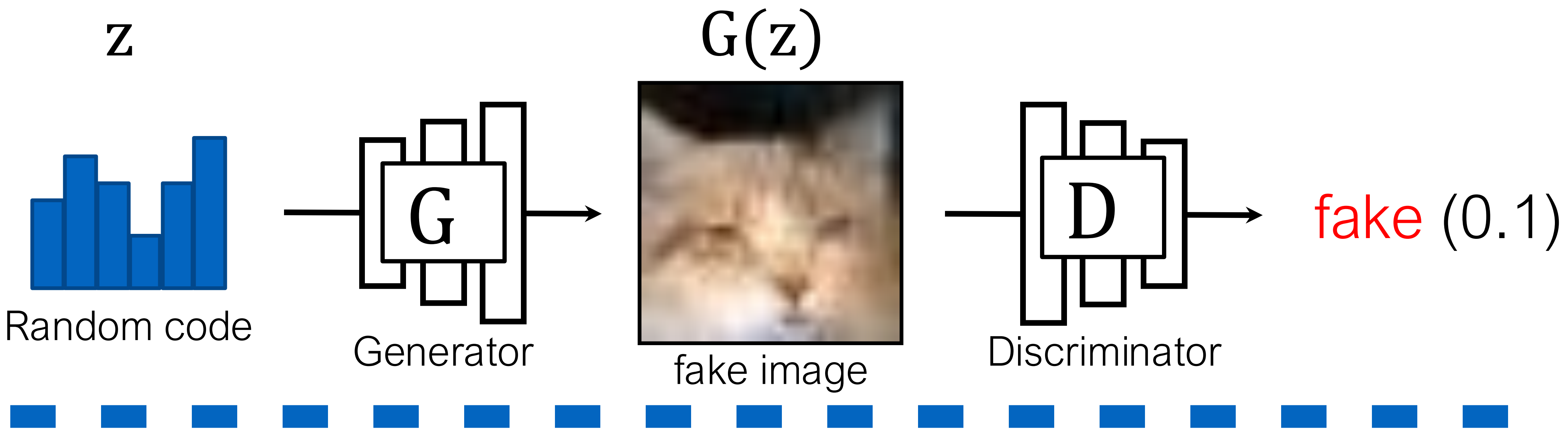
[Goodfellow et al. 2014]



A two-player game:

- G tries to generate fake images that can fool D.
- D tries to detect fake images.

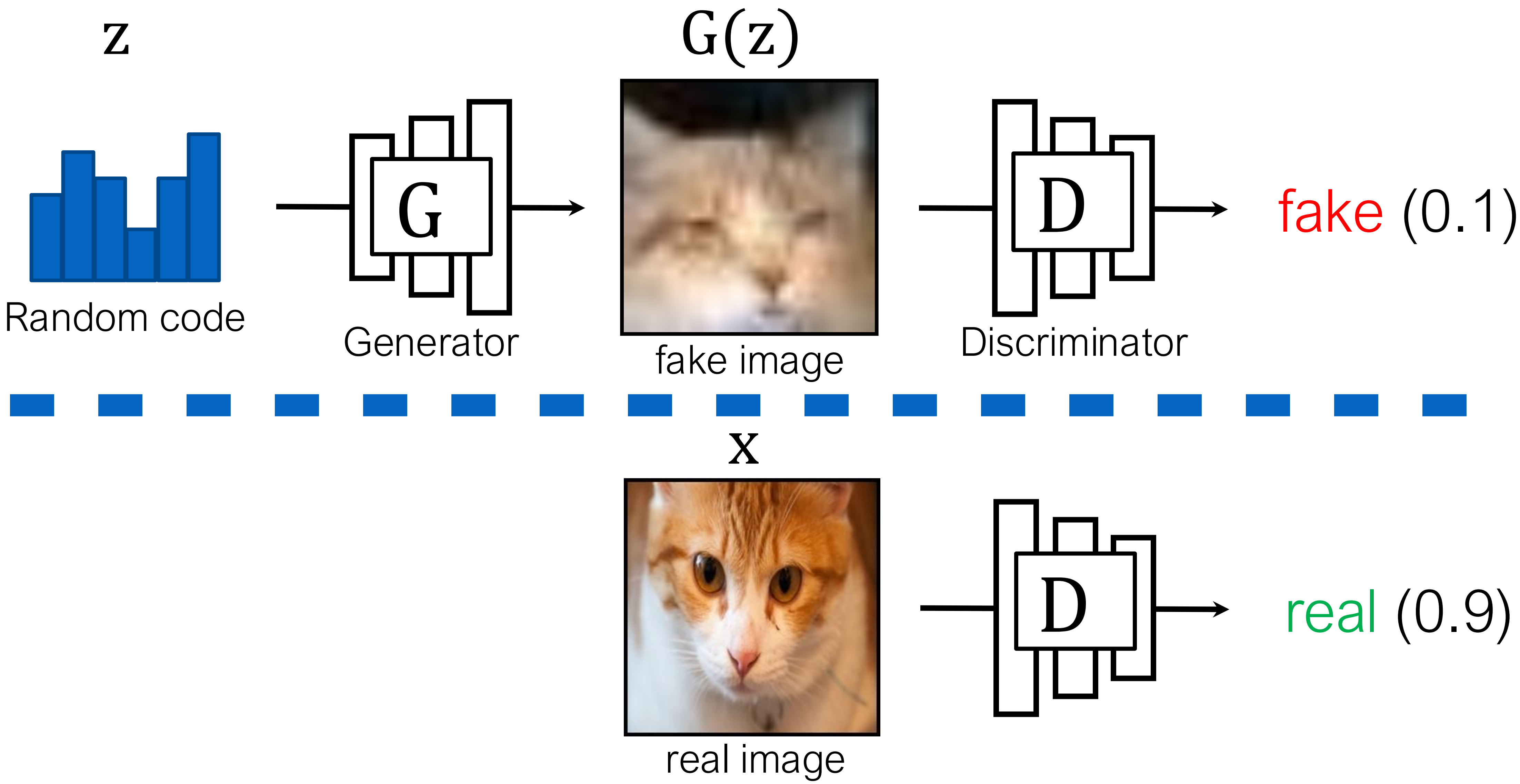
[Goodfellow et al. 2014]



Learning objective (GANs)

$$\min_{G} \max_{\boxed{D}} \boxed{\mathbb{E}_z[\log(1 - D(G(z))]}$$

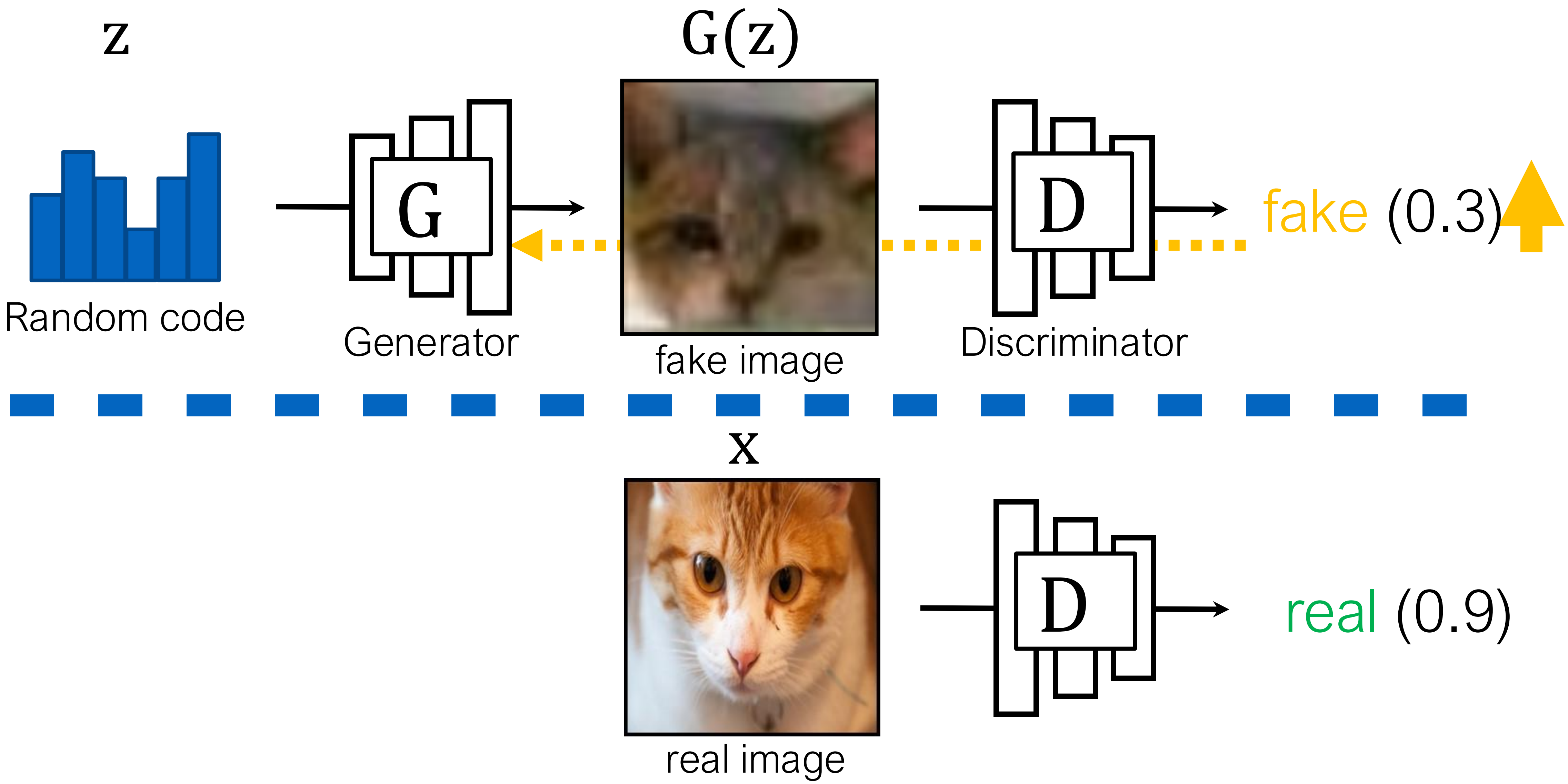
[Goodfellow et al. 2014]



Learning objective (GANs)

$$\min_{G} \max_{\boxed{D}} \boxed{\mathbb{E}_z[\log(1-D(G(z)))]} + \boxed{\mathbb{E}_x[\log D(x)]}$$

[Goodfellow et al. 2014]



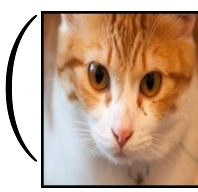
Learning objective (GANs)

$$\min_{\boxed{G}} \max_{D} \boxed{\mathbb{E}_z[\log(1 - D(G(z)))]} + \mathbb{E}_x[\log D(x)]$$

[Goodfellow et al. 2014]

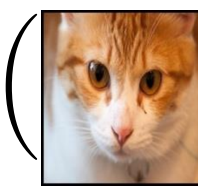
GANs Training Breakdown

- From the discriminator D’s perspective:
  - binary classification: real vs. fake.
  - Nothing special: similar to 1 vs. 7 or cat vs. dog

$\max\limits_{D} \mathbb{E}[\log(1 - D($  $))] + \mathbb{E}[\log D($  $)]$

GANs Training Breakdown

- From the discriminator D’s perspective:
  - binary classification: real vs. fake.
  - Nothing special: similar to 1 vs. 7 or cat vs. dog

$\max_{D} \mathbb{E}[\log(1 - D($  $))] + \mathbb{E}[\log D($  $)]$

- From the generator G’s perspective:
  - Optimizing a loss that depends on a classifier D
  - We have done it before (Perceptual Loss)

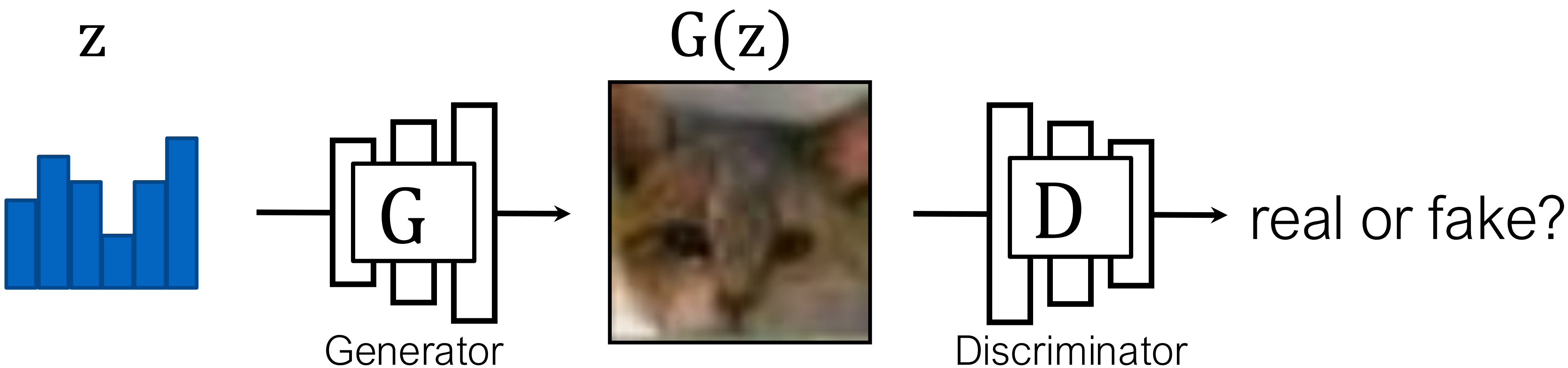
$$\min_{G} \mathbb{E}_z[\mathcal{L}_D(G(z))]$$

GAN loss for G

$$\min_{G} \mathbb{E}_{(x,y)} ||F(G(x)) - F(y)||$$

Perceptual Loss for G

GANs Training Breakdown



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

D tries to identify the fakes

- Training: iterate between training D and G with backprop.
- Global optimum when G reproduces data distribution.

[Goodfellow et al., 2014]

$p_g = p_{data}$ is the unique global minimizer of the GAN objective.

Proof

Optimal discriminator given fixed G

$$
\begin{aligned}
C(G) &= \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}[\log D_G^*(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x} \sim p_g}[\log(1 - D_G^*(\boldsymbol{x}))] \\
&= \mathbb{E}_{\boldsymbol{x} \sim p_{\text{data}}}\left[\log \frac{p_{\text{data}}(\boldsymbol{x})}{P_{\text{data}}(\boldsymbol{x}) + p_g(\boldsymbol{x})}\right] + \mathbb{E}_{\boldsymbol{x} \sim p_g}\left[\log \frac{p_g(\boldsymbol{x})}{p_{\text{data}}(\boldsymbol{x}) + p_g(\boldsymbol{x})}\right]
\end{aligned}
$$

$$
C(G) = -\log(4) + KL\left(p_{\text{data}} \left\| \frac{p_{\text{data}} + p_g}{2}\right.\right) + KL\left(p_g \left\| \frac{p_{\text{data}} + p_g}{2}\right.\right)
$$

$$
C(G) = -\log(4) + 2 \cdot \underbrace{JSD\left(p_{\text{data}} \| p_g\right)}_{\geq 0, \quad 0 \iff p_g = p_{data}} \quad \square
$$

KLD (Kullback–Leibler divergence): $\mathcal{KL}(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx$

JSD (Jensen–Shannon divergence): $\mathcal{JSD}(p \parallel q) = \frac{1}{2}\mathcal{KL}(p \parallel \frac{p+q}{2}) + \frac{1}{2}\mathcal{KL}(q \parallel \frac{p+q}{2})$

What has driven GAN progress?

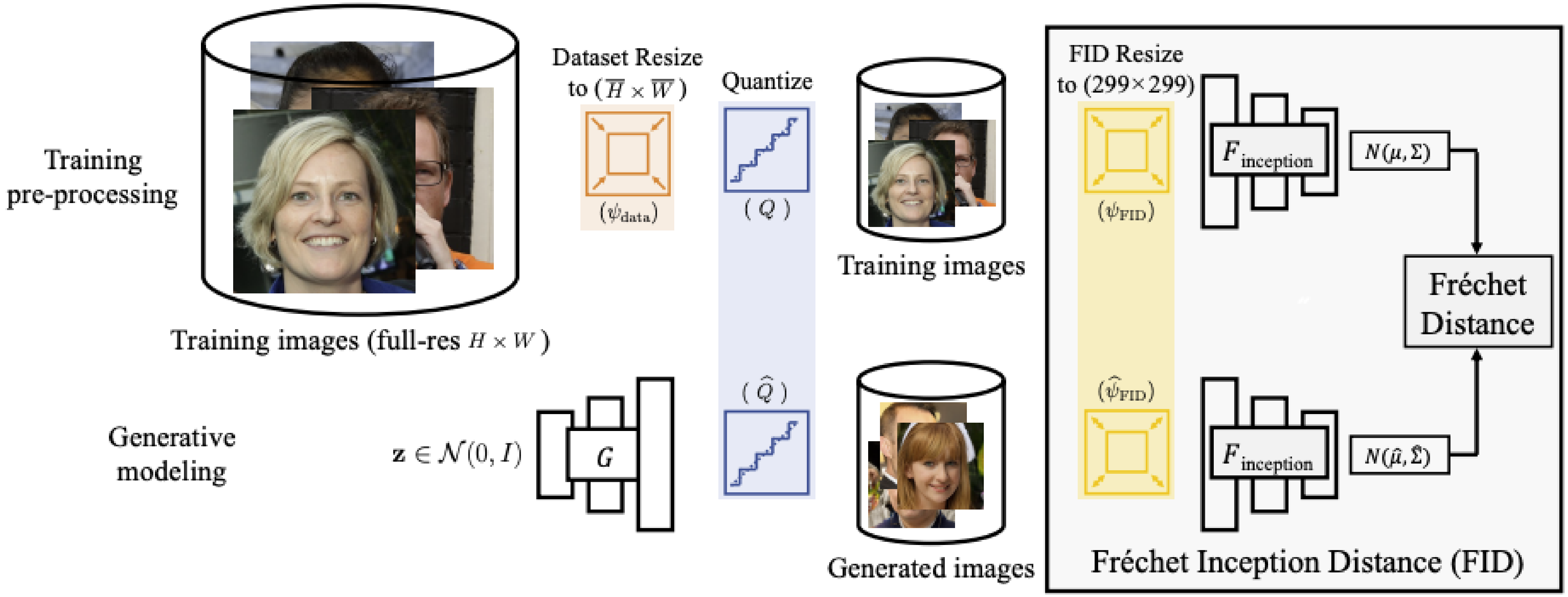
Ian Goodfellow @goodfellow_ian · Jan 14

4.5 years of **GAN progress** on face generation. arxiv.org/abs/1406.2661 arxiv.org/abs/1511.06434 arxiv.org/abs/1606.07536 arxiv.org/abs/1710.10196 arxiv.org/abs/1812.04948

What has driven GAN progress?

Samples from StyleGAN2 [Karras et al., CVPR 2020]

GANs evaluation (FID)



Fréchet Inception Distance (FID)

$$\mathbf{FID} = ||\mu - \hat{\mu}||_2^2 + \mathbf{Tr}(\Sigma + \widehat{\Sigma} - 2(\Sigma\widehat{\Sigma})^{1/2})$$

GANs evaluation (FID)

Clean-fid libraries for evaluating generative models

```
Python 3.7.10 (default, Feb 26 2021, 18:47:35)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

pip install clean-fid

Daily downloads (July, 2022): 100

Daily downloads (Feb, 2024) : 20, 000

Total downloads: 18, 000, 000+

[Parmar et al., CVPR 2022]