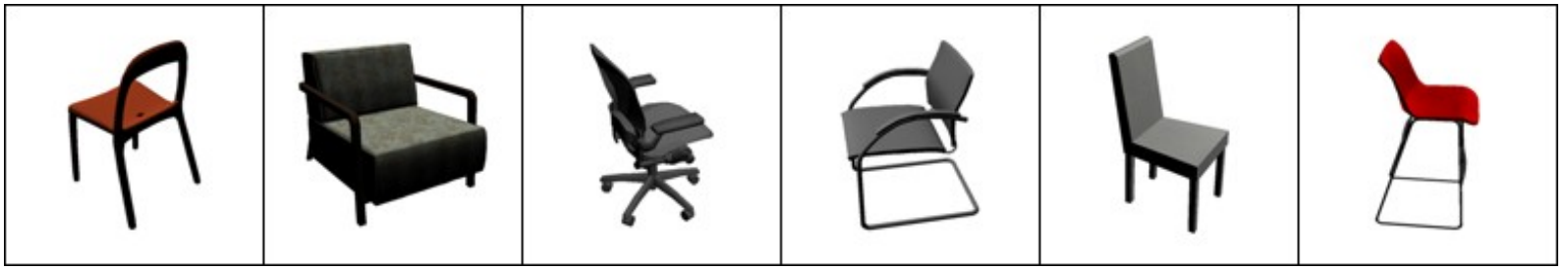
MLHEP

Day -1.1

Generative models

Image generation

- Chairs (type, view, orientation)



- Classifier

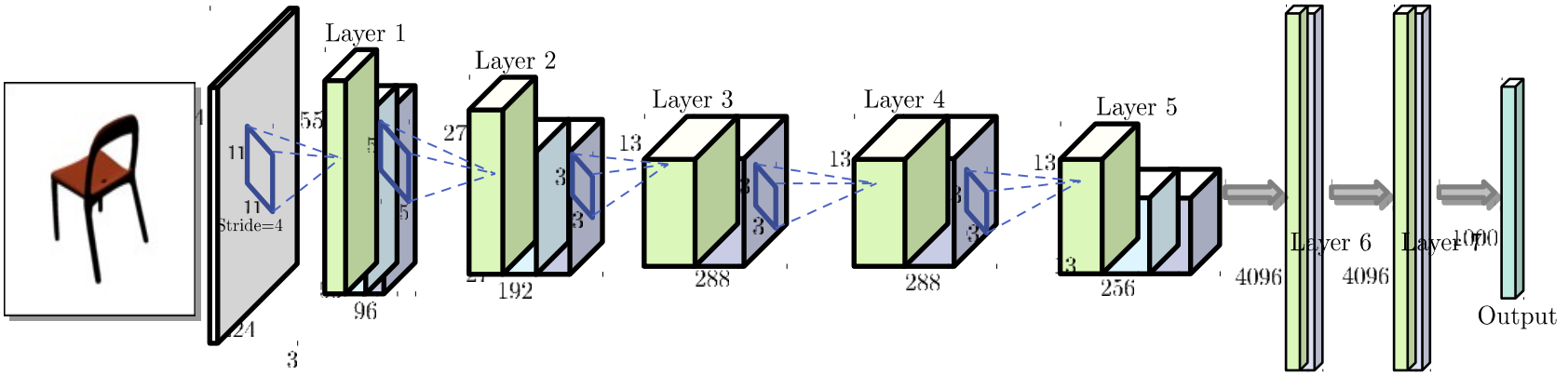
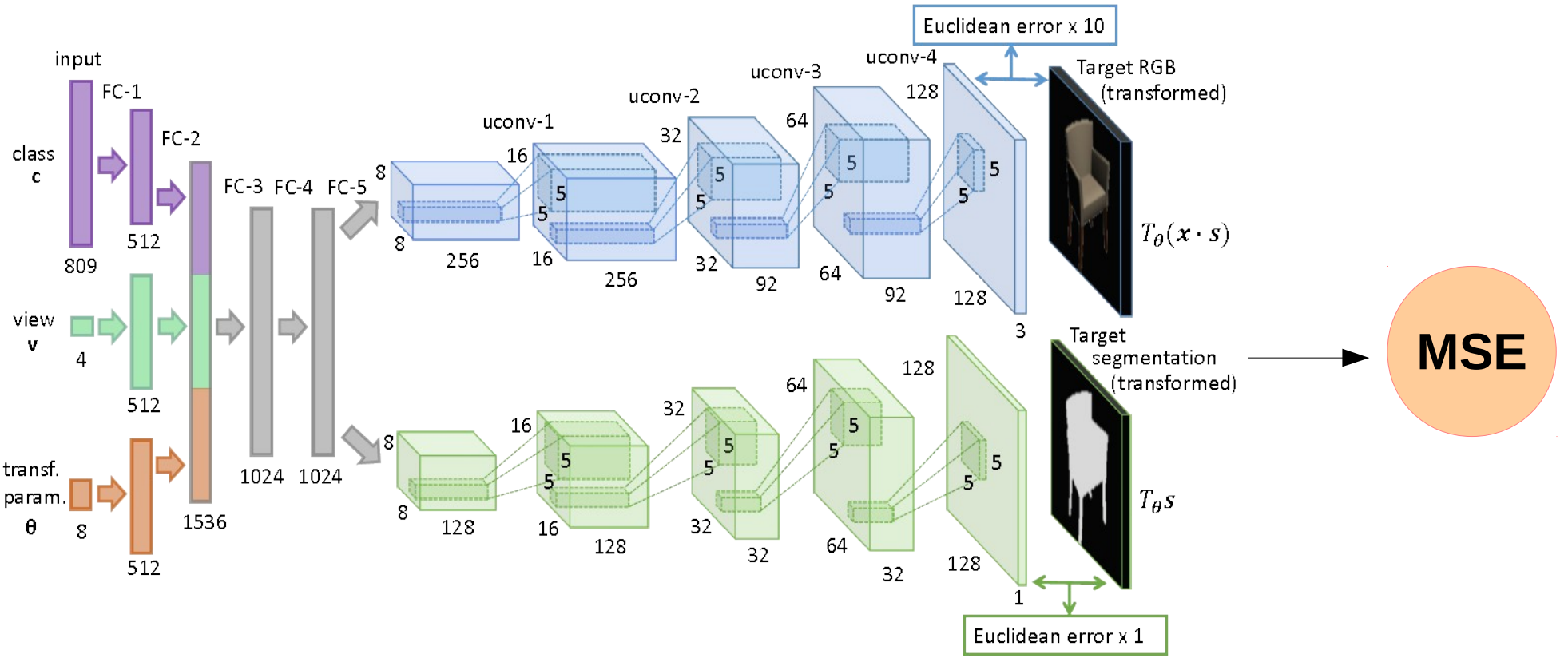


Image generation

- Generator



Mean Squared Error

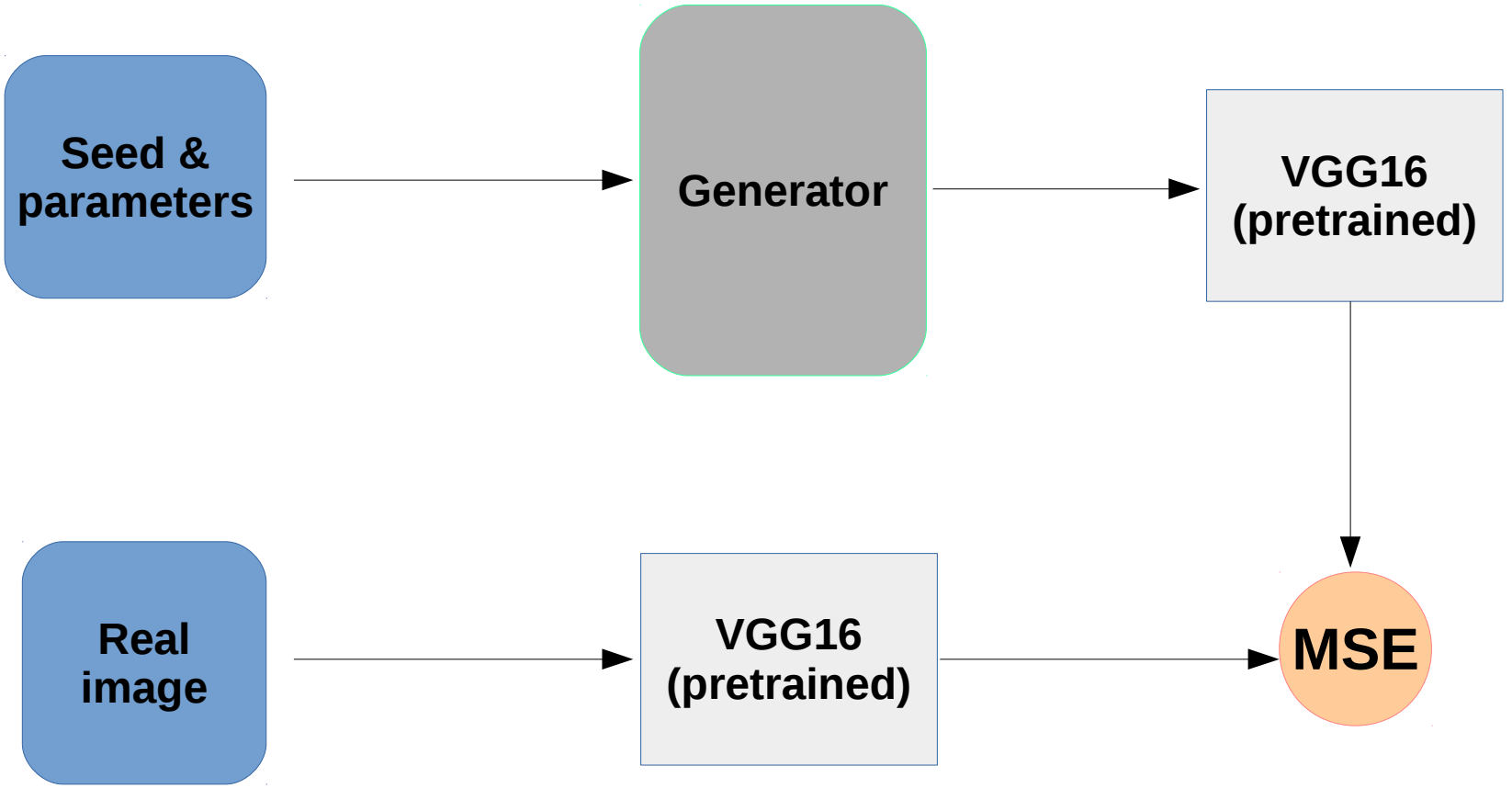
Pixelwise MSE:

- A **“cat on the left”** is closer to **“dog on the left”** than to **“cat on the right”**
- We may want to avoid that effect
- Can we obtain image representation that is less sensitive to small shifts?

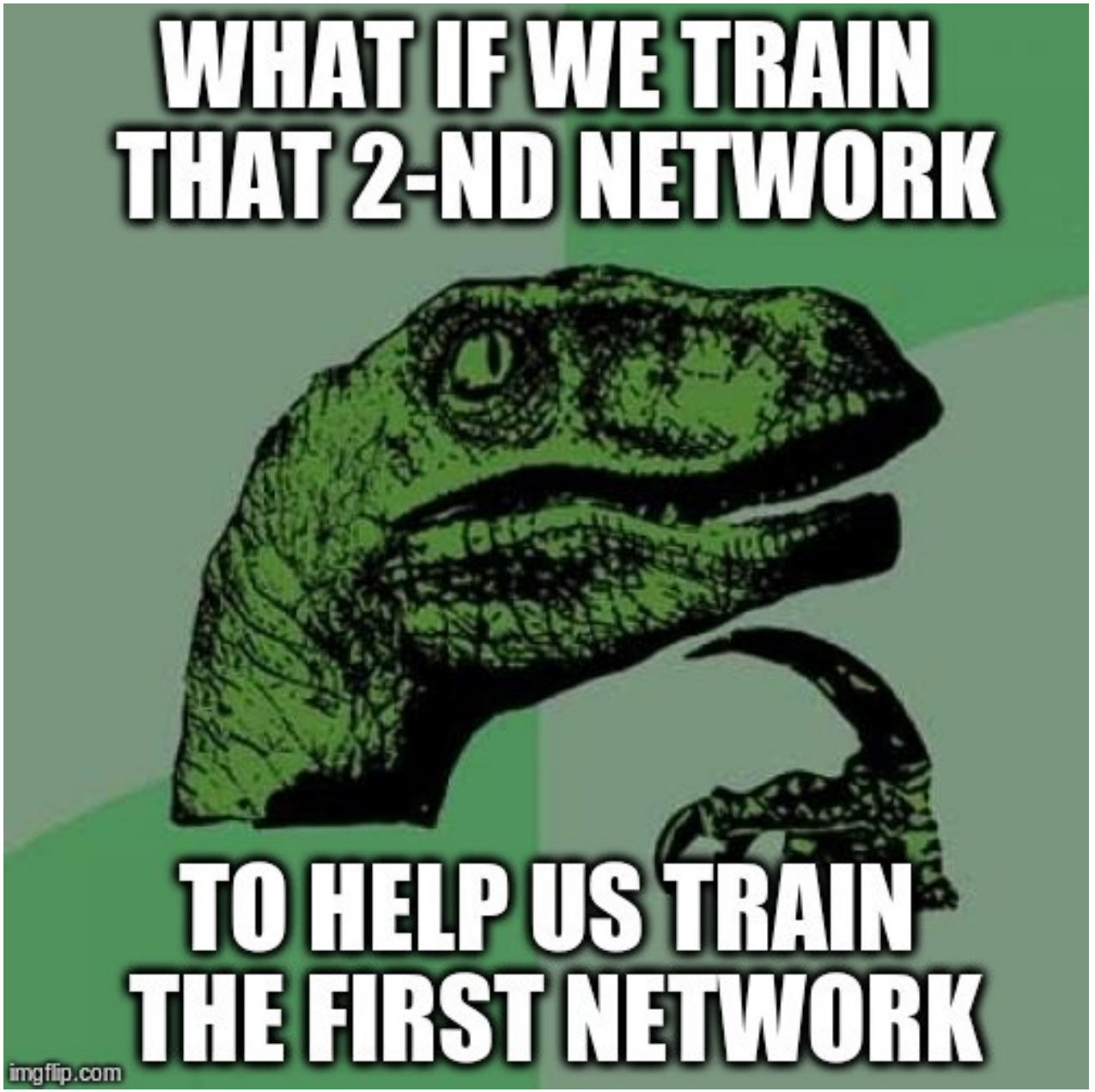
Problem: MSE sucks at this task.

Ideas?

Sketch: using pre-trained nets



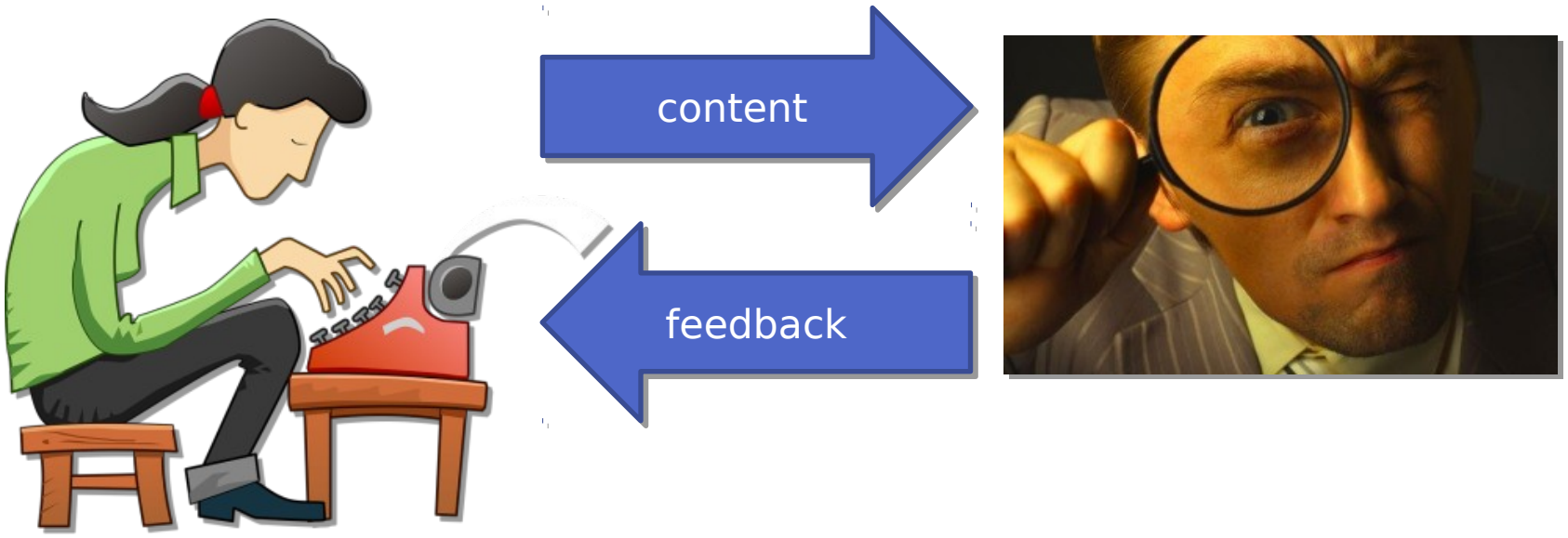
$$L = \| f(img) - f(Gen(seed)) \|$$


WHAT IF WE TRAIN
THAT 2-ND NETWORK
TO HELP US TRAIN
THE FIRST NETWORK
imgflip.com

Generative Adversatial Networks

Generator

Discriminator

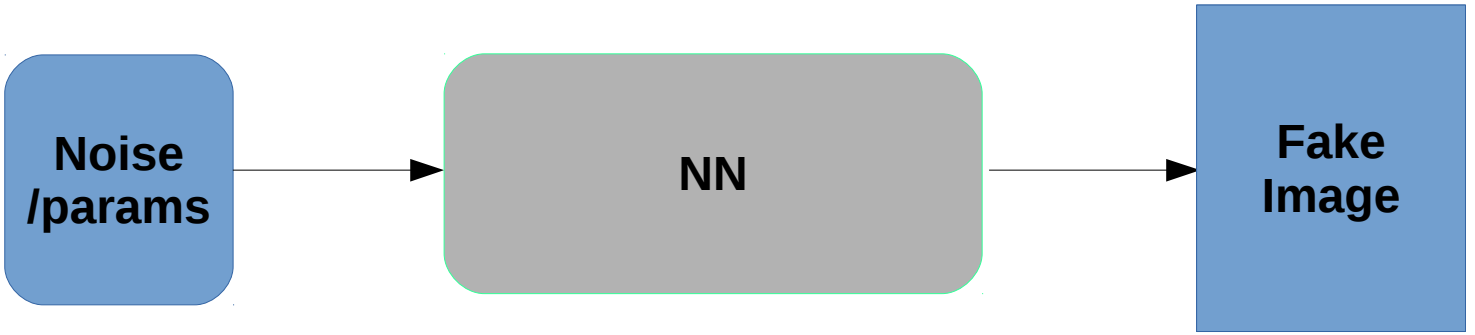


Generate image
(should be plausible)

Tell if image is plausible
(image) → P(fake)

Generative Adversarial Networks

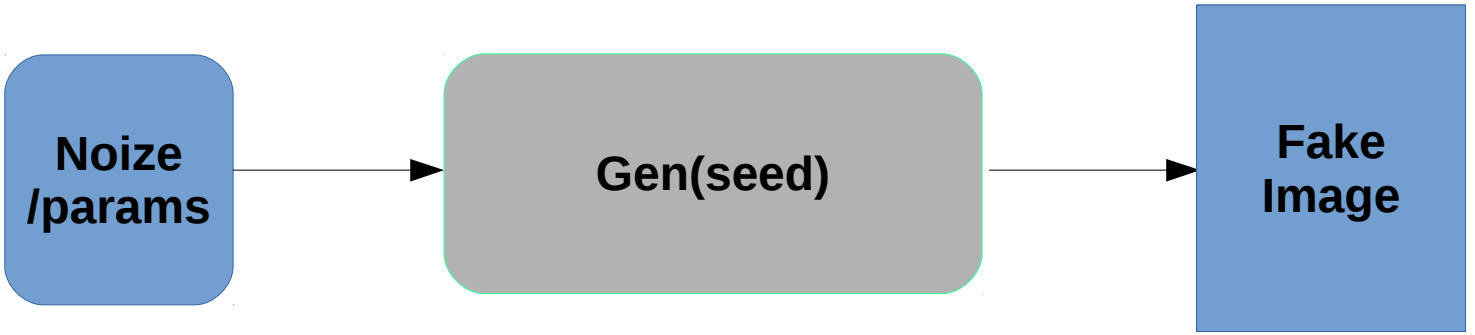
- Generator



- Discriminator

Generative Adversarial Networks

- Generator

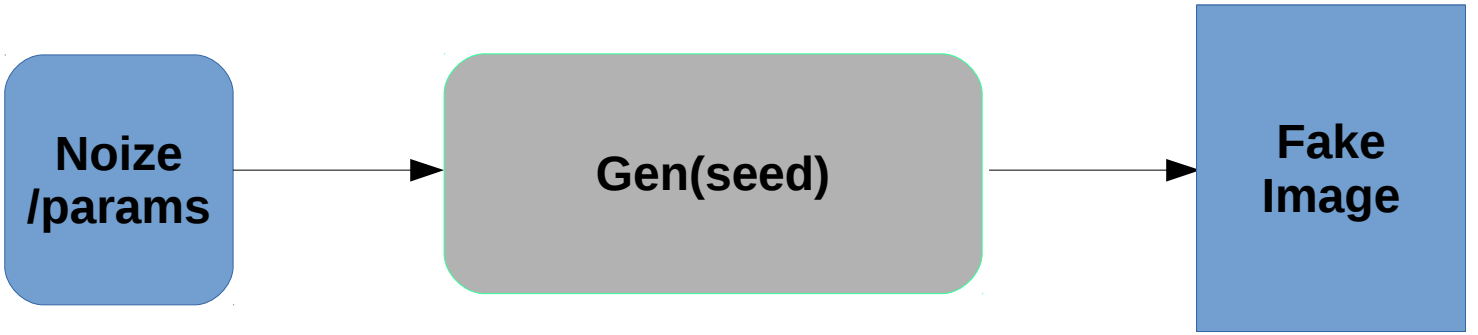


- Discriminator

$$L_D = -\log[1 - Disc(real\ data)] - \log Disc(Gen(seed))$$

Generative Adversarial Networks

- Generator $L_G = -\log\left[1 - Disc\left(Gen\left(seed\right)\right)\right]$



- Discriminator

$$L_D = -\log\left[1 - Disc\left(real\ data\right)\right] - \log Disc\left(Gen\left(seed\right)\right)$$

Generative Adversarial Networks

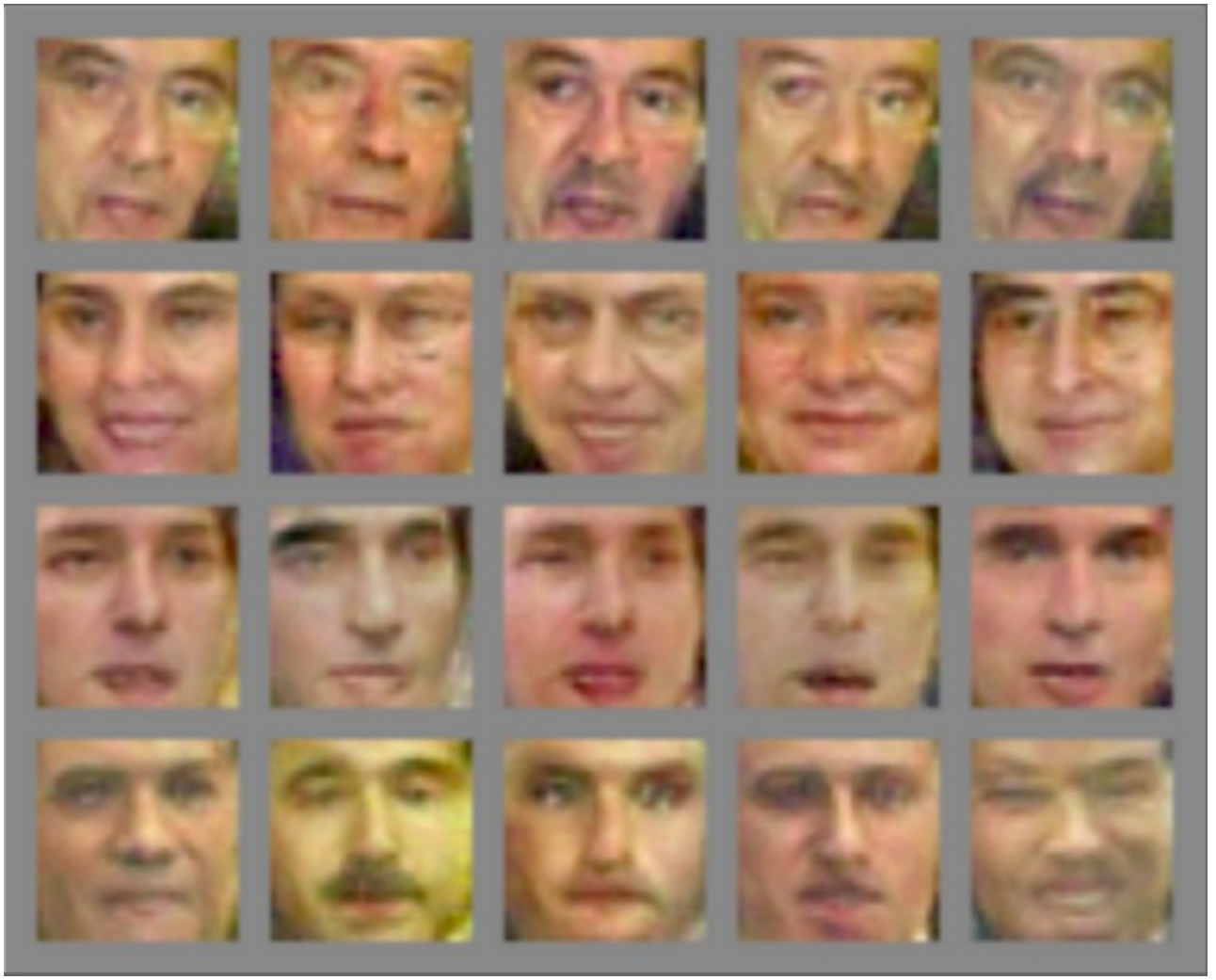
Algorithm

- sample noise **z** and images **x**
- for k in 1...K

  Train discriminator(**x**), discriminator(generator(**z**))
- For m in 1...M

  Train generator(z)

Generative Adversarial Networks



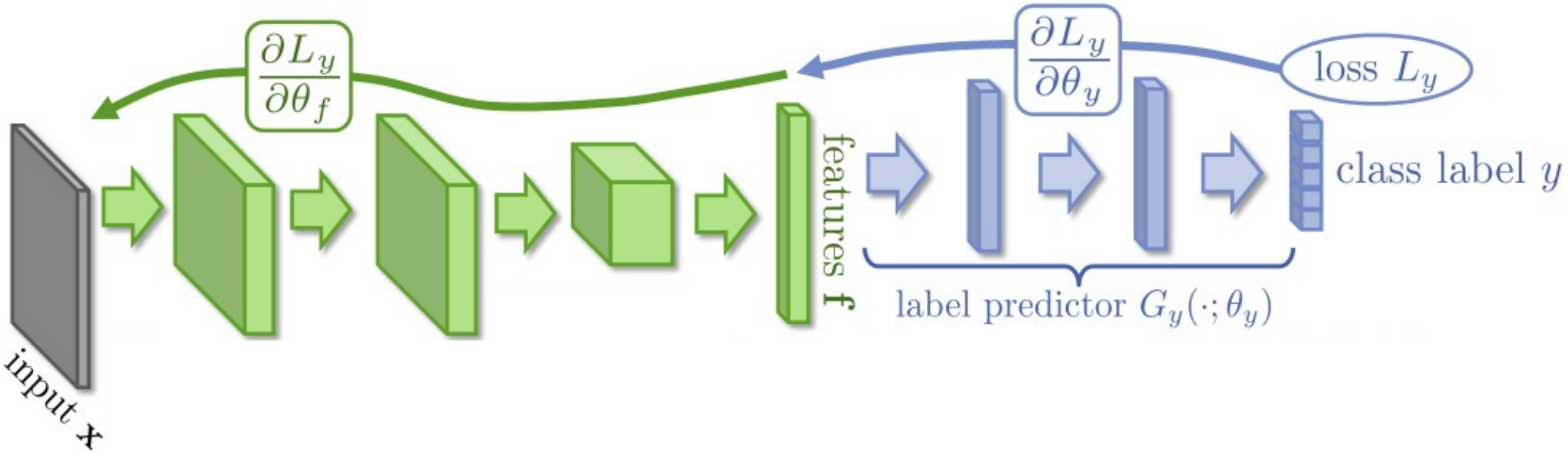
Adversarial domain adaptation

- Two domains
  - e.g. mnist digits Vs actual digits on photos
- First domain is labeled, second isn't
- Wanna learn for the second domain

- Immediate use case: monte-carlo vs real data

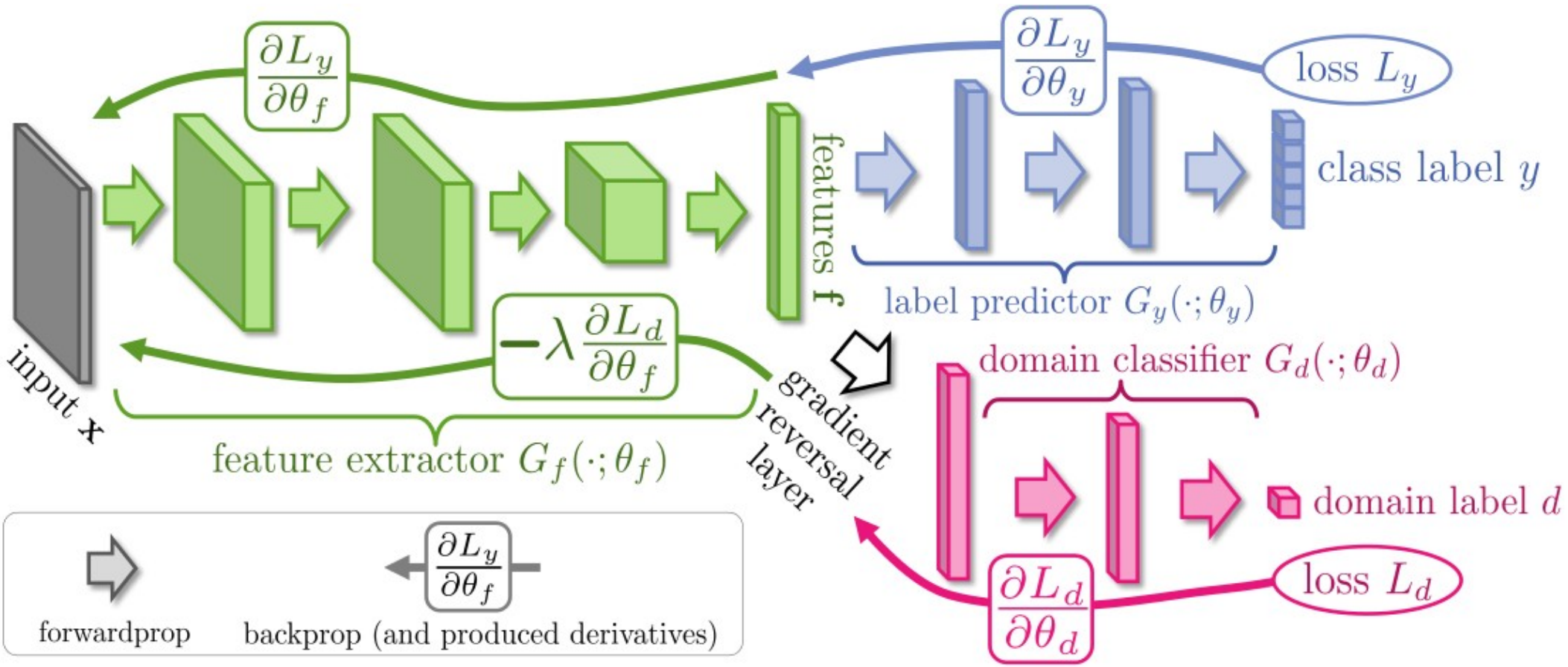
Domain adaptation

- Idea: discriminator should not be able to distinguish features on two domains



Domain adaptation

- Idea: discriminator should not be able to distinguish features on two domains



Domain adaptation

- Idea: discriminator should not be able to distinguish features on two domains

$$-\log P\left(real\middle|h\left(x_{real}\right)\right)-\log\left[1-P\left(real\middle|h\left(x_{mc}\right)\right)\right]\rightarrow \min_{discriminator}$$

$$L_{classifier}\left(y_{mc},y\left(h\left(x_{mc}\right)\right)\right)-\log P\left(real\middle|h\left(x_{mc}\right)\right)\rightarrow \min_{classifier}$$

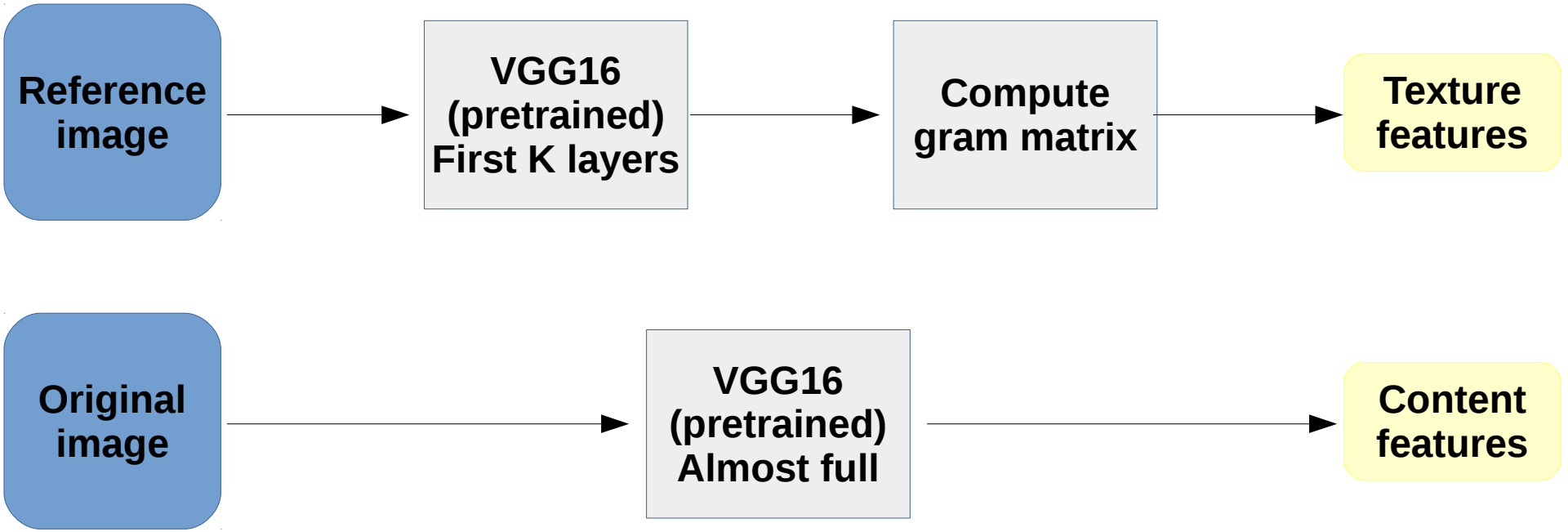
Art style transfer

- Ideas?

Art style transfer

- Formulate and optimize texture loss

$$L = \|Texture(x_{ref}) - Texture(x_{cand})\| + \|Content(x_{orig}) - Content(x_{cand})\|$$



Art style transfer