Deep Image Prior

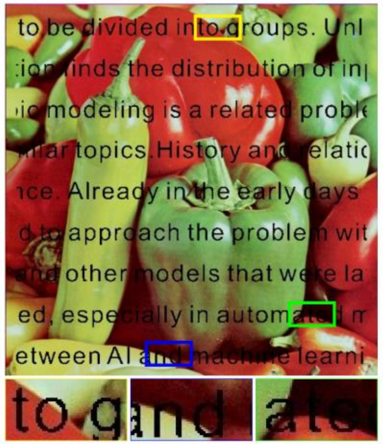
Ulyanov et al. 2017

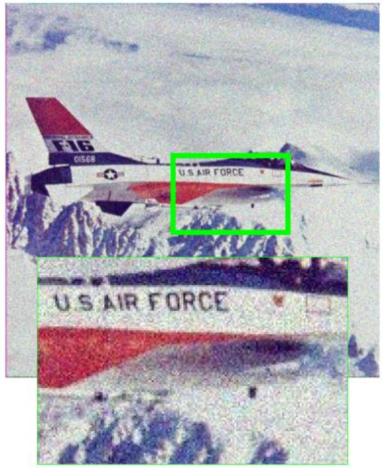
Michael Scherbela
Deep Learning Seminar
Oct 5, 2022

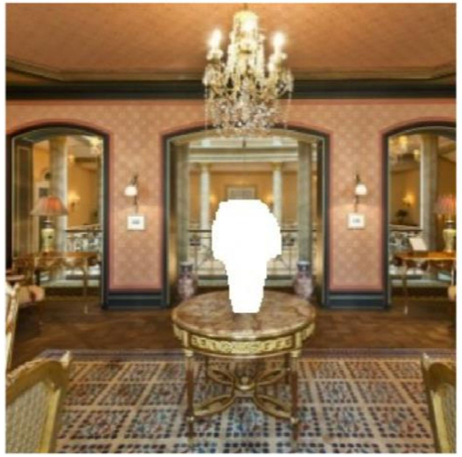
Goal: Reconstruct corrupted images

Text / watermark | Noise | Impainting

Corrupted image

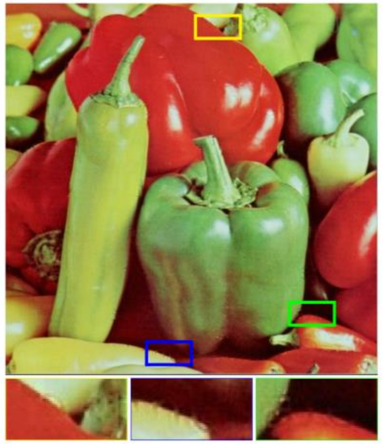






...

Reconstructed image



...

Regularized approach

x_0 … corrupted image

x_{rec} … reconstructed image

$$x_{\text{rec}} = \underset{x}{\operatorname{argmin}}\, d(x_0, x) + R(x)$$

e.g.

$$x_{\text{rec}} = \underset{x}{\operatorname{argmin}} \underbrace{\|x_0 - x\|^2}_{\text{similarity to input image}} + \underbrace{\lambda\, TV(x)}_{\text{regularizer / prior}}$$

Regularizer $\boldsymbol{R(x)}$ judges how „image-like“ $\boldsymbol{x}$ is:

- Hand-crafted (e.g. total variation)
- Learned by training (e.g. on ImageNet)

$$TV(x) = \sum_{ij} \sqrt{\left|x_{i,j} - x_{i+1,j}\right|^2 + \left|x_{i,j} - x_{i,j+1}\right|^2}$$

Regularized approach

x_0 … corrupted image

x_{rec} … reconstructed image

$$x_{\text{rec}} = \underset{x}{\operatorname{argmin}}\, d(x_0, x) + R(x)$$

e.g.

$$x_{\text{rec}} = \underset{x}{\operatorname{argmin}} \underbrace{\|x_0 - x\|^2}_{\text{similarity to input image}} + \underbrace{\lambda\, TV(x)}_{\text{regularizer / prior}}$$

Regularizer $R(x)$ judges how „image-like" x is:

- Hand-crafted (e.g. total variation)
- Learned by training (e.g. on ImageNet)

$$TV(x) = \sum_{ij} \sqrt{\left|x_{i,j} - x_{i+1,j}\right|^2 + \left|x_{i,j} - x_{i,j+1}\right|^2}$$

Deep Image Prior

f_θ … conv-net

z … fixed random input

$$\theta_{\text{rec}} = \underset{\theta}{\operatorname{argmin}}\, d\big(x_0, f_\theta(z)\big)$$

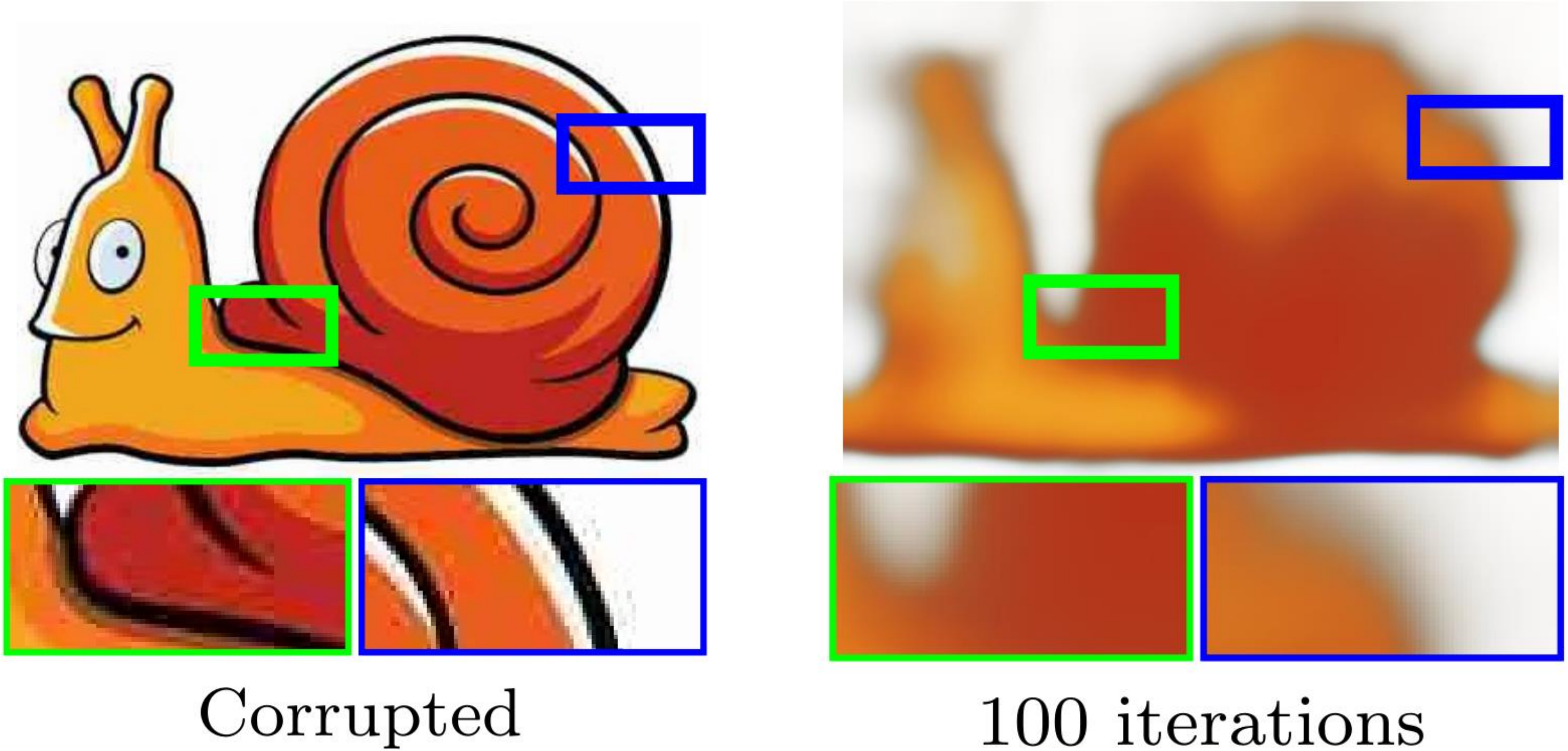
$$x_{\text{rec}} = f_\theta(z)$$

Regularization through structure / biases in function f_θ

Optimization in parameter space θ, instead of pixel space x

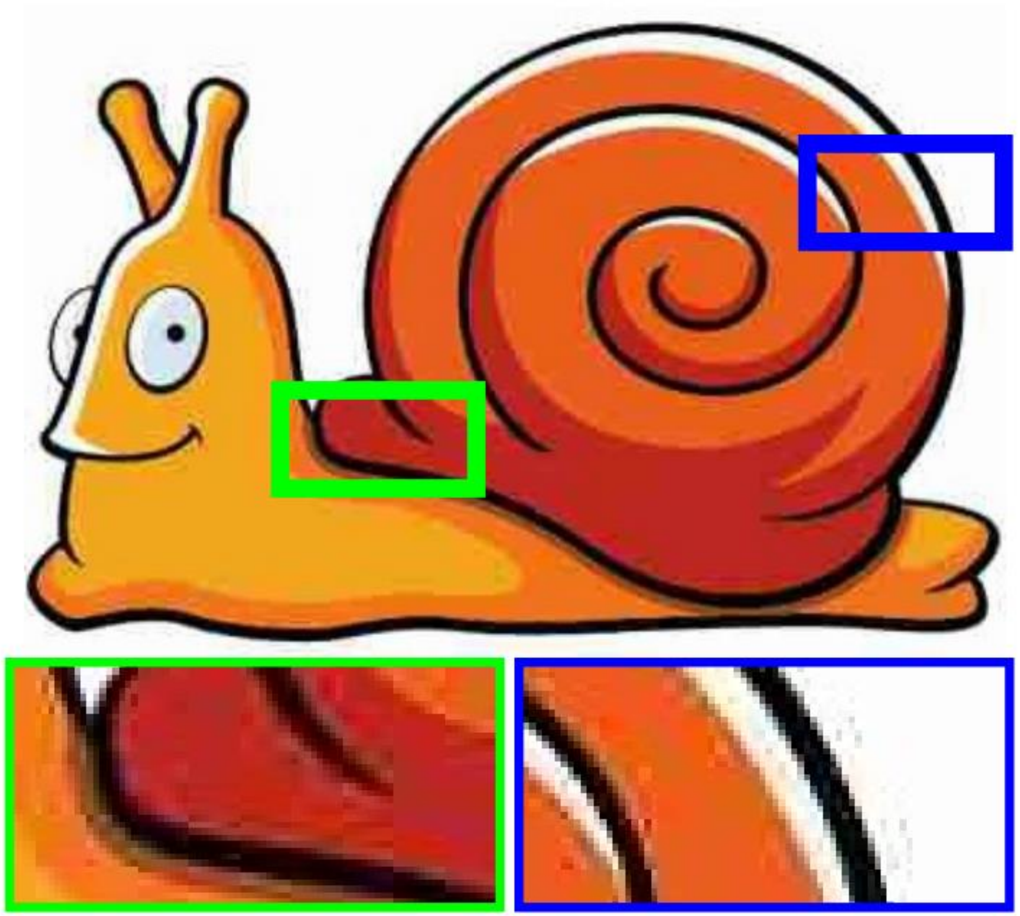
Training requires early stopping, otherwise corrupted image is reproduced

Example: JPEG artifact removal

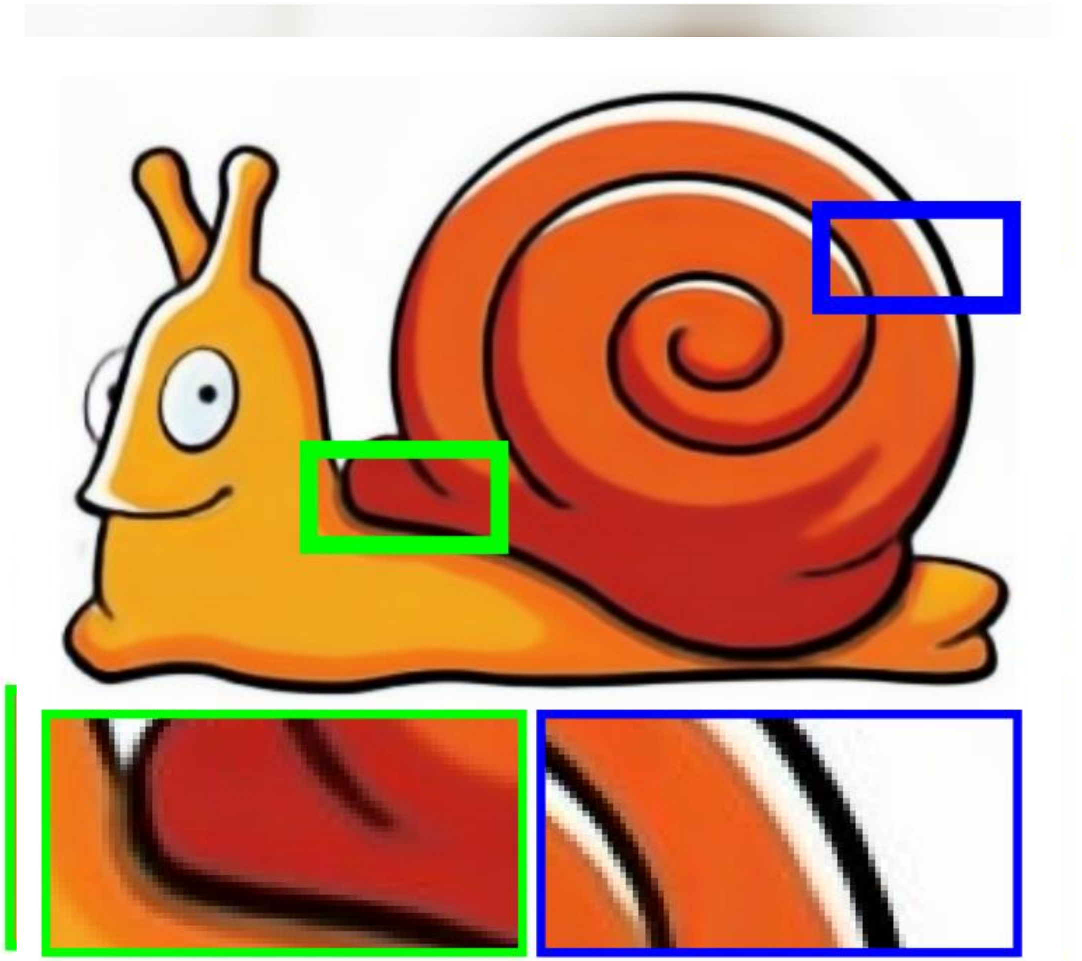


Training requires early stopping, otherwise corrupted image is reproduced

Example: JPEG artifact removal



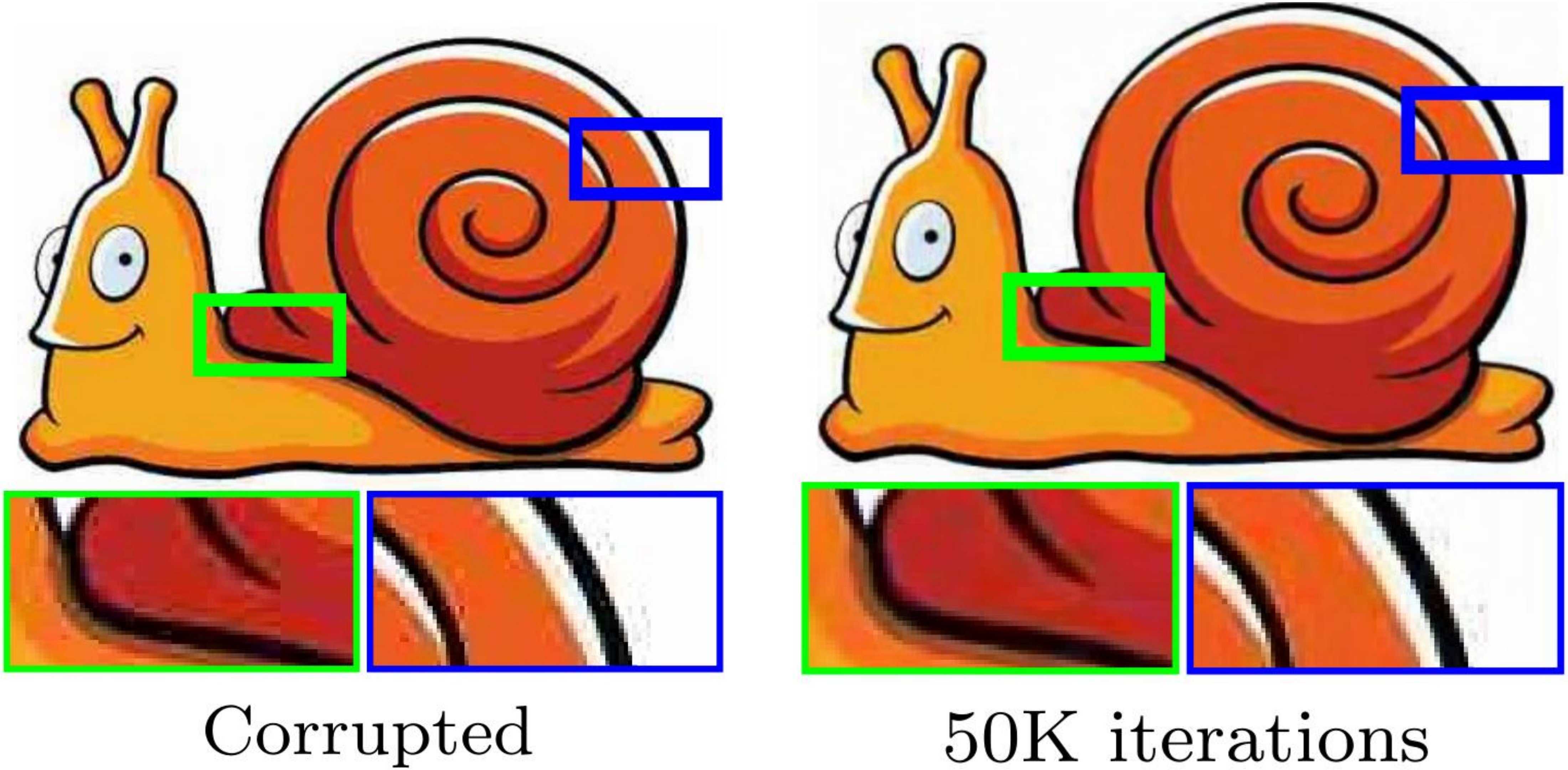
Corrupted



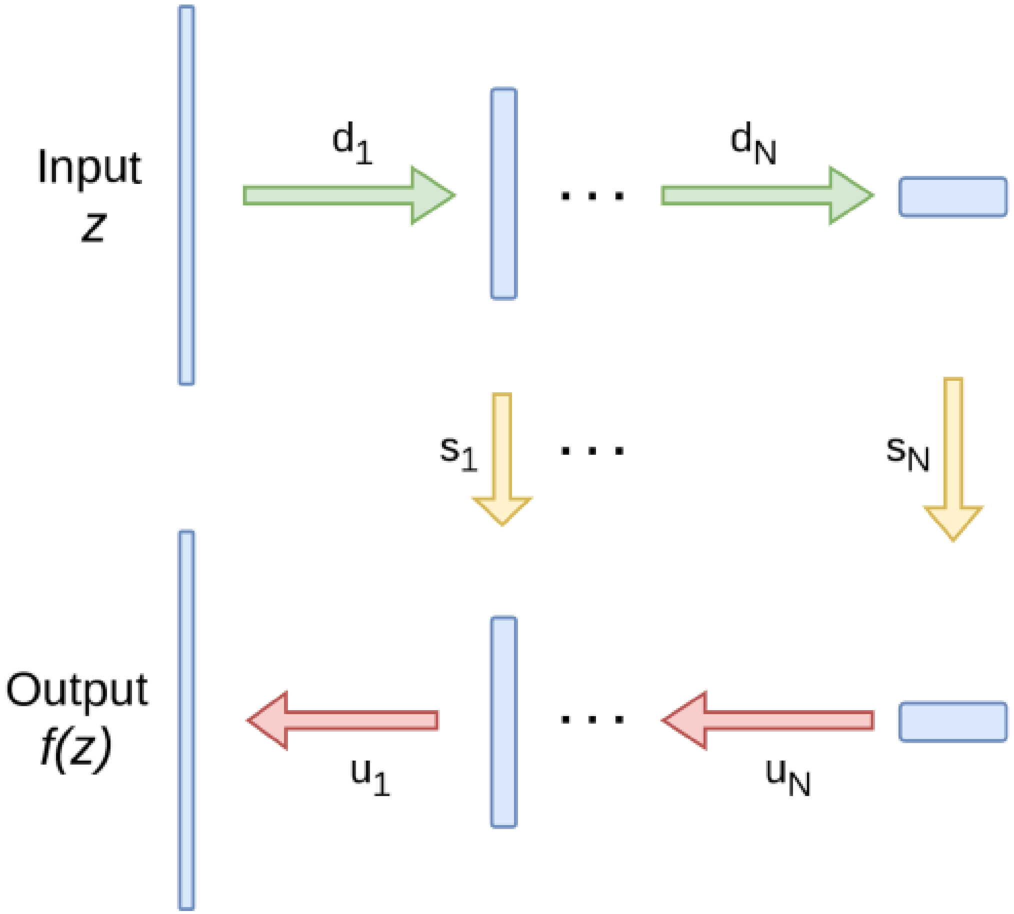
2400 iterations

Training requires early stopping, otherwise corrupted image is reproduced

Example: JPEG artifact removal



Used architecture: Encoder + decoder / U-net



Architecture

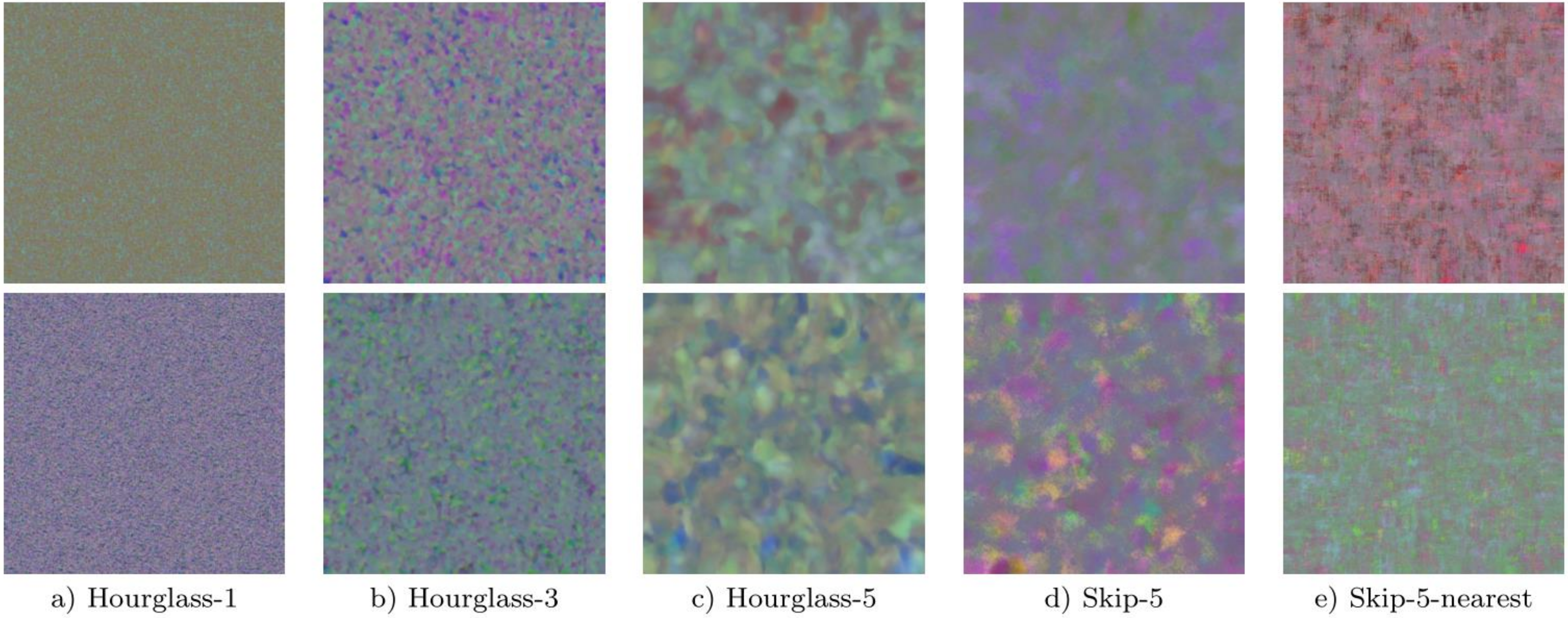
- 5-6 downsampling layers + 5-6 upsampling layers
- Each layer 2x:
  - Convolution
  - Batch-Norm
  - Leaky ReLU
  - No skip connection / residual
- Downsampling: Strides during first convolution
- Upsampling: nearest / bilinear

Hyperparameters

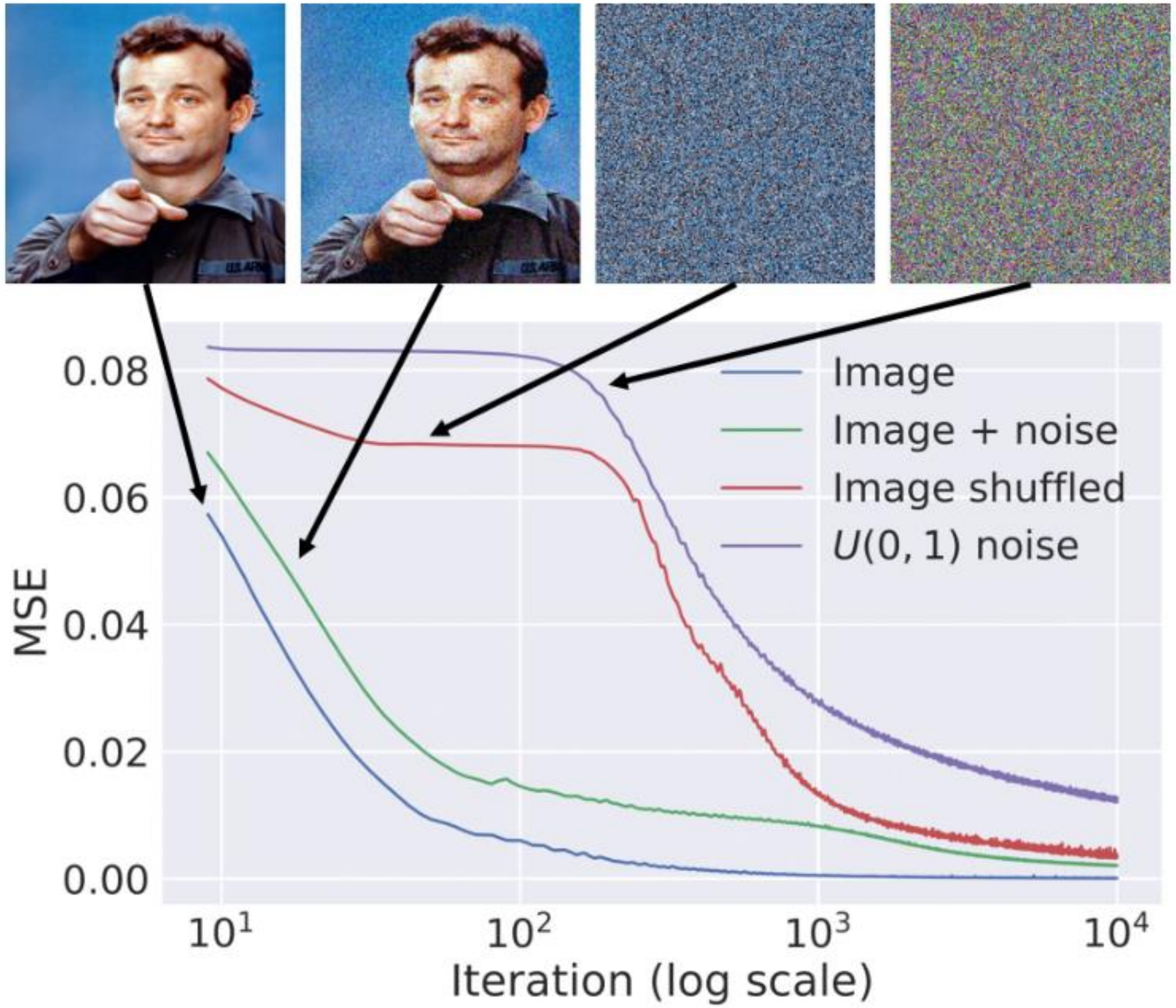
- Similar across tasks, but slightly tuned
- 2000 – 3000 optimization steps
- Adam: LR = 10^{-1} - 10^{-3}

Why it works: U-nets produce image-like outputs

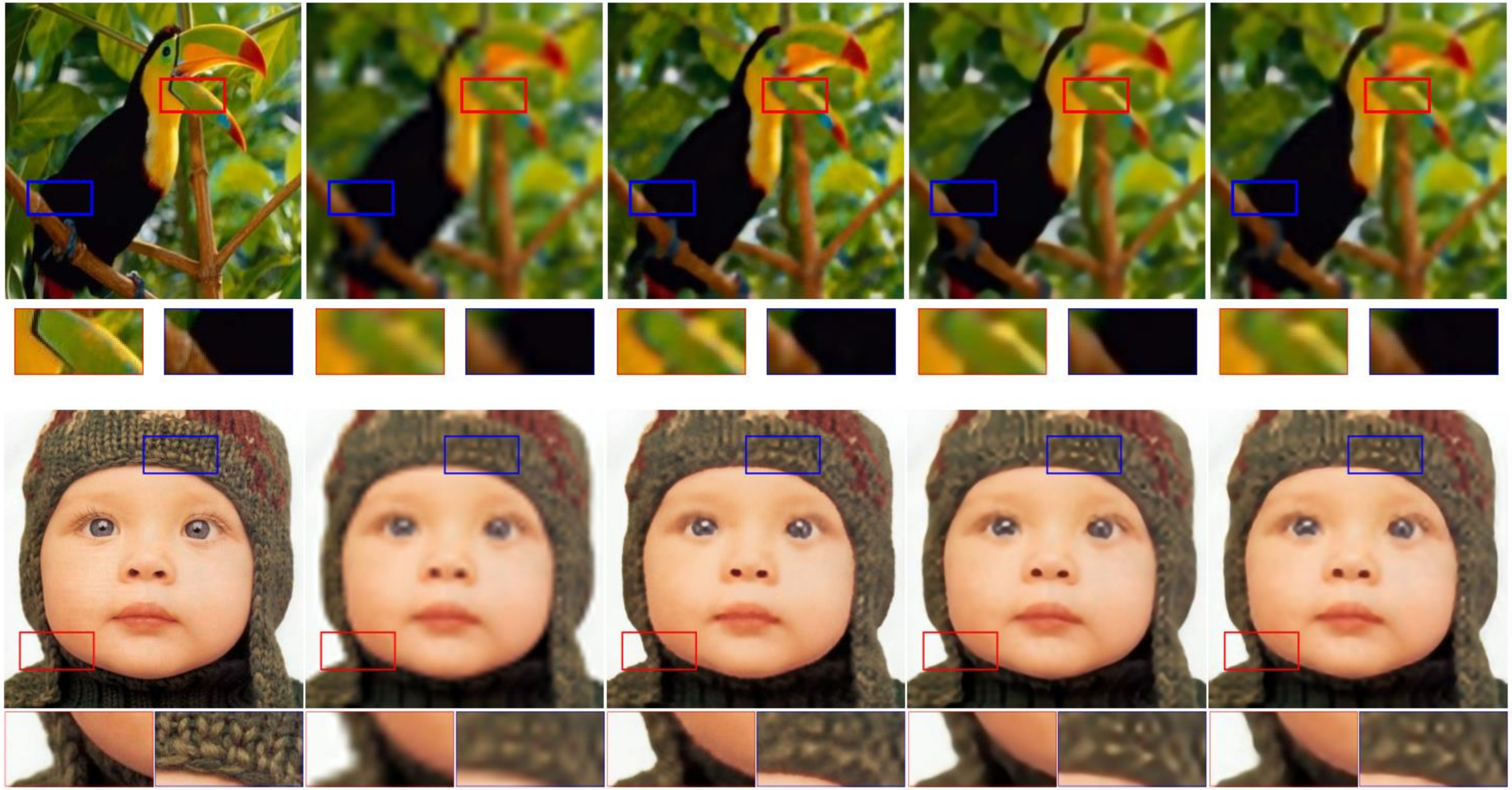
Outputs of untrained, randomly intialized U-nets $f_{\theta_0}(z)$



Why it works: U-nets can easily model images, but struggle to reproduce high-frequency noise



Pretty pictures: Super-resolution



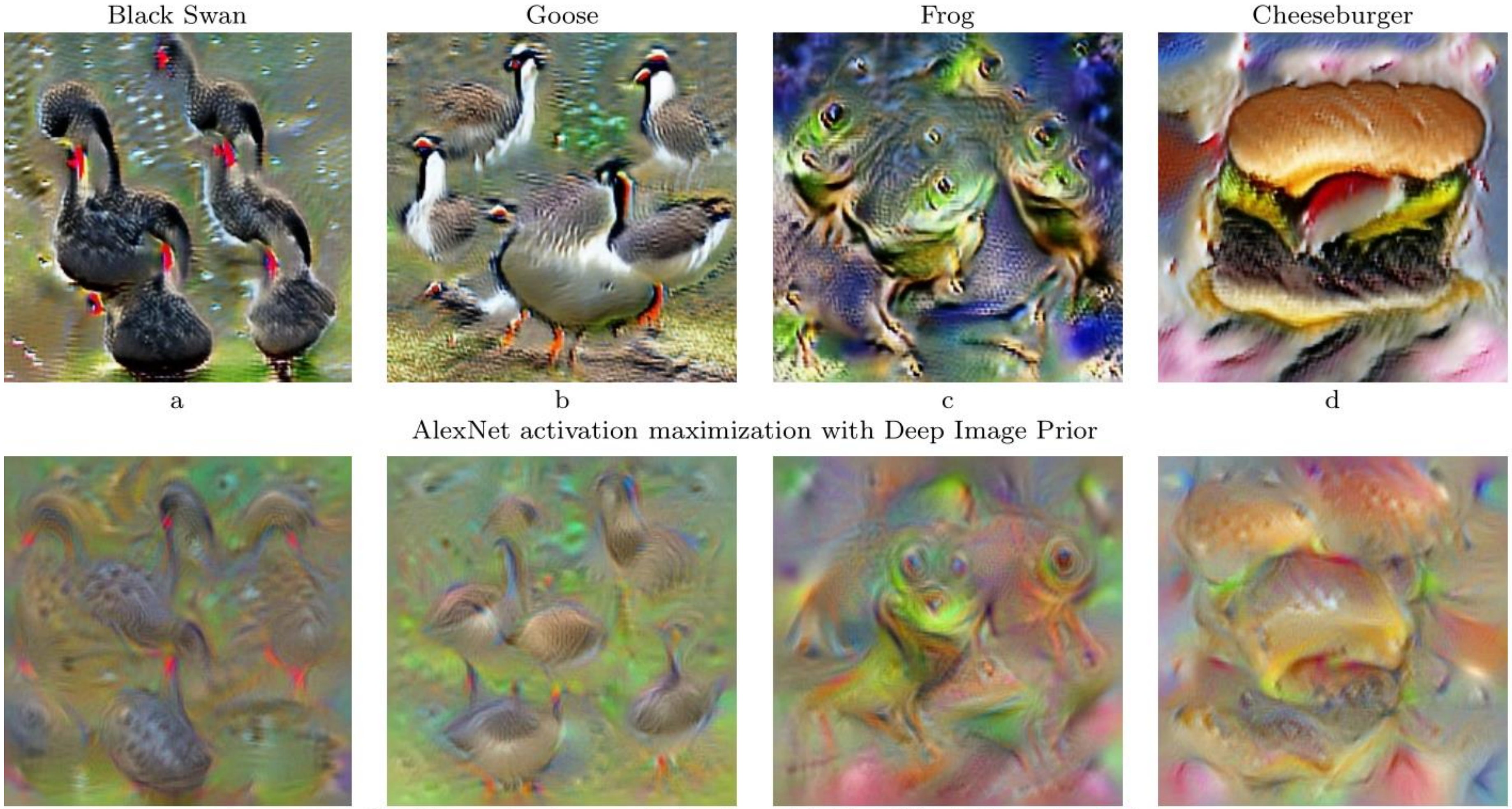
(f) Original image (g) Bicubic, **Not trained** (h) Ours, **Not trained** (i) LapSRN, **Trained** (j) VDSR, **Trained**

Pretty picture: Activation maximization for AlexNet

Which input looks most like a Cheesburger?

Deep Image Prior

Total Variation Prior



AlexNet activation maximization with Deep Image Prior

AlexNet activation maximization with Total Variation prior [38]

Pretty pictures: Inpainting

(a) Input (white=masked)

(b) Encoder-decoder, depth=6

Pretty pictures: Inpainting

(a) Input (white=masked)

(e) ResNet, depth=8

Adding skip connections (i.e. ResNet) deteriorates image prior

Unexpected Overlap

> “… Deep image prior is also used in [6] to perform phase retrieval for Fourier ptychography.”

[6] Boominathan, L., Maniparambil, M., Gupta, H., Baburajan, R., Mitra, K.: Phase retrieval for fourier ptychography under varying amount of measurements. CoRR (2018)