GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Deep Learning

Lecture 10: Image-to-Image models

Alexander Ecker
Institut für Informatik, Uni Göttingen

alxecker

https://alexanderecker.wordpress.com

– Credit: some of the slides based on Fei Fei Li, Justin Johnson and Serena Yeung's slides –

Today’s topics

Last homework

Current homework

U-net for medial image segmentation

Fast style transfer and super resolution

Unpaired domain translation

Reparameterization trick

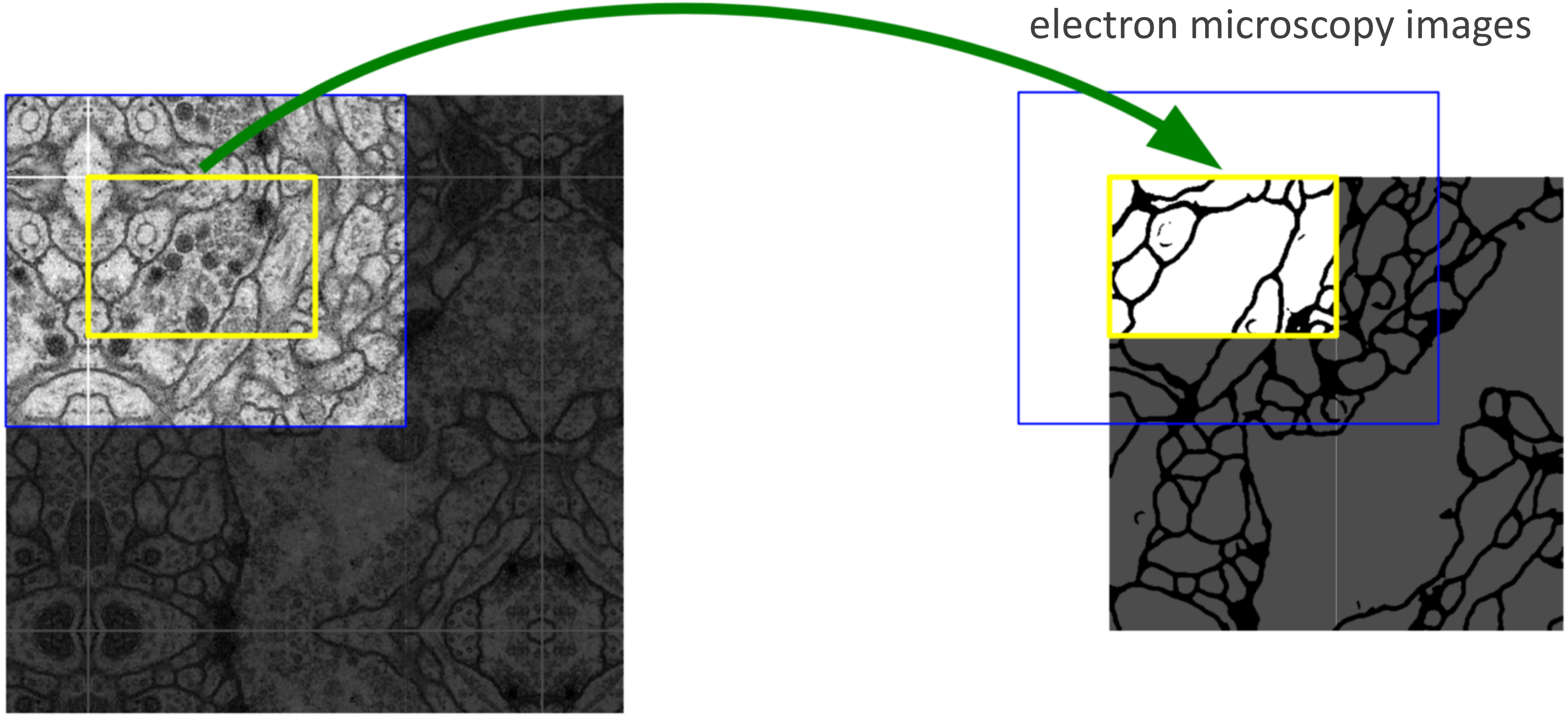
To generate $\mathrm{x} \sim \mathcal{N}(\mu, \sigma^2)$, sample $\epsilon \sim \mathcal{N}(0, 1)$:

$$x = \mu + \sigma\epsilon$$

U-net: biomedical image segmentation

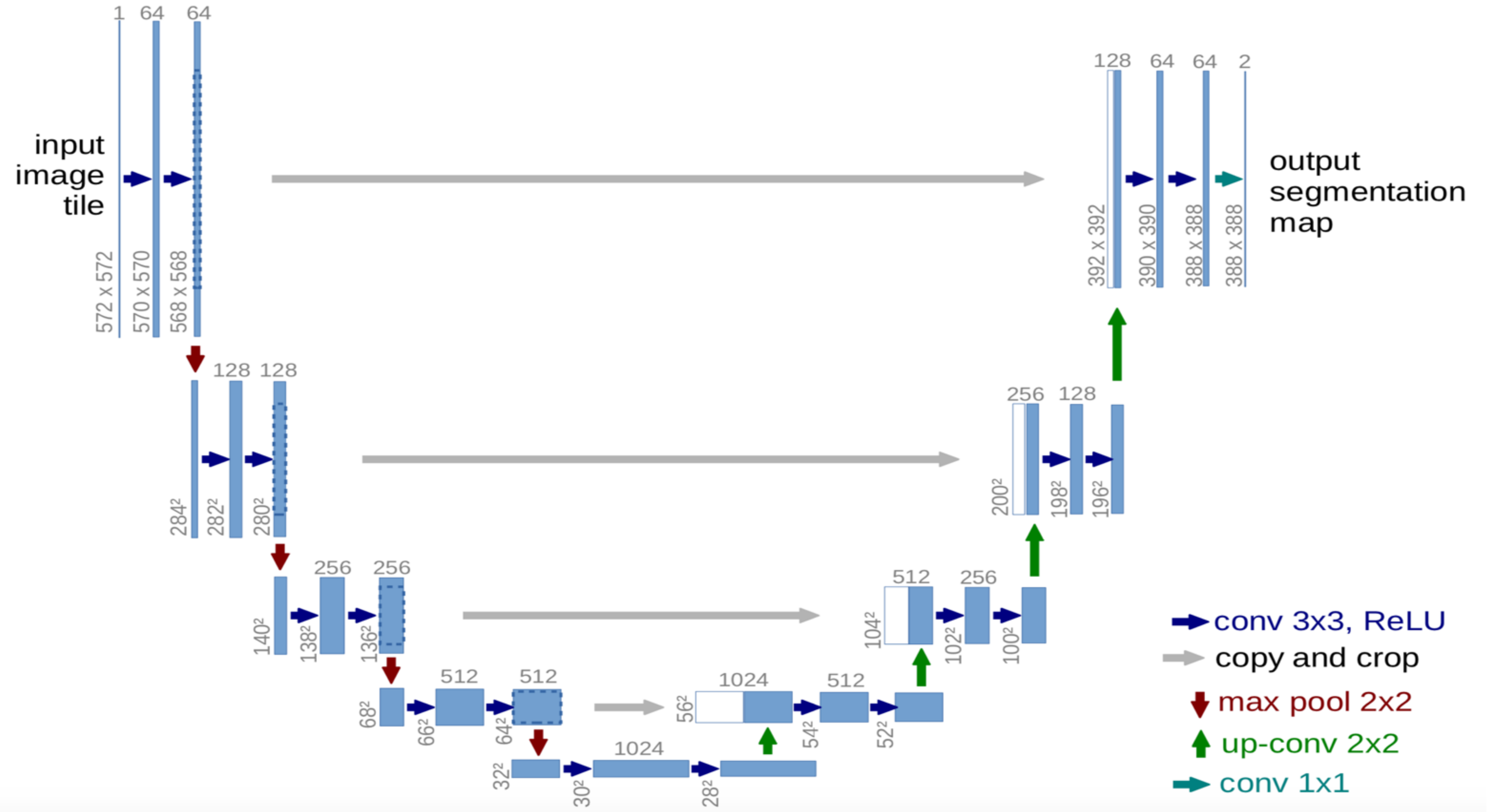
Biomedical image segmentation

Task: segment neuronal structures from electron microscopy images



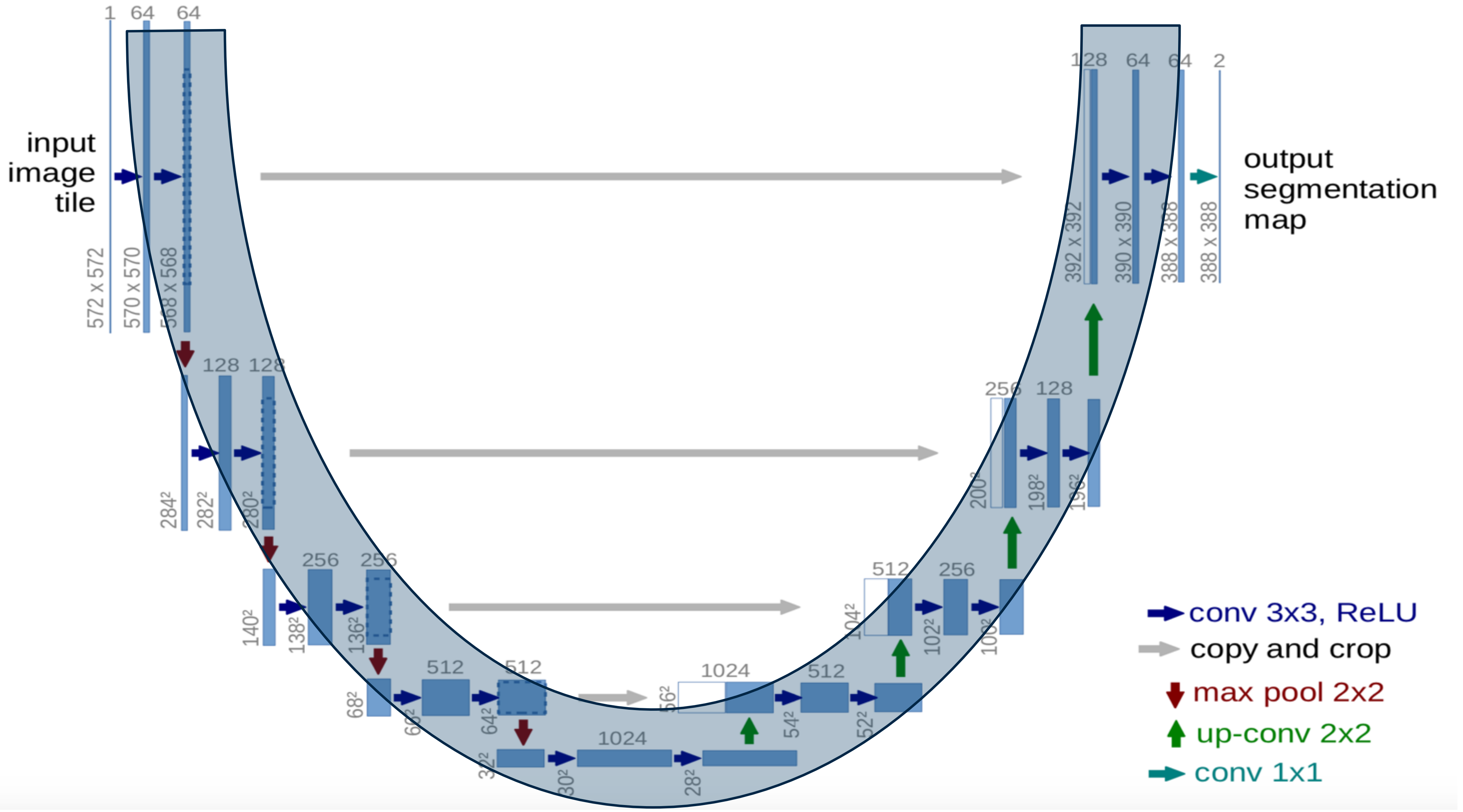
Ronneberger, Fischer, Brox, *MICCAI* 2015

U-net



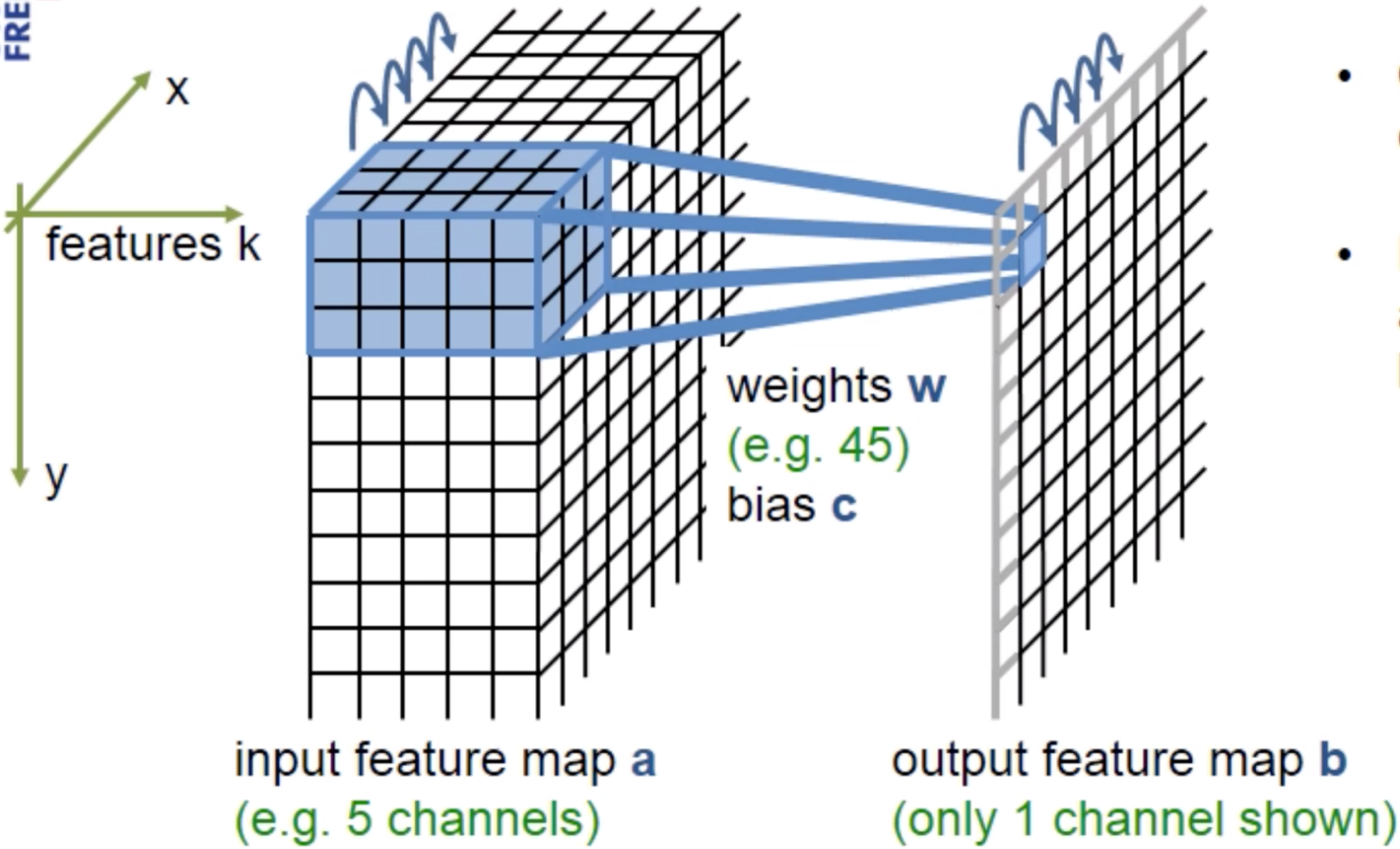
Ronneberger, Fischer, Brox, *MICCAI* 2015

U-net

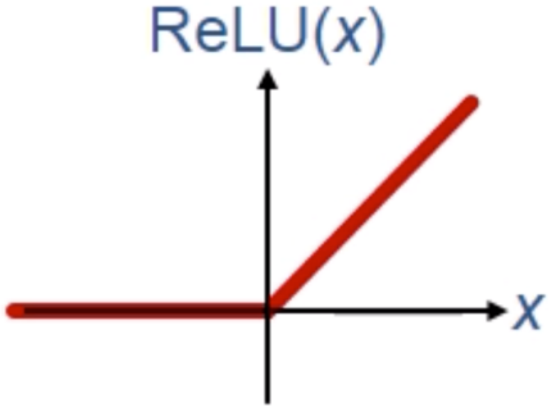


Ronneberger, Fischer, Brox, *MICCAI* 2015

3x3 convolution + ReLU

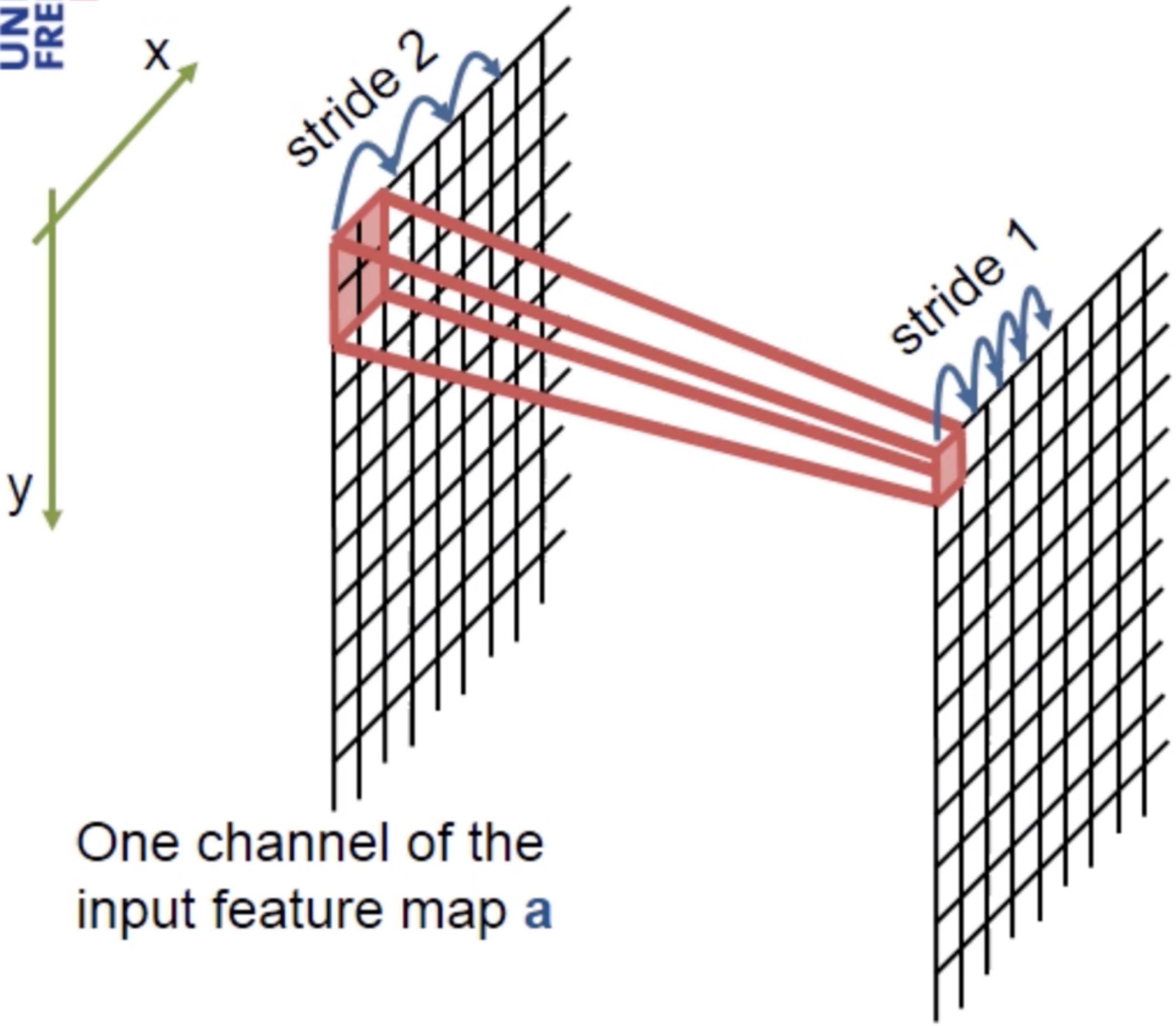


- Only valid part of convolution is used.
- For 3x3 convolutions a 1-pixel border is lost



Slide credit: Ronneberger, Fischer, Brox

2x2 max-pooling



One channel of the input feature map **a**

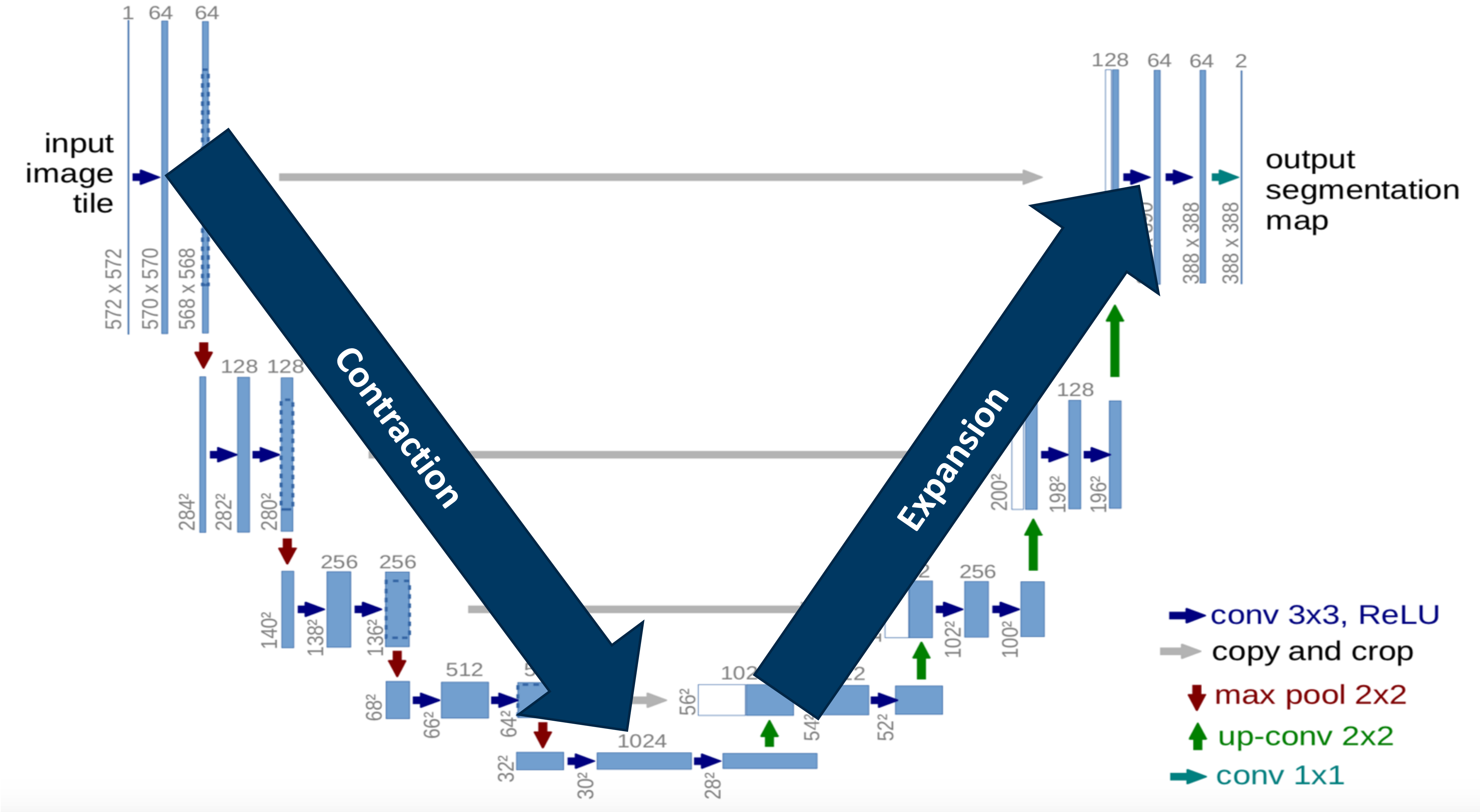
One channel of the output feature map **b**

resulting feature map has factor 2 lower spatial resolution

$$b_{x,y,k} = \max_{\substack{i\in\{0,1\} \\ j\in\{0,1\}}} \left(a_{2x+i,2y+j,k} \right)$$

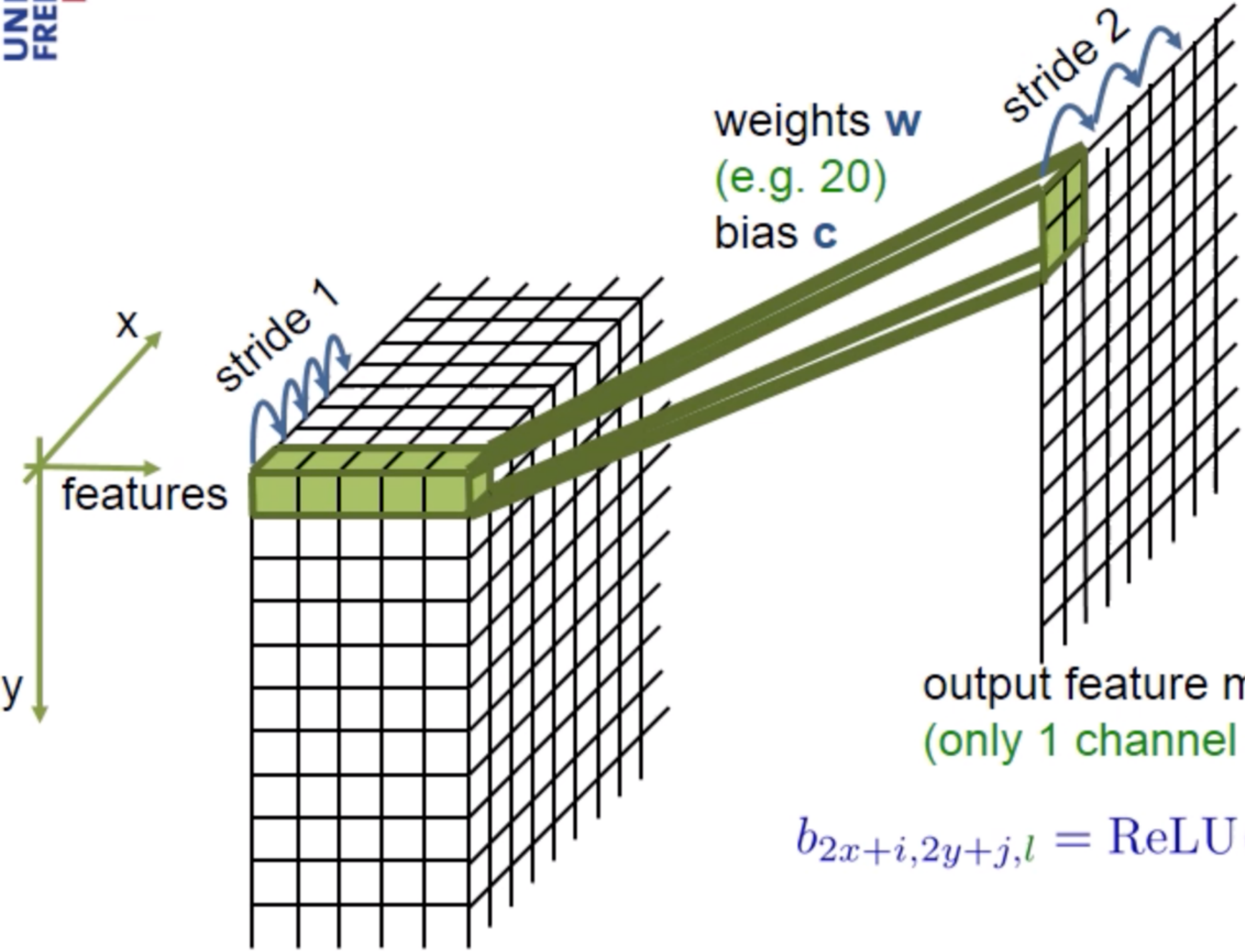
Slide credit: Ronneberger, Fischer, Brox

U-net



Ronneberger, Fischer, Brox, *MICCAI* 2015

2x2 up-convolution



resulting feature maps have factor 2 higher resolution

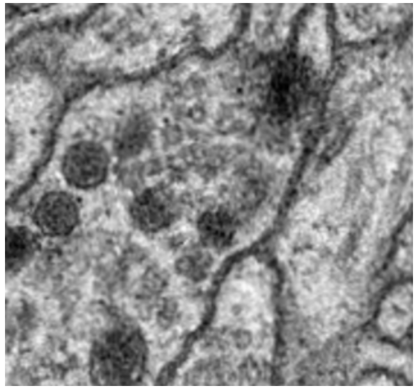
$$b_{2x+i,2y+j,l} = \mathrm{ReLU}\Big(\sum_{\substack{i\in\{0,1\} \\ j\in\{0,1\} \\ k\in\{1,\cdots,K\}}} w_{i,j,k,l} \cdot a_{x,y,k} + c_l \Big)$$

input feature map **a**
(e.g. 5 channels)

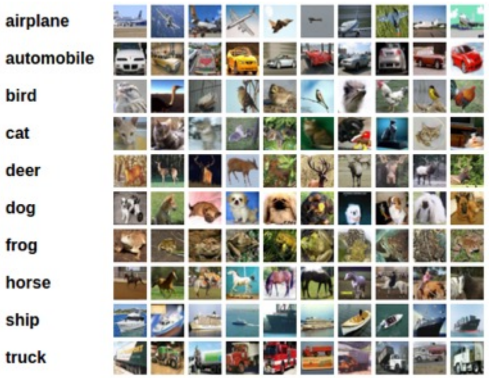
Slide credit: Ronneberger, Fischer, Brox

U-net: training

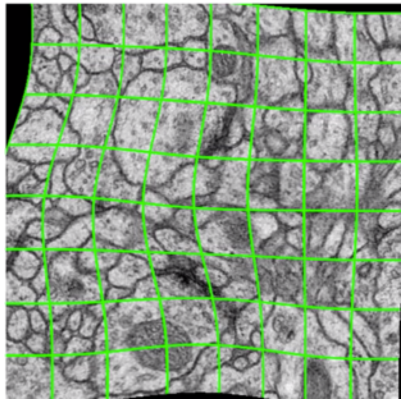
No pre-training on ImageNet



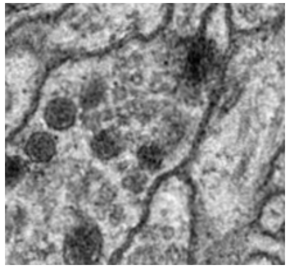
≠

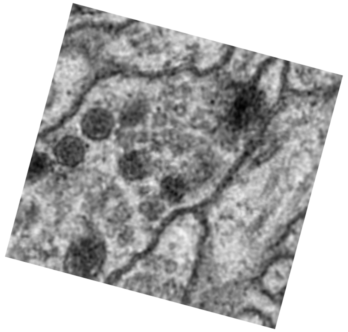


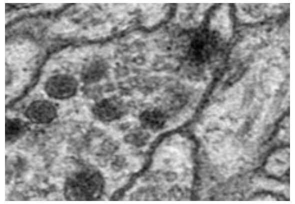
Elastic image deformations

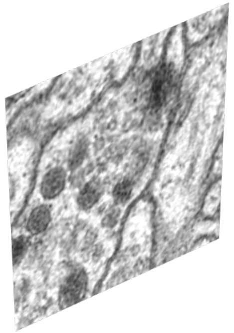


Instead: heavy data augmentation









...

Fast style transfer

Recap: neural style transfer

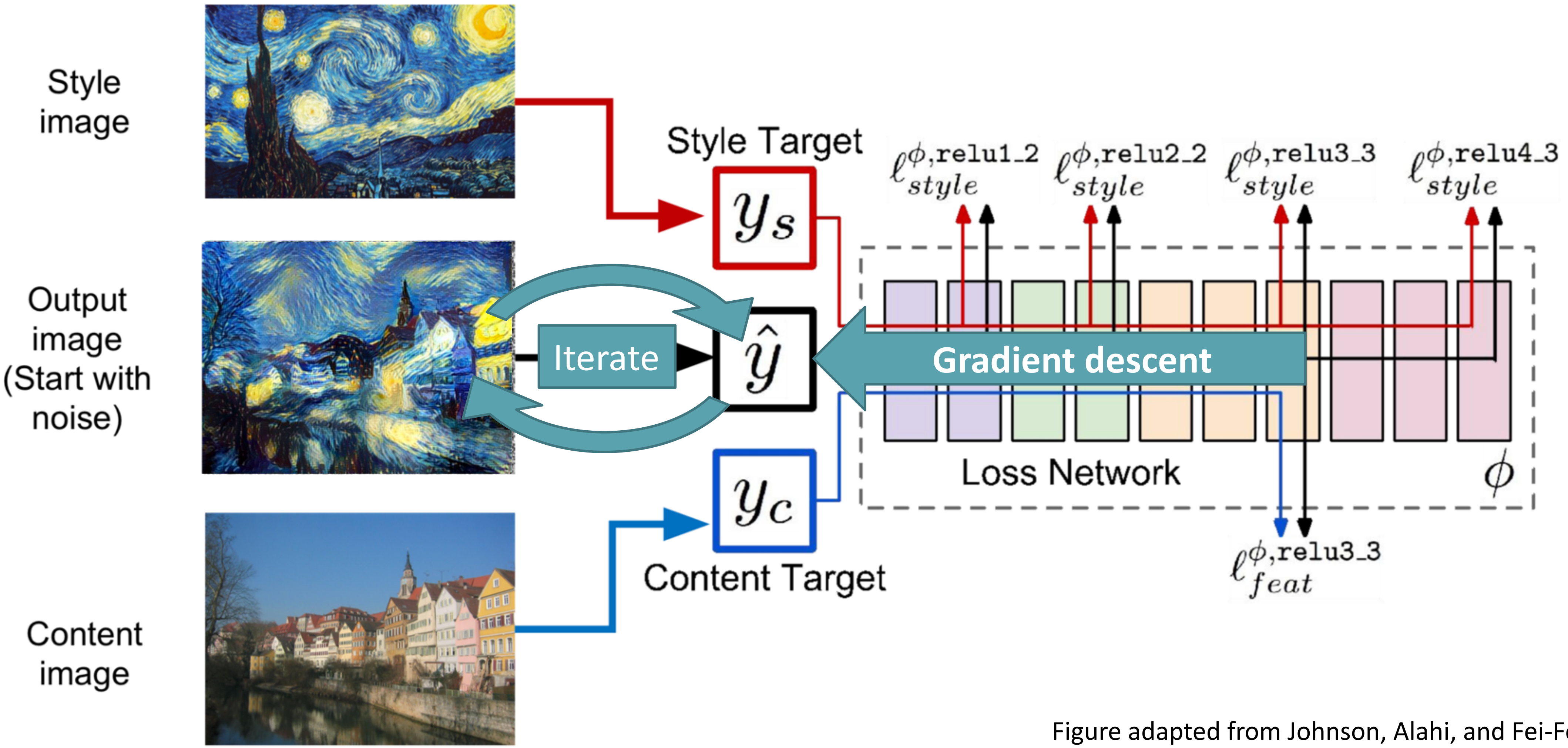
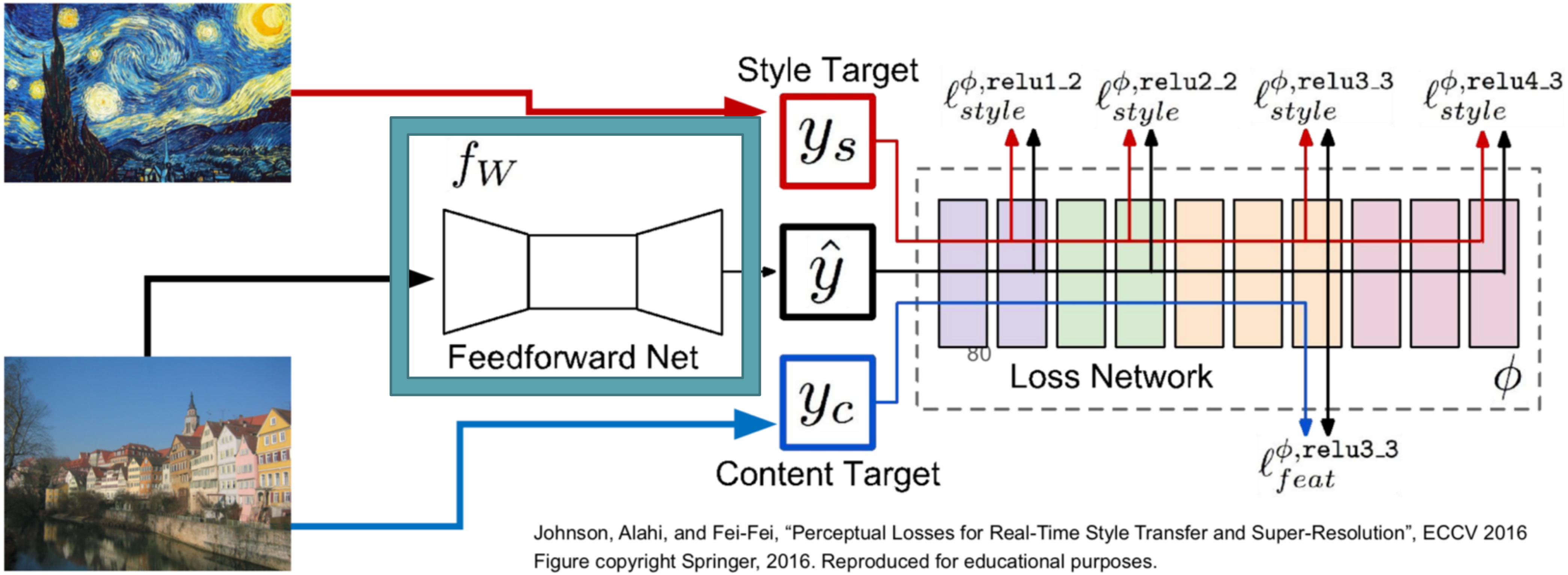


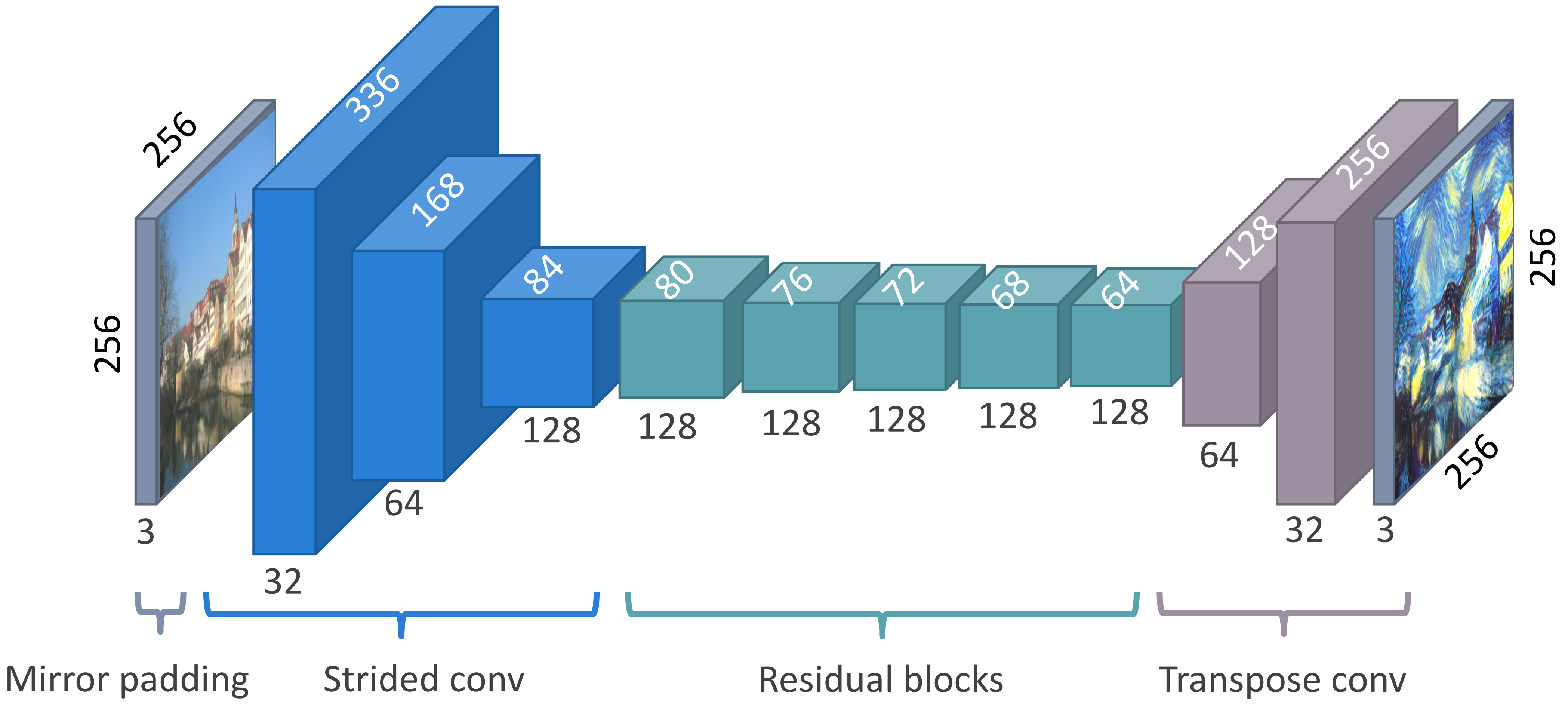
Figure adapted from Johnson, Alahi, and Fei-Fei, *ECCV* 2016

Fast style transfer



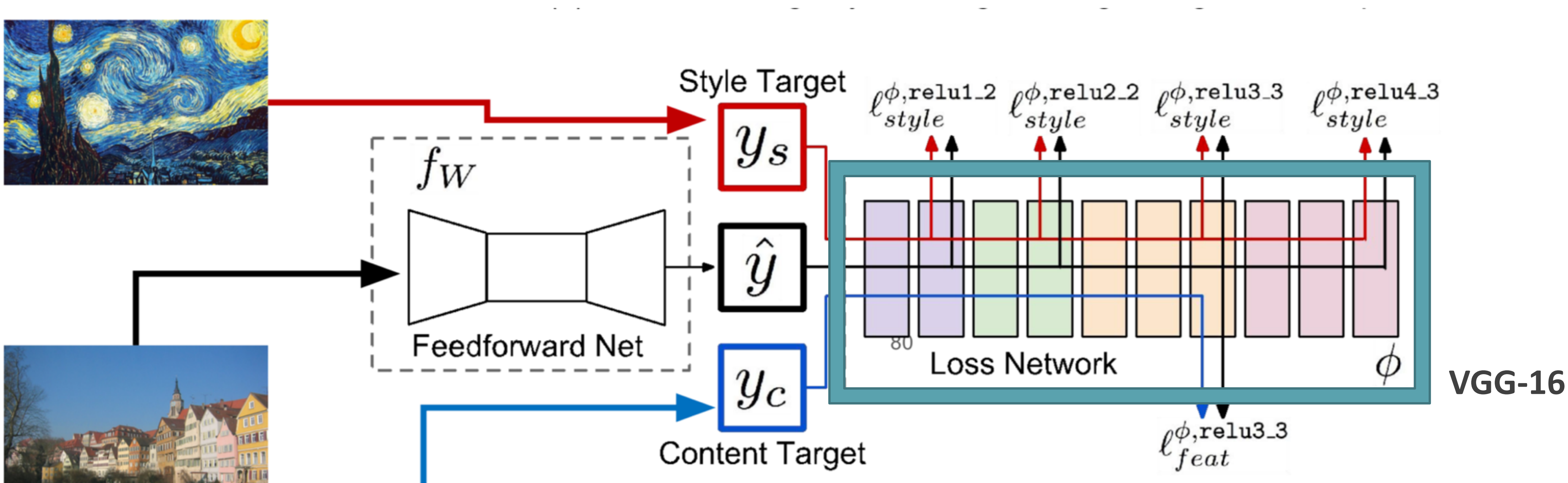
Johnson, Alahi, and Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution", ECCV 2016

Fast style transfer: network architecture



Johnson, Alahi, and Fei-Fei, *ECCV* 2016

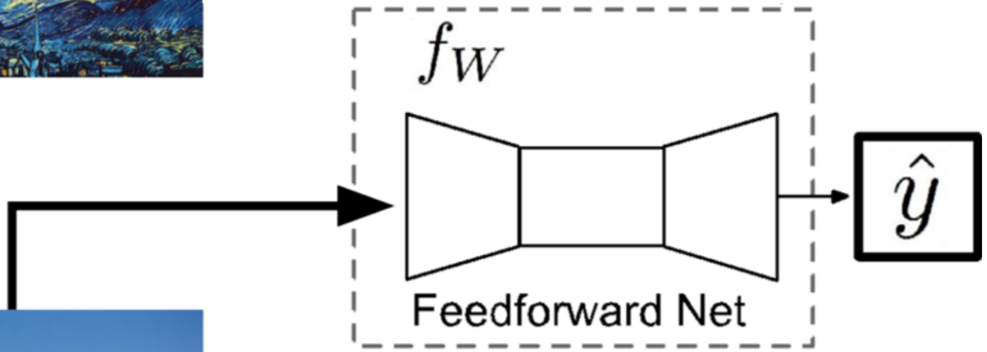
Fast style transfer: training



Johnson, Alahi, and Fei-Fei, "Perceptual Losses for Real-Time Style Transfer and Super-Resolution", ECCV 2016
Figure copyright Springer, 2016. Reproduced for educational purposes.

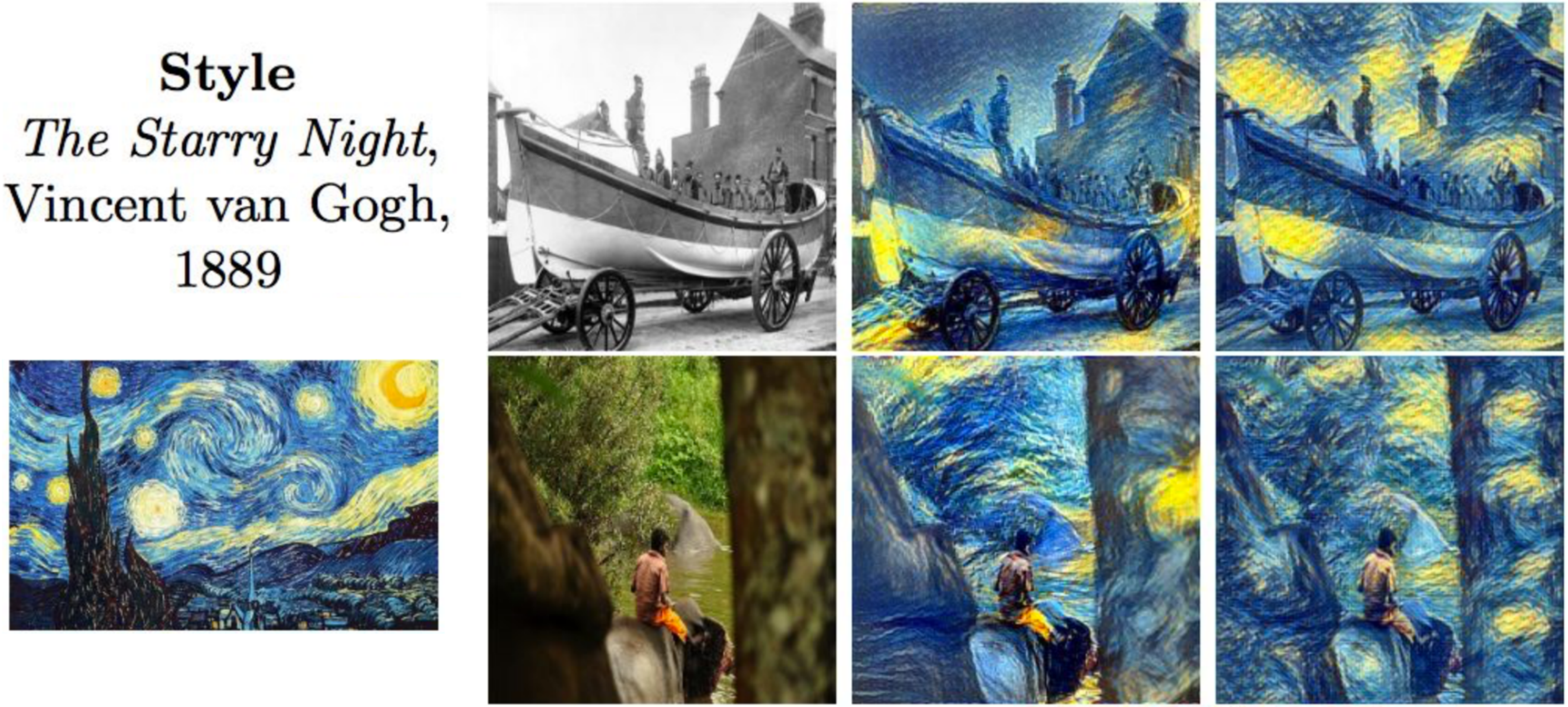
Fast style transfer: image generation

Style target is fixed: need to train one network for each style



Johnson, Alahi, and Fei-Fei, *ECCV* 2016

Fast style transfer: results



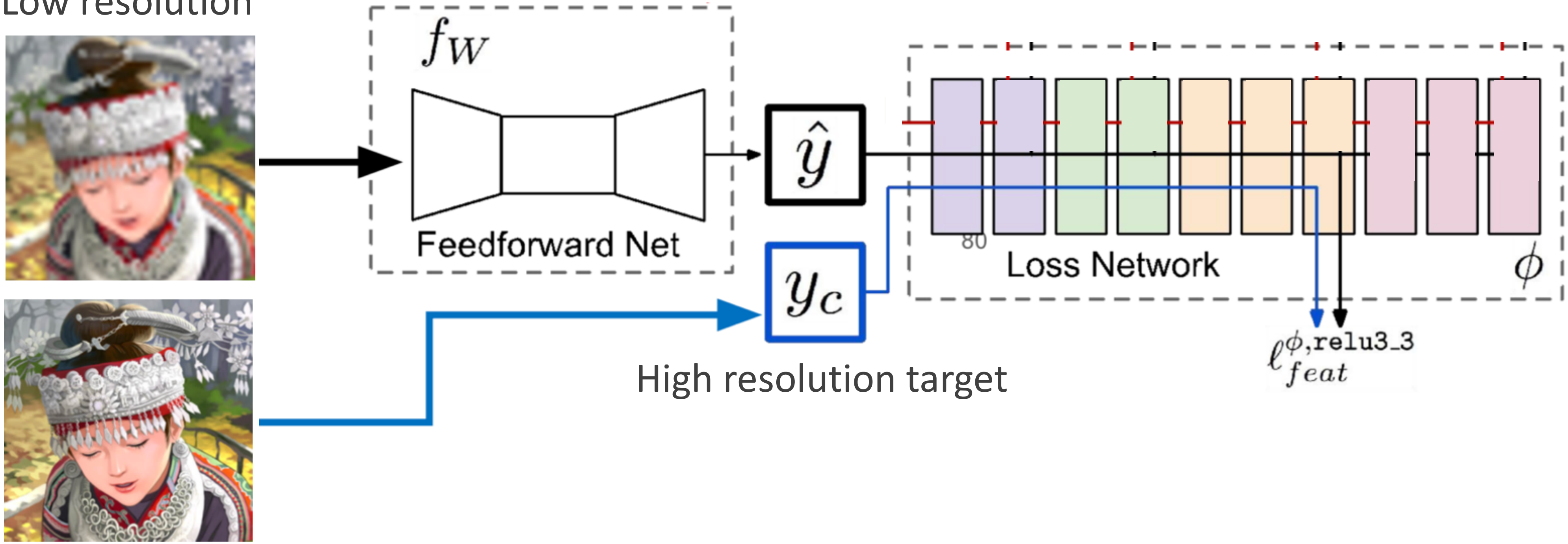
Johnson, Alahi, and Fei-Fei, *ECCV* 2016

Fast style transfer: results

Johnson, Alahi, and Fei-Fei, *ECCV* 2016

Super resolution: same net + content loss

Low resolution



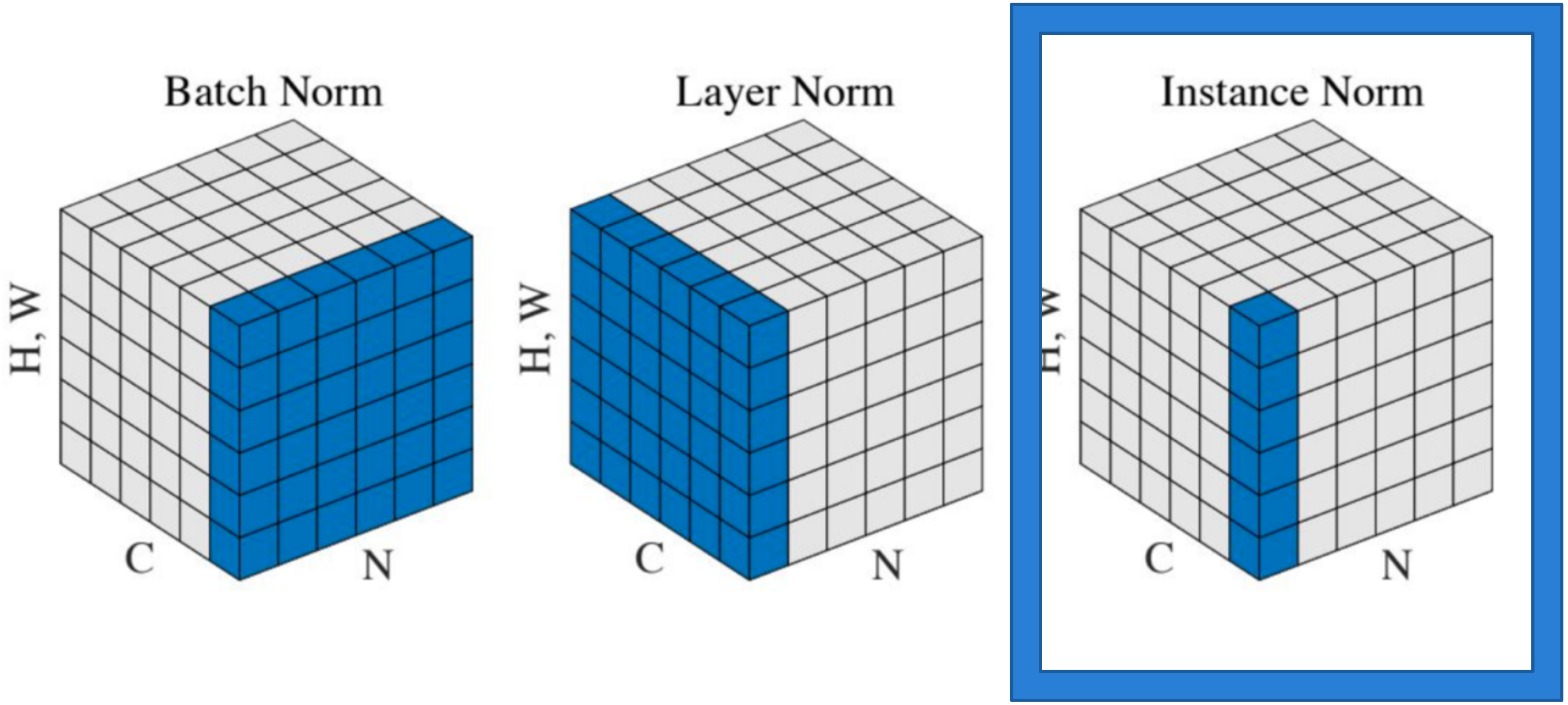
High resolution

Johnson, Alahi, and Fei-Fei, *ECCV* 2016

Super resolution:
Perecptual loss delivers strong performance

Johnson, Alahi, and Fei-Fei, *ECCV* 2016

Instance normalization was invented for fast style transfer



Ulyanov, Vedaldi, Lempitsky, arXiv 2016

Slide credit: Fei Fei Li, Justin Johnson, Serena Yeung

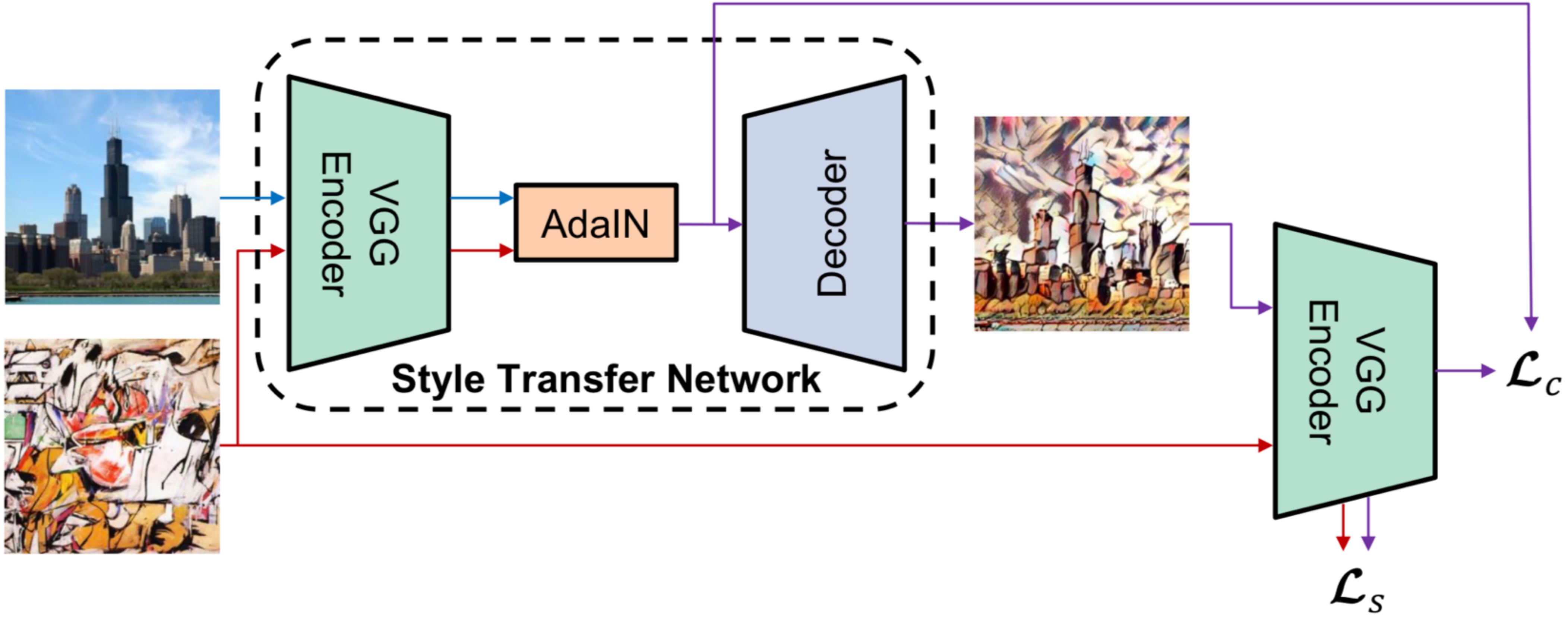
Fast style transfer with instance norm

Same architecture as before

Replacing batch norm by Instance norm improves stylization results

Ulyanov, Vedaldi, Lempitsky, arXiv 2016

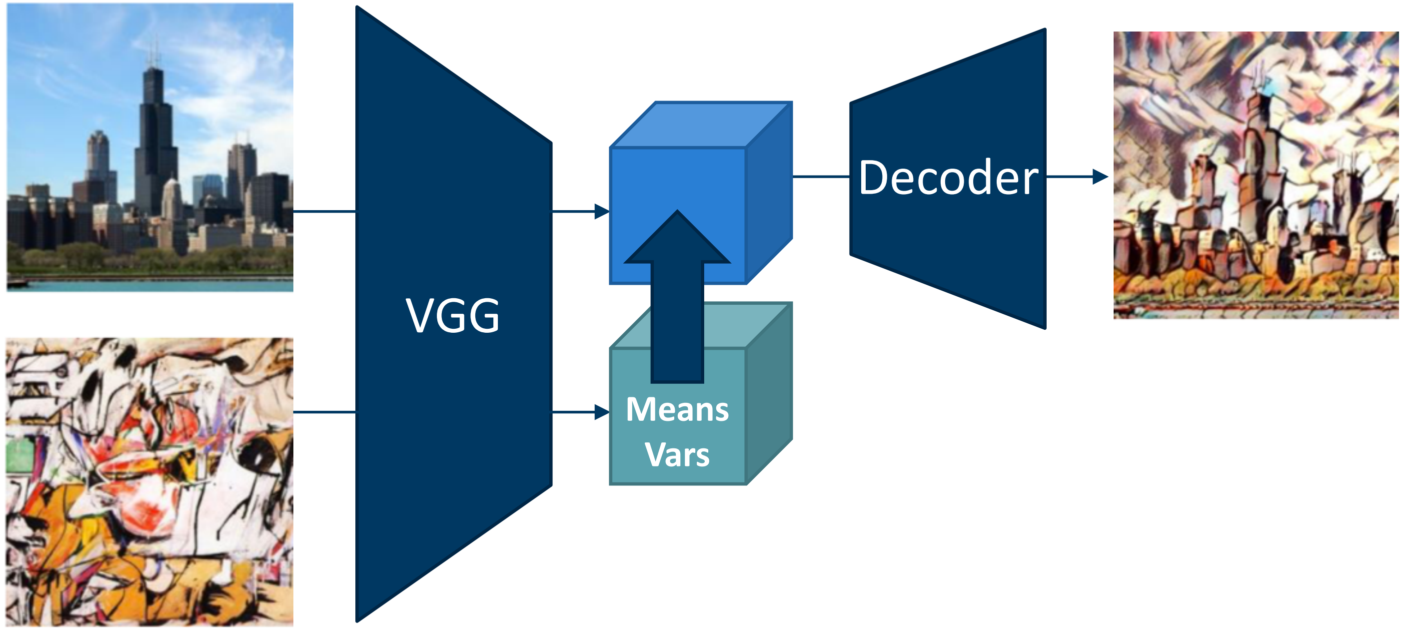
Arbitrary fast style transfer:
Adaptive instance normalization (AdaIN)



Huang & Belongie, *ICCV* 2017

Huang & Belongie, *ICCV* 2017

AdaIN

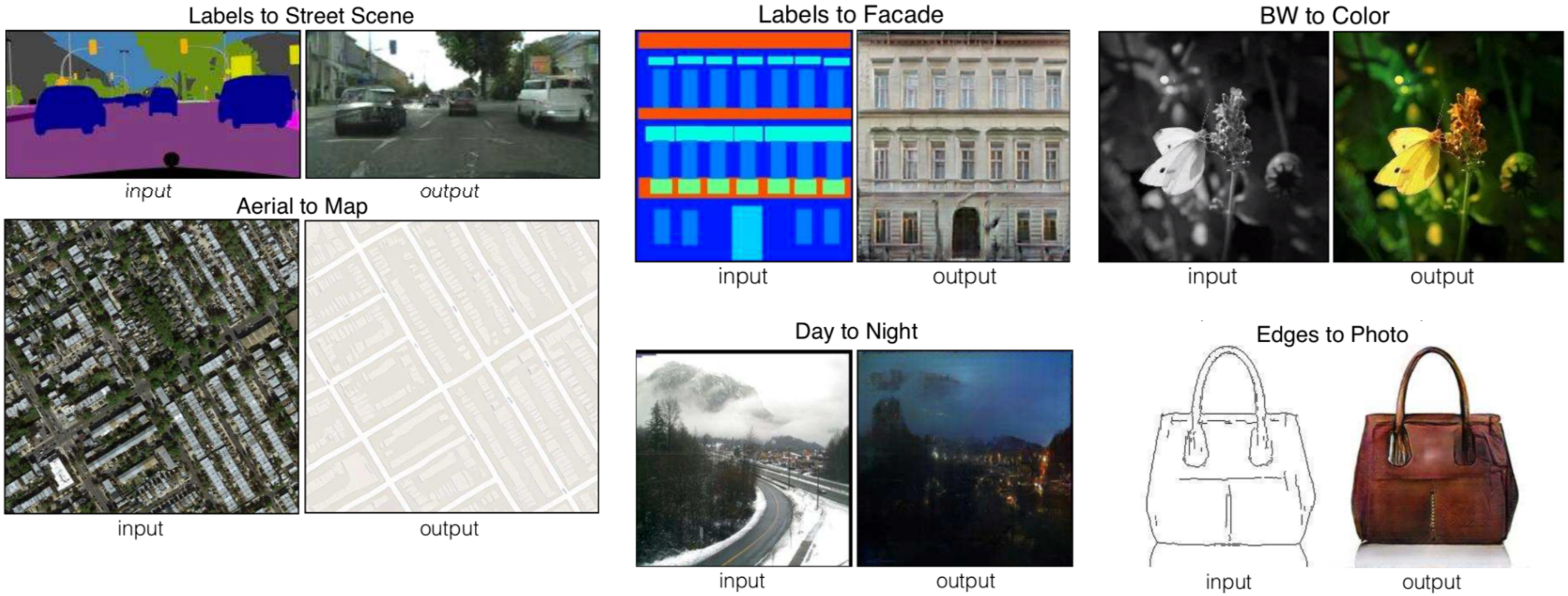


AdaIN results

Huang & Belongie, *ICCV* 2017

Unpaired domain translation

Domain translation



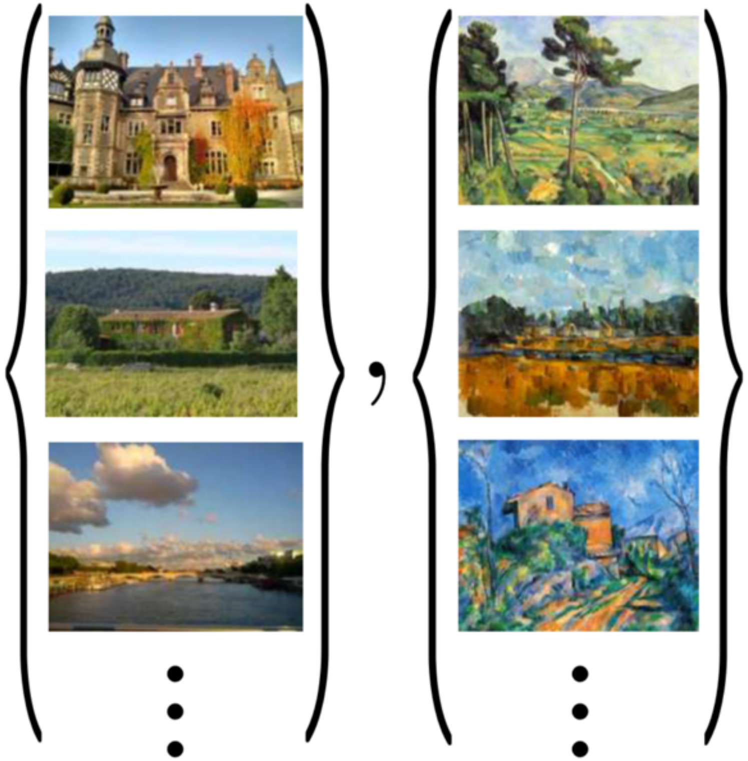
Unpaired domain translation

Paired training data

x_i y_i

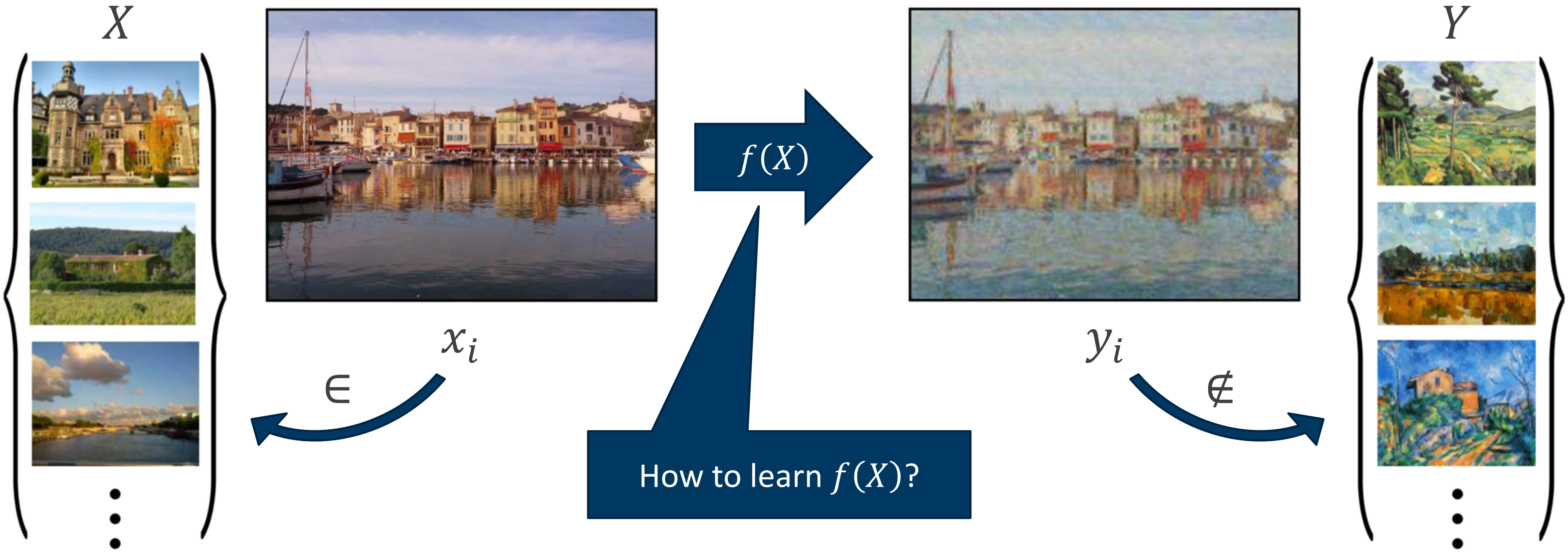
Unpaired training data

X Y



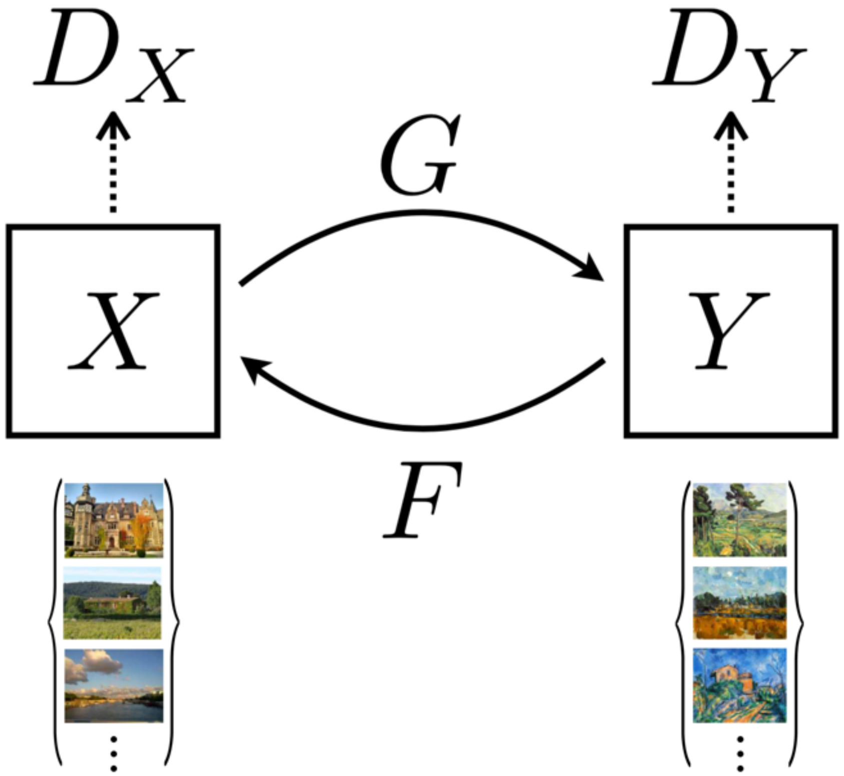
Zhu et al., "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks ", *ICCV* 2017

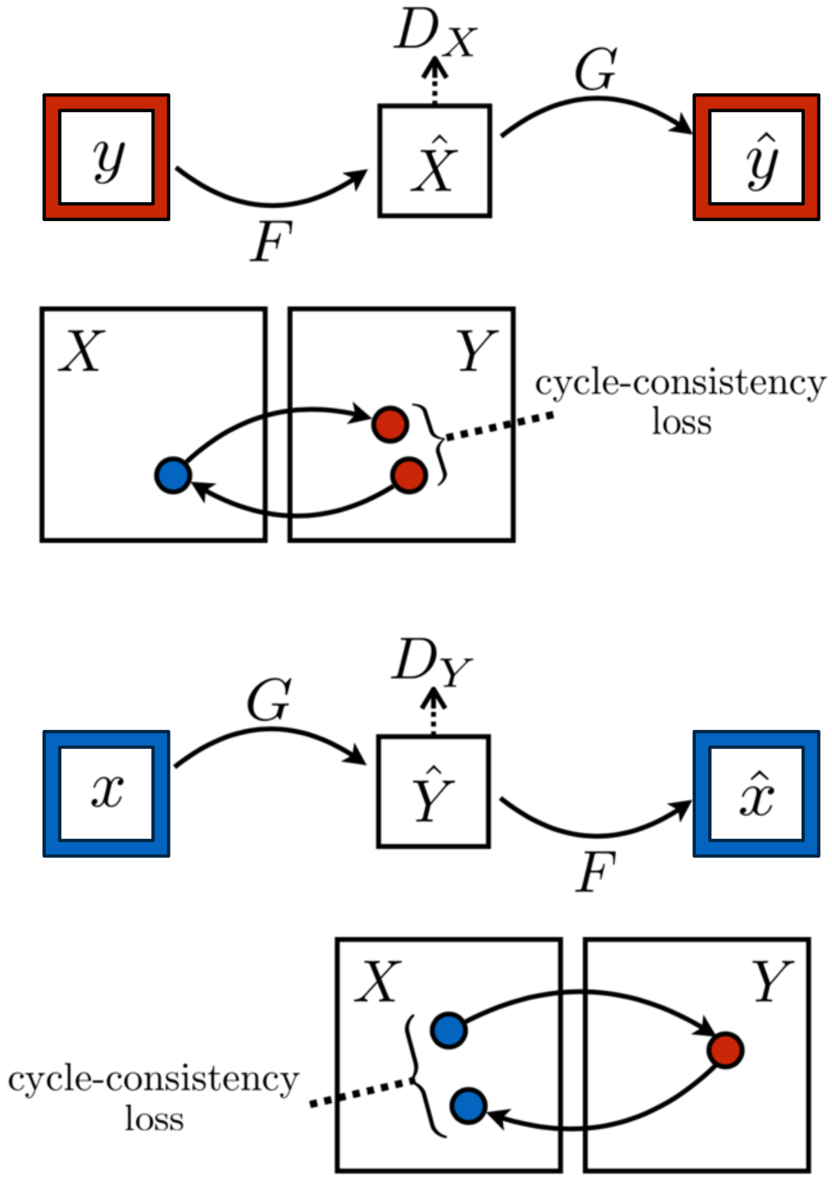
How to learn from unpaired data?



Zhu et al., "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks ", *ICCV* 2017

CycleGAN





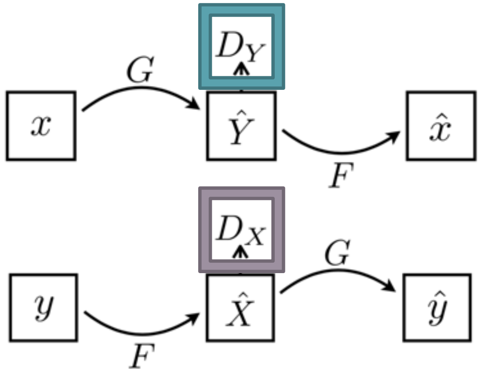
Zhu et al., "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks ", *ICCV* 2017

CycleGAN loss functions

For generator $G: X \rightarrow Y$ and discriminator D_Y:

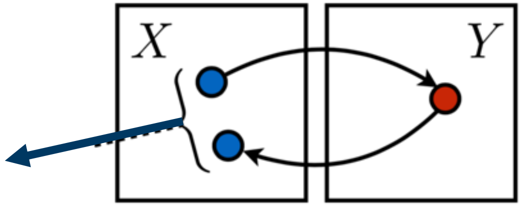
$$\mathcal{L}_{\mathrm{GAN}}(G, D_Y) = \mathbb{E}_{y\sim p(y)}[\log D_Y(y)] + \mathbb{E}_{x\sim p(x)}[\log(1 - D_Y(G(x)))]$$

(analogous for generator $F: Y \rightarrow X$ and discriminator D_X: $\mathcal{L}_{\mathrm{GAN}}(G, D_Y)$)



Cycle consistency loss:

$$\mathcal{L}_{\mathrm{cyc}}(G, F) = \mathbb{E}_{x\sim p(x)}\left[\|F(G(x)) - x\|_1\right] + \mathbb{E}_{y\sim p(y)}\left[\|G(F(y)) - y\|_1\right]$$



Full objective function:

$$\mathcal{L}(G, F, D_X, D_Y) = \mathcal{L}_{\mathrm{GAN}}(G, D_Y) + \mathcal{L}_{\mathrm{GAN}}(F, D_X) + \mathcal{L}_{\mathrm{cyc}}(G, F)$$

Zhu et al., "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", *ICCV* 2017

CycleGAN results

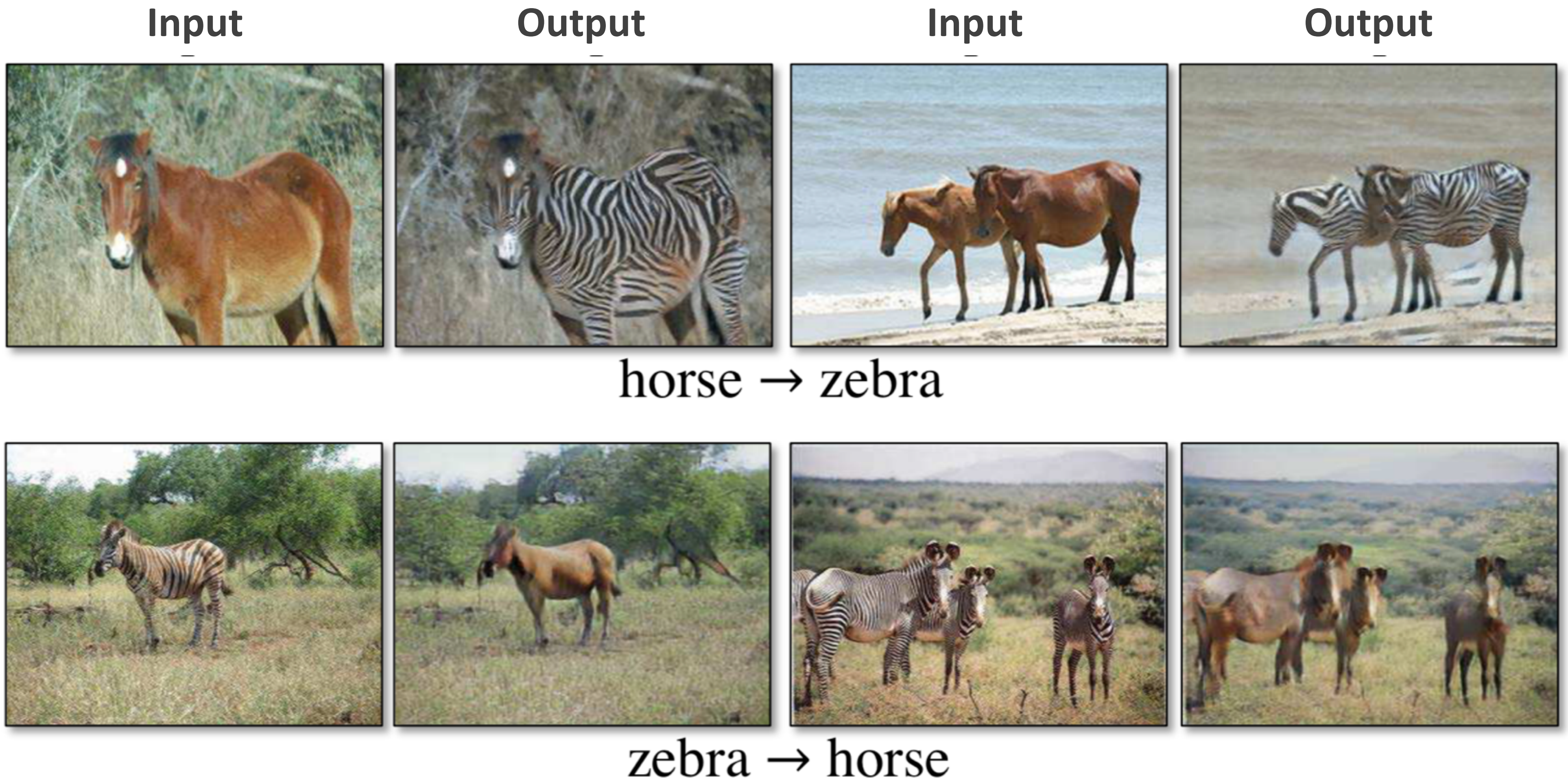
Input **Output** **Input** **Output**

winter Yosemite → summer Yosemite

summer Yosemite → winter Yosemite

Zhu et al., *ICCV* 2017

CycleGAN results



Zhu et al., *ICCV* 2017

CycleGAN results

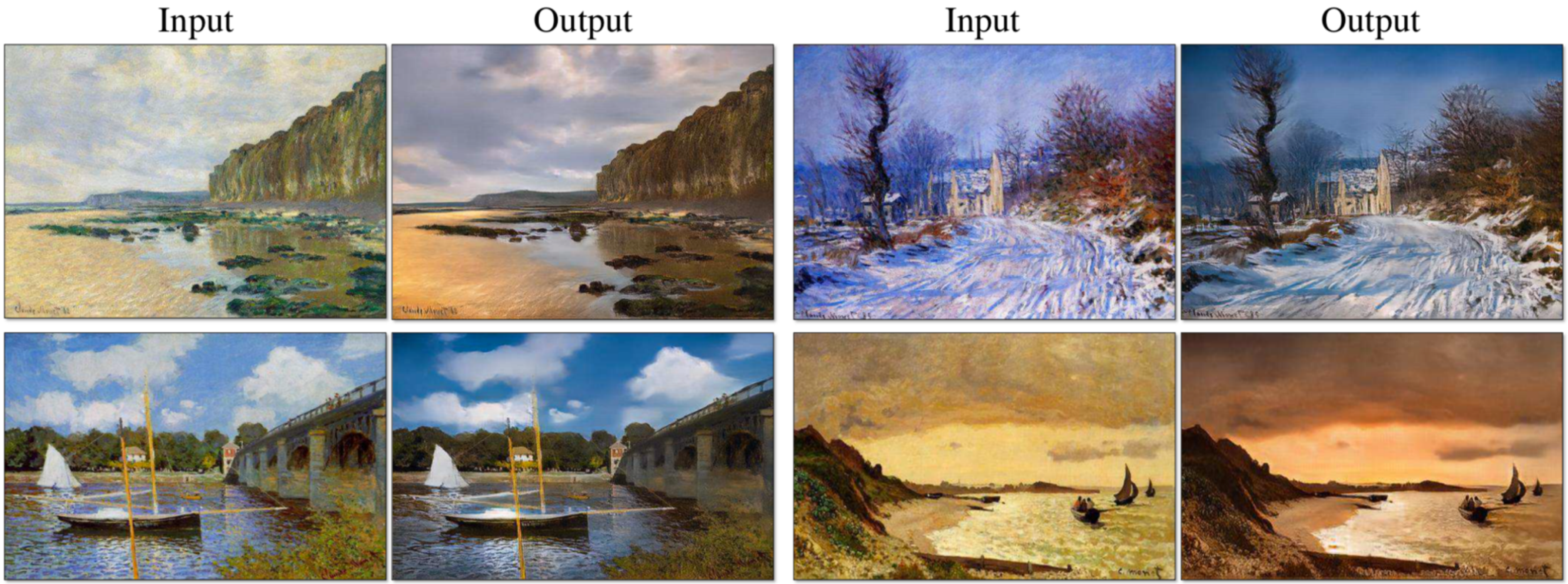
Input **Output** **Input** **Output**

apple → orange

orange → apple

Zhu et al., *ICCV* 2017

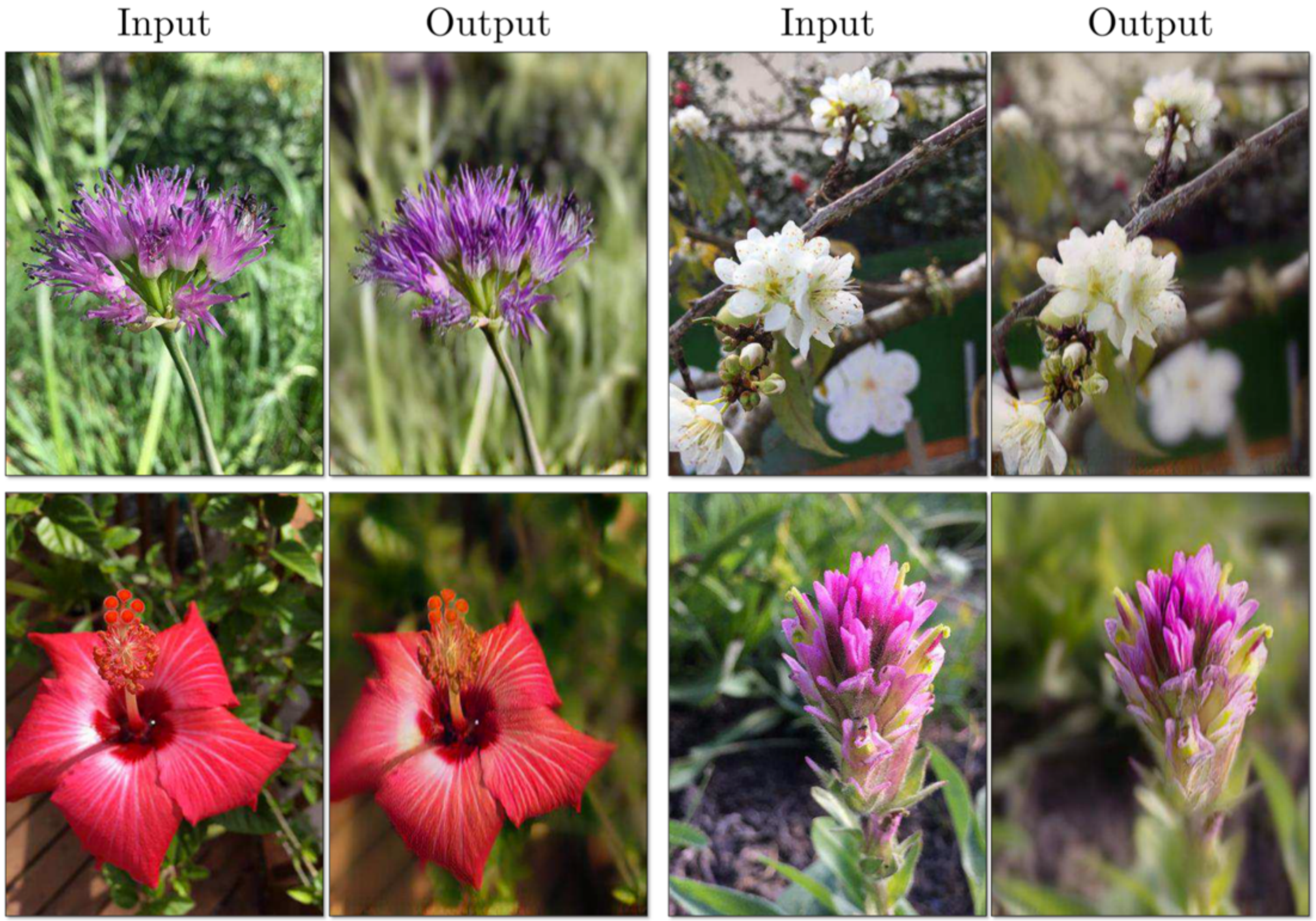
CycleGAN: Monet → Photograph



Zhu et al., *ICCV* 2017

CycleGAN: photo enhancement

iPhone
→ DSLR



Zhu et al., *ICCV* 2017

Questions!