



Computer Vision; Image Transformation; Super-Resolution, Denoising, Colorization, and Depth Estimation



[YouTube Playlist](#)

Maziar Raissi

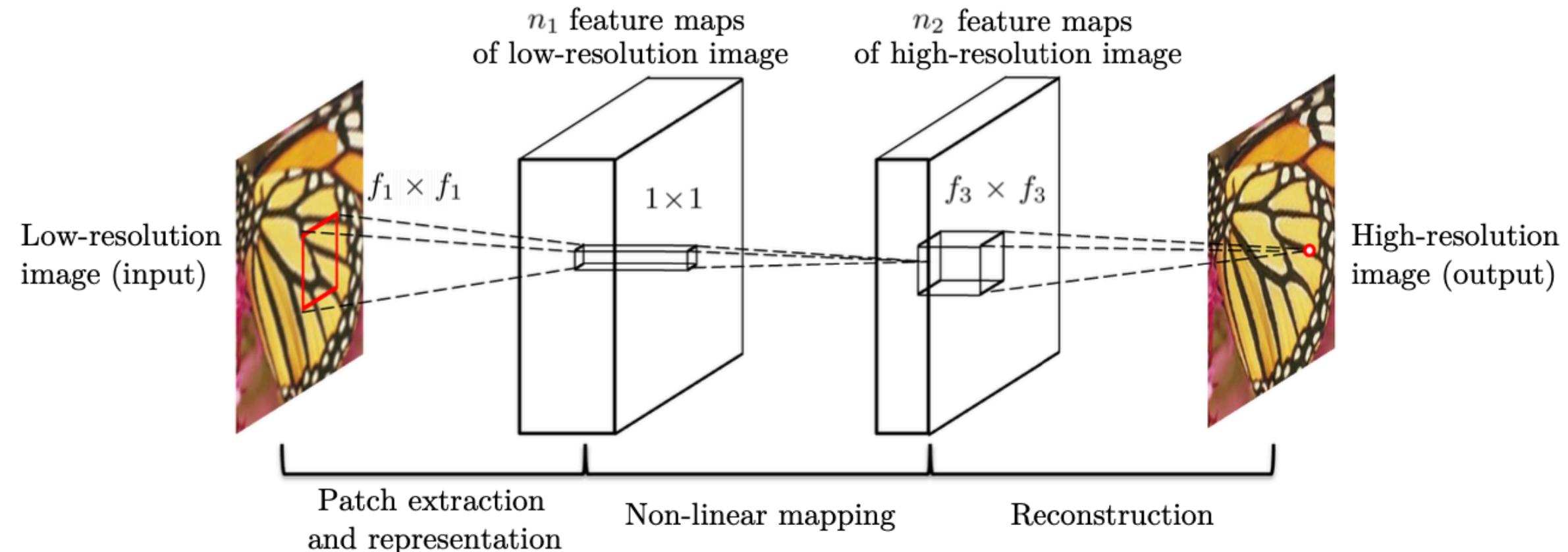
Assistant Professor

Department of Applied Mathematics

University of Colorado Boulder

maziar.raissi@colorado.edu

Learning a Deep Convolutional Network for Image Super-Resolution


[YouTube Video](#)


low resolution image \triangleright bicubic interpolation
 $Y \rightarrow$ interpolated image
 $F(Y) \rightarrow$ an image as similar as possible to X
 $X \rightarrow$ high resolution image

$$F_1(Y) = \max(0, W_1 * Y + B_1)$$

$$W_1 \in \mathbb{R}^{c \times f_1 \times f_1 \times n_1} \quad B_1 \in \mathbb{R}^{n_1}$$

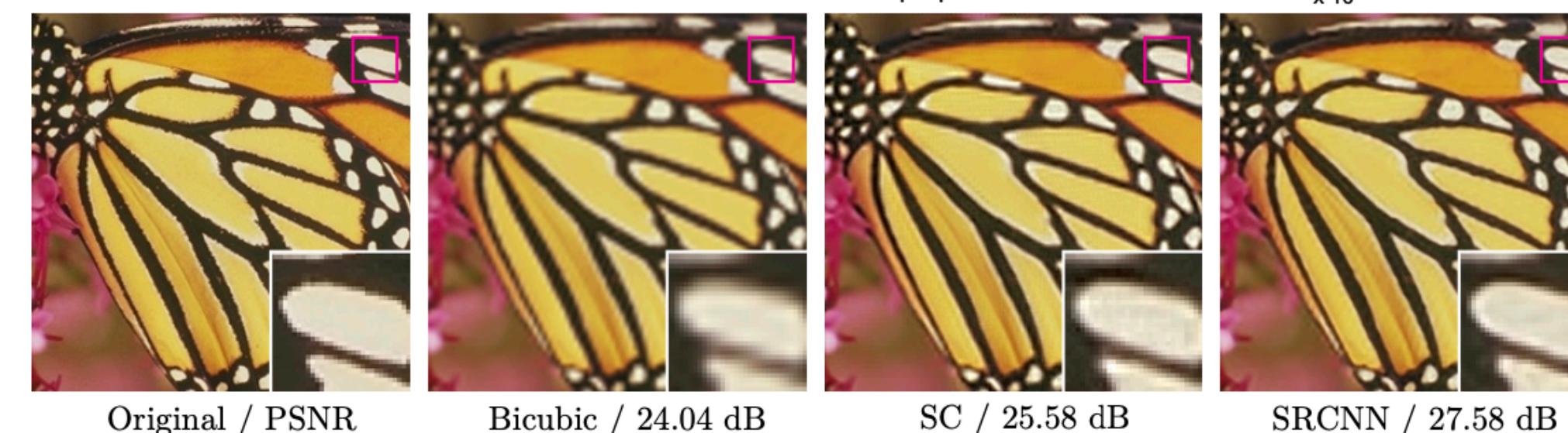
$$F_2(Y) = \max(0, W_2 * F_1(Y) + B_2)$$

$$W_2 \in \mathbb{R}^{n_1 \times 1 \times 1 \times n_2} \quad B_2 \in \mathbb{R}^{n_2}$$

$$F_3(Y) = W_3 * F_2(Y) + B_3$$

$$W_3 \in \mathbb{R}^{n_2 \times f_3 \times f_3 \times c} \quad B_3 \in \mathbb{R}^c$$

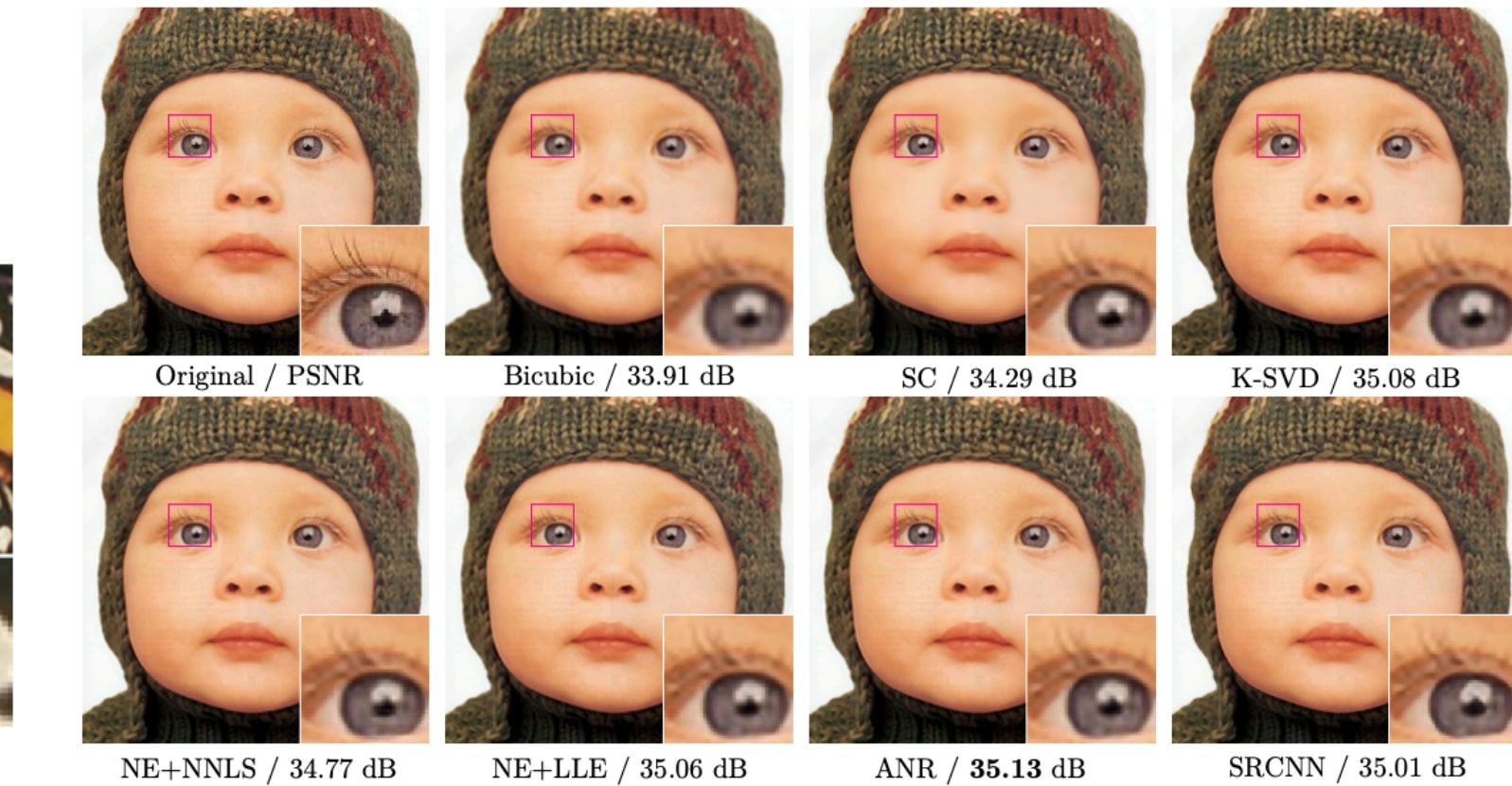
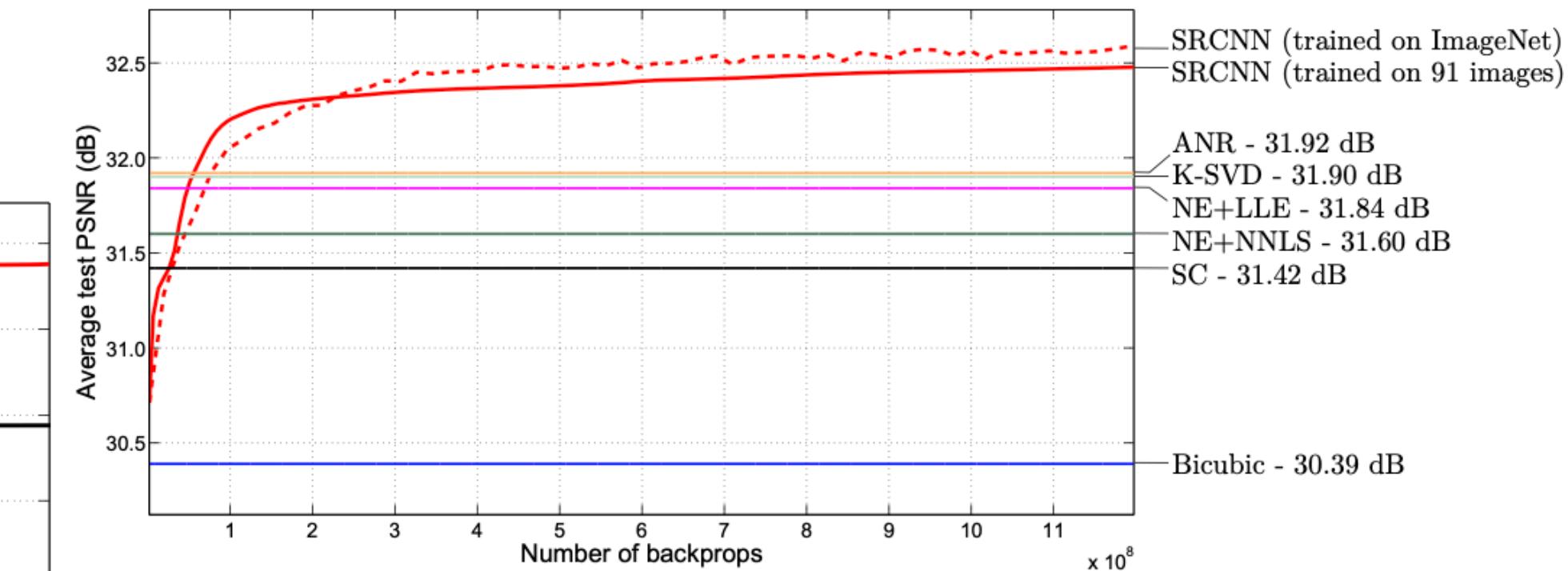
$$L(\Theta) = \frac{1}{n} \sum_{i=1}^n \|F(Y_i; \Theta) - X_i\|^2$$



Super-Resolution Convolutional Neural Network (SRCNN)
 Sparse Coding (SC) based method
 Peak Signal-to-Noise Ratio (PSNR)

Set5 [2] images	$n_1 = 128, n_2 = 64$		$n_1 = 64, n_2 = 32$		$n_1 = 32, n_2 = 16$	
	PSNR	Time	PSNR	Time	PSNR	Time
	32.60	0.60	32.52	0.18	32.26	0.05

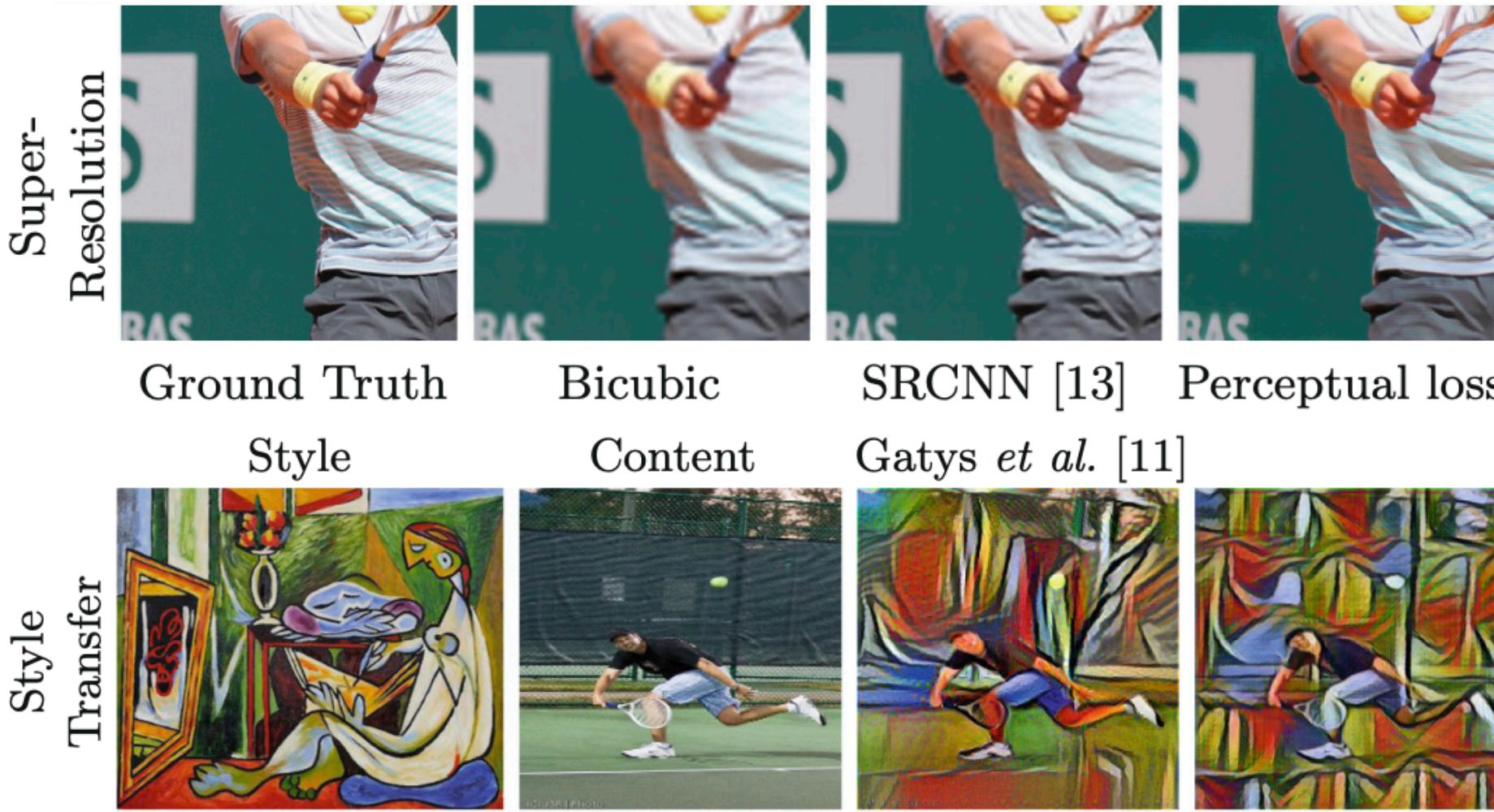
$$\begin{aligned} PSNR &= 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \\ &= 20 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \\ MAX_I &= 255 \end{aligned}$$



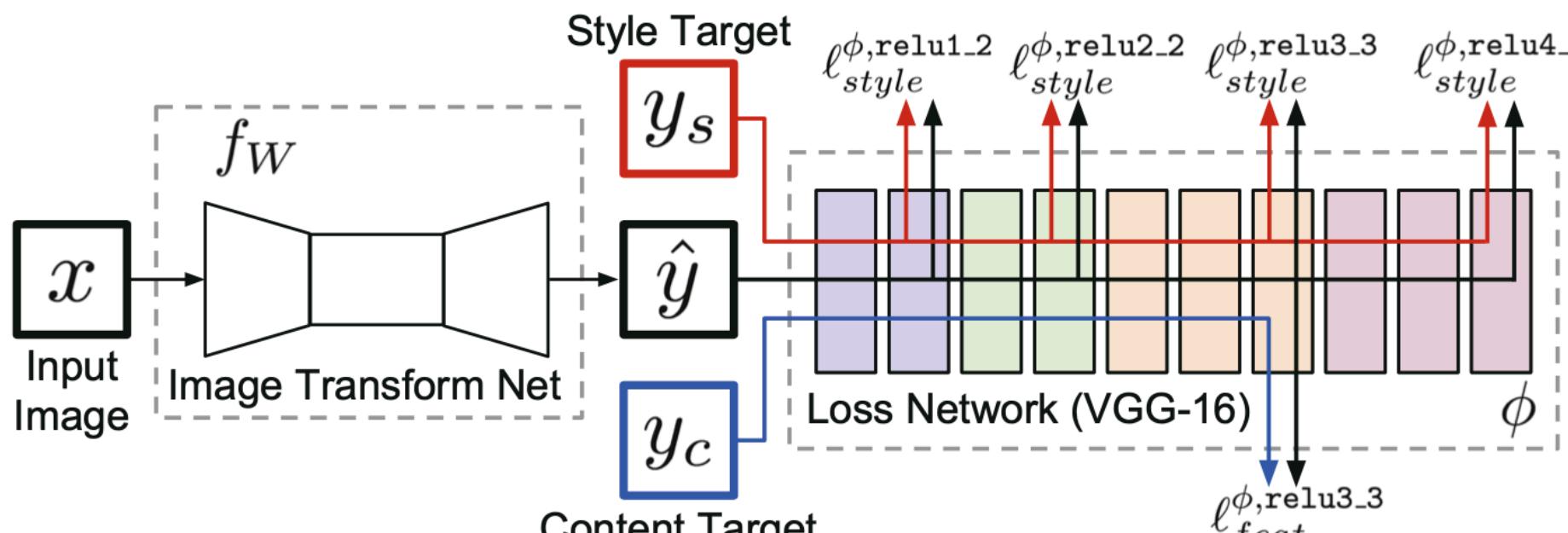
Dong, Chao, et al. "Learning a deep convolutional network for image super-resolution." *European conference on computer vision*. Springer, Cham, 2014.

Dong, Chao, et al. "Image super-resolution using deep convolutional networks." *IEEE transactions on pattern analysis and machine intelligence* 38.2 (2015): 295-307.

Perceptual Losses for Real-Time Style Transfer and Super-Resolution


[YouTube Video](#)


"For example, consider two identical images offset from each other by one pixel; despite their perceptual similarity they would be very different as measured by per-pixel losses."



$$W^* = \arg \min_W \mathbb{E}_{x,y} \left[\sum_i \lambda_i \ell_i(\underbrace{f_W(x)}_{\hat{y}}, y) \right]$$

Style transfer: $y_c = x$ & train one network per style target y_s

Super-resolution: $y_c = \text{high-resolution image}$ & style reconstruction loss is not used

Feature Reconstruction Loss

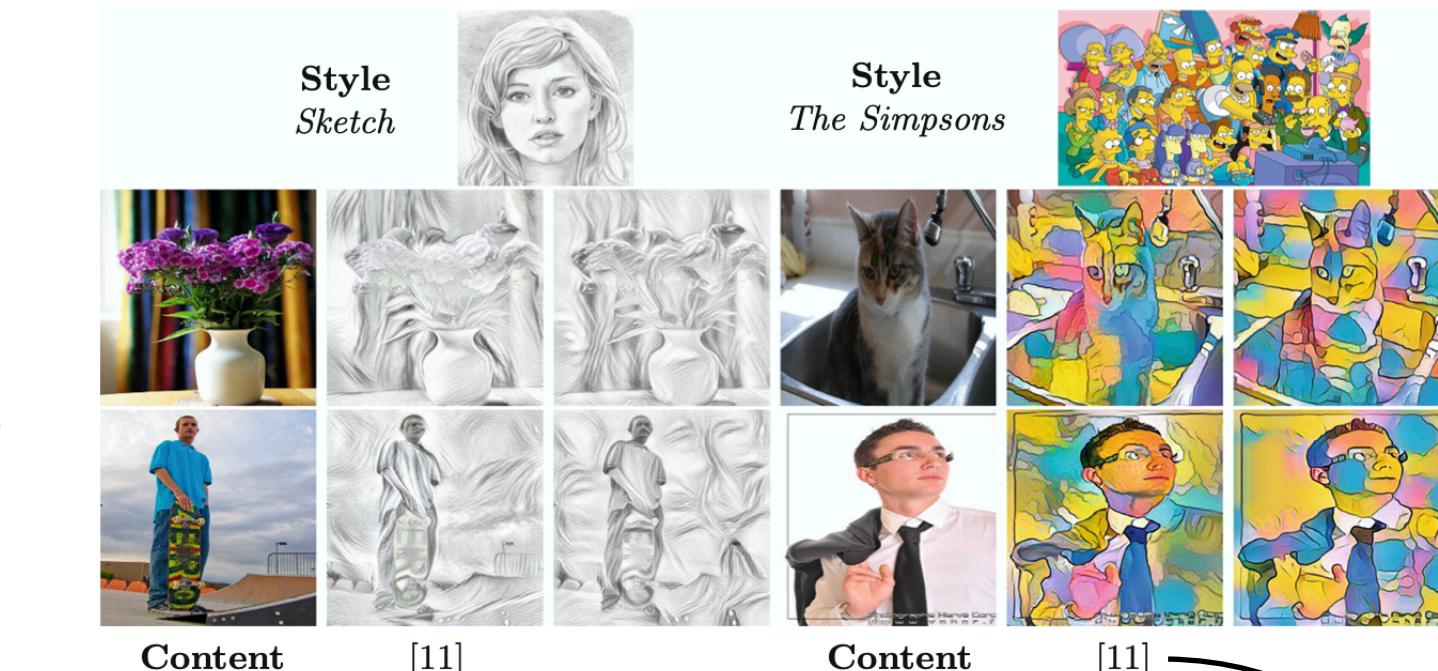
$\phi_j(x) \in \mathbb{R}^{C_j \times H_j \times W_j} \rightarrow \text{activations of the } j\text{-th layer of network } \phi$

$$\ell_{feat}^{\phi,j}(\hat{y}, y) = \frac{1}{C_j H_j W_j} \|\phi_j(\hat{y}) - \phi_j(y)\|_2^2$$

Style Reconstruction Loss

$$G_j^\phi(x)_{c,c'} = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'}$$

$G_j^\phi(x) \in \mathbb{R}^{C_j \times C_j} \rightarrow \text{Gram Matrix}$



Gram matrix captures information about which features tend to activate together.

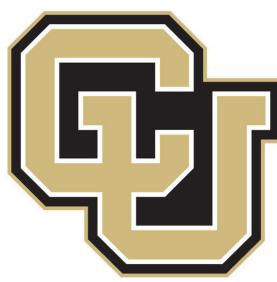
$$\ell_{style}^{\phi,j}(\hat{y}, y) = \|G_j^\phi(\hat{y}) - G_j^\phi(y)\|_F^2$$

$$\hat{y} = \arg \min_y \lambda_c \ell_{feat}^{\phi,j}(y, y_c) + \lambda_s \ell_{style}^{\phi,j}(y, y_s) + \lambda_{TV} \ell_{TV}(y)$$



Speedup over [11] at 100, 300, 500 optimization iterations

Image Size	Speedup		
	100	300	500
256 × 256	212x	636x	1060x
512 × 512	205x	615x	1026x
1024 × 1024	208x	625x	1042x



Boulder

Image Style Transfer Using Convolutional Neural Networks

Content Representation

$p \rightarrow$ photograph

$x \rightarrow$ generated image (randomly initialized)

$P^l \rightarrow$ feature representation of p at layer l

$F^l \rightarrow$ feature representation of x at layer l

$$\mathcal{L}_{\text{content}}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} (F_{ij}^l - P_{ij}^l)^2$$

$F_{i,j}^l \rightarrow$ activation of i -th filter at position j in layer l

$$F^l \in \mathbb{R}^{N_l \times M_l}$$

$N_l \rightarrow$ number of feature maps

$M_l \rightarrow$ height times width of the feature map

Style Representation

$$G^l \in \mathbb{R}^{N_l \times N_l} \rightarrow \text{Gram matrix}$$

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l \rightarrow \text{inner product between the vectorized feature maps } i \text{ and } j \text{ in layer } l$$

captures texture but not global arrangement

$a \rightarrow$ artwork image $A^l \rightarrow$ style representation of a

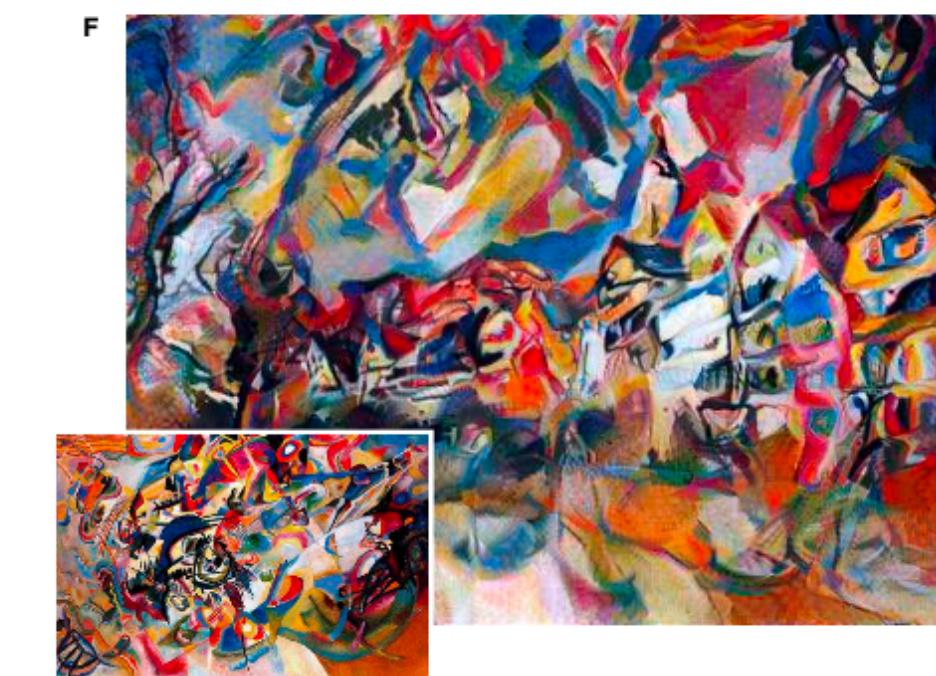
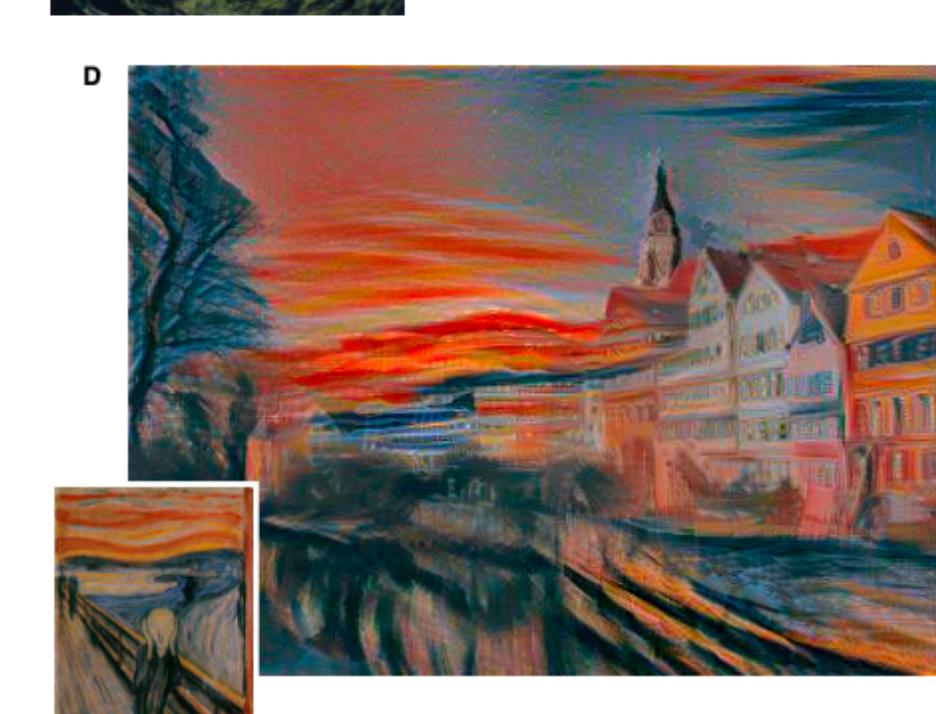
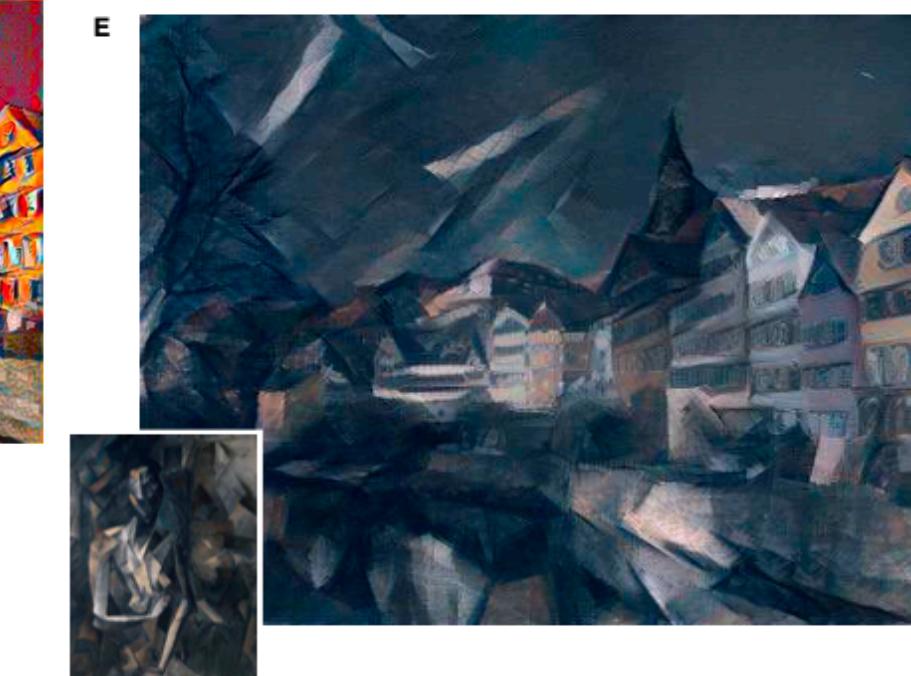
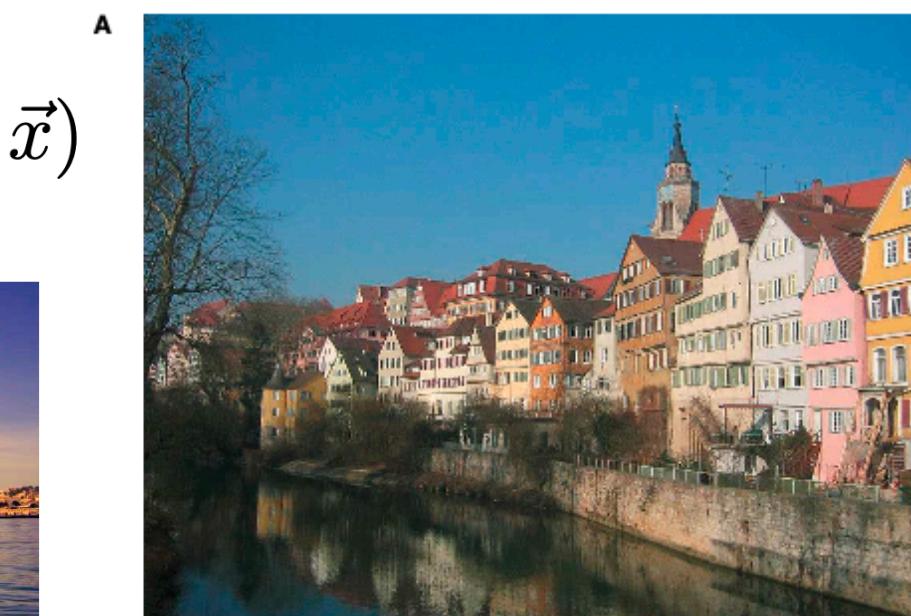
$G^l \rightarrow$ style representation of x

$$\mathcal{L}_{\text{style}}(\vec{a}, \vec{x}) = \sum_{l=0}^L w_l E_l \quad E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - A_{ij}^l)^2 \quad \boxed{\text{The ratio } \alpha/\beta}$$

Style Transfer

$$\mathcal{L}_{\text{total}}(\vec{p}, \vec{a}, \vec{x}) = \alpha \mathcal{L}_{\text{content}}(\vec{p}, \vec{x}) + \beta \mathcal{L}_{\text{style}}(\vec{a}, \vec{x})$$

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial \vec{x}}$$



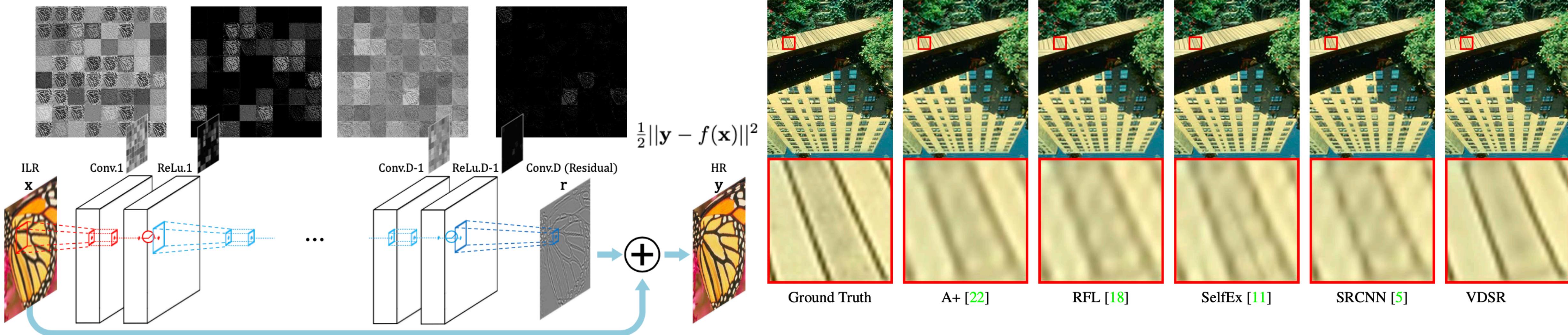
Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "Image style transfer using convolutional neural networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "A neural algorithm of artistic style." *arXiv preprint arXiv:1508.06576* (2015).

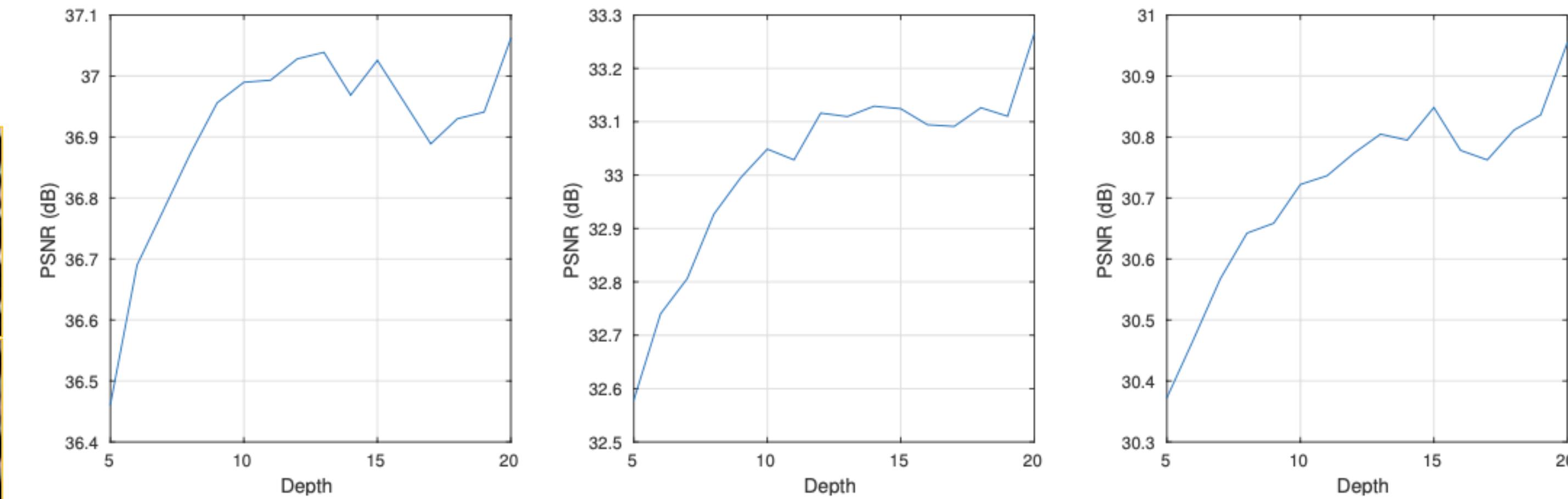
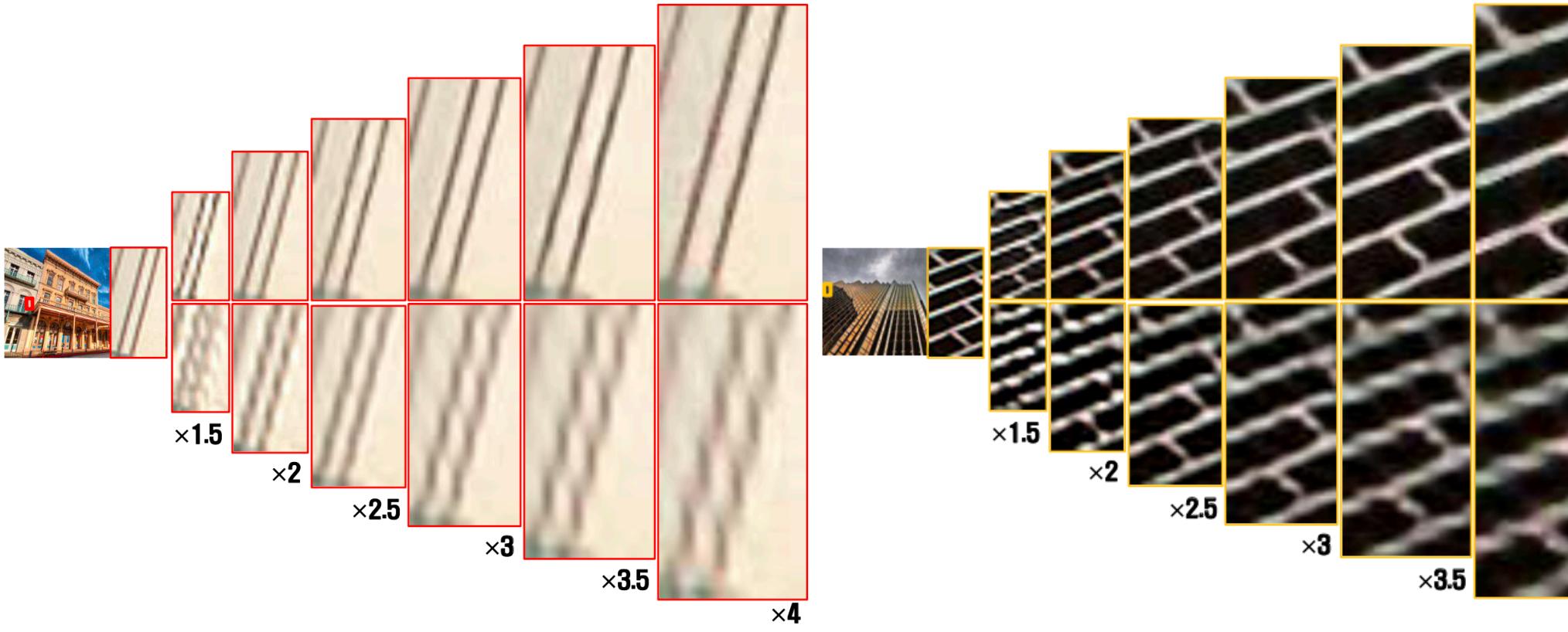
Accurate Image Super-Resolution Using Very Deep Convolutional Networks


[YouTube Video](#)

Single Image Super-Resolution (SISR)



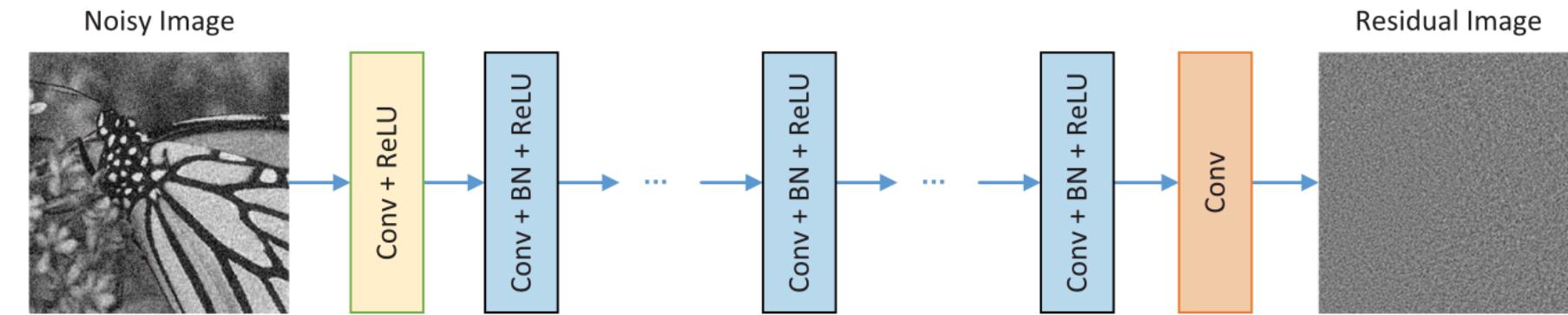
- Residual-Learning
- High Learning Rates
- Adjustable Gradient Clipping
- Multi-scale



Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising


[YouTube Video](#)

Denoising Convolutional Neural Networks (DnCNNs)



$x \rightarrow$ clean image

$y \rightarrow$ noisy observation

$y = x + v \rightarrow$ image degradation model

AWGN (Additive White Gaussian Noise)
with standard deviation σ

Receptive Field of DnCNN (3×3 conv)

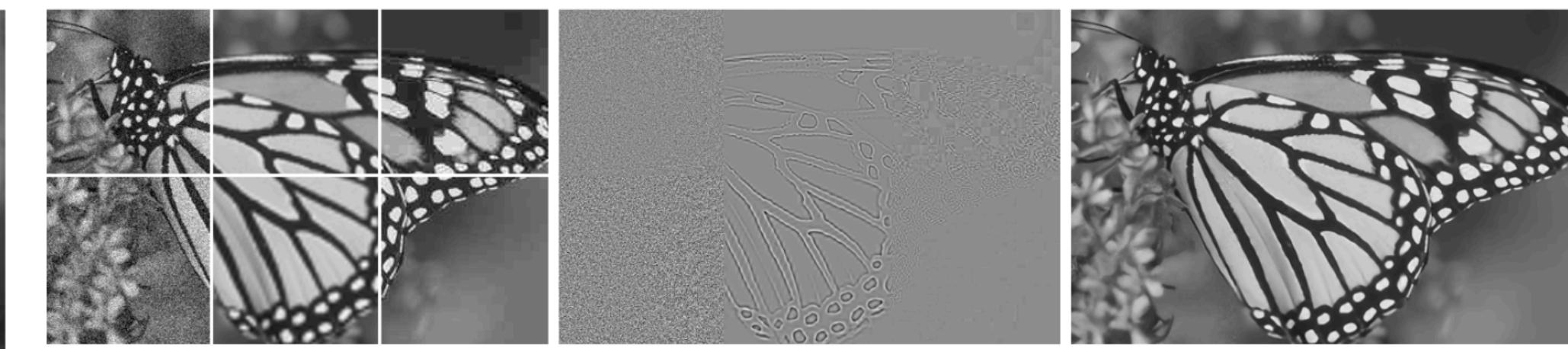
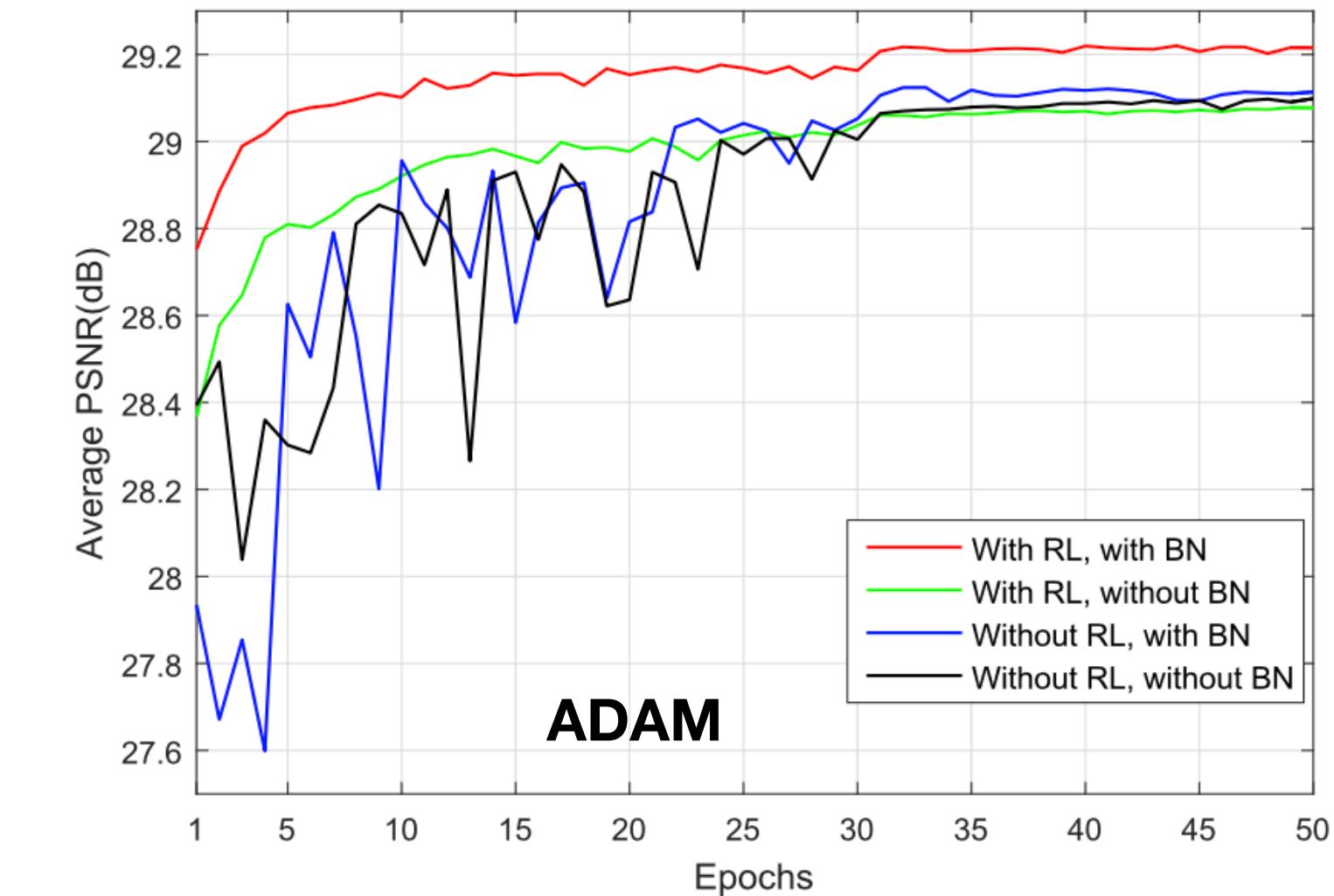
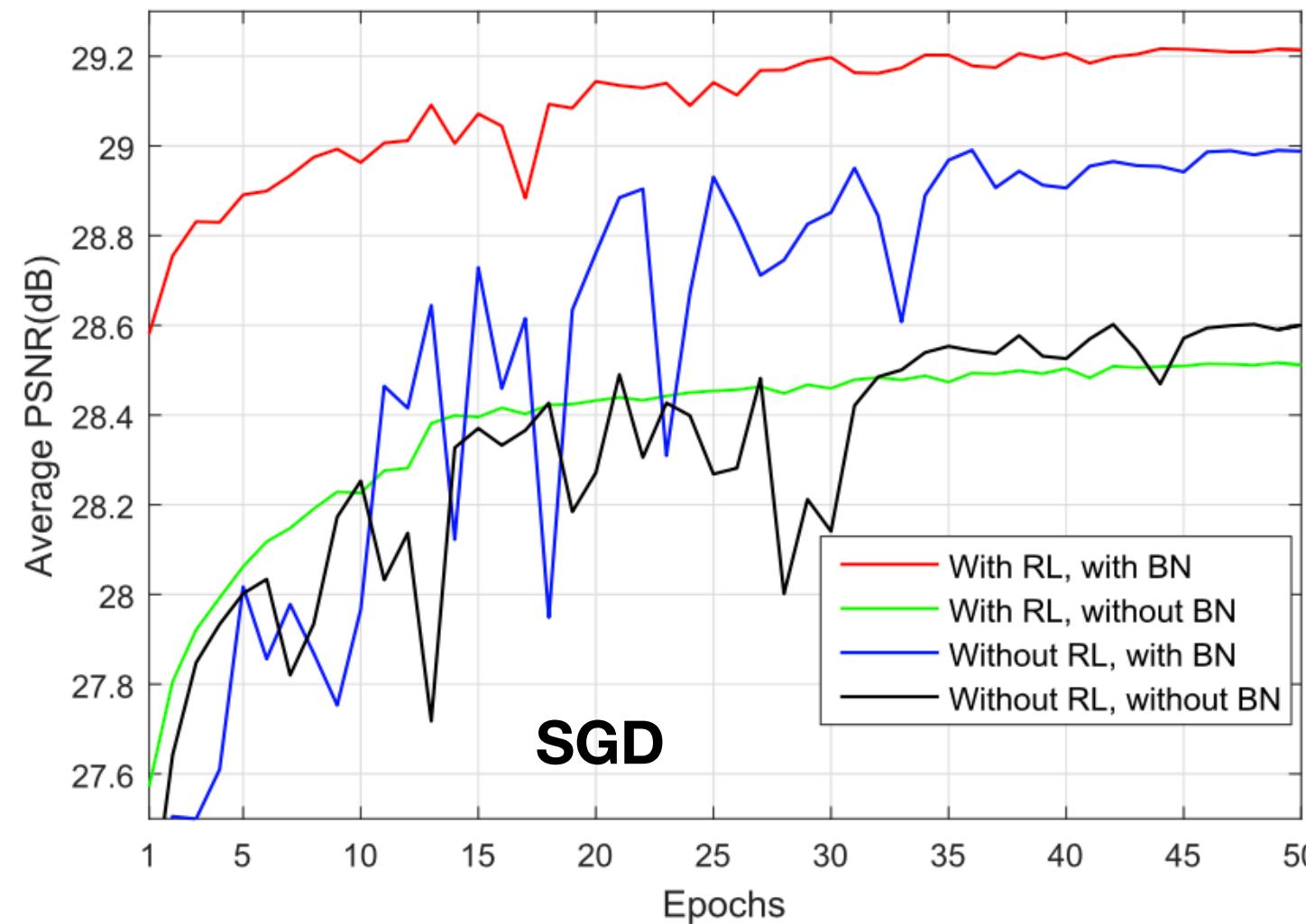
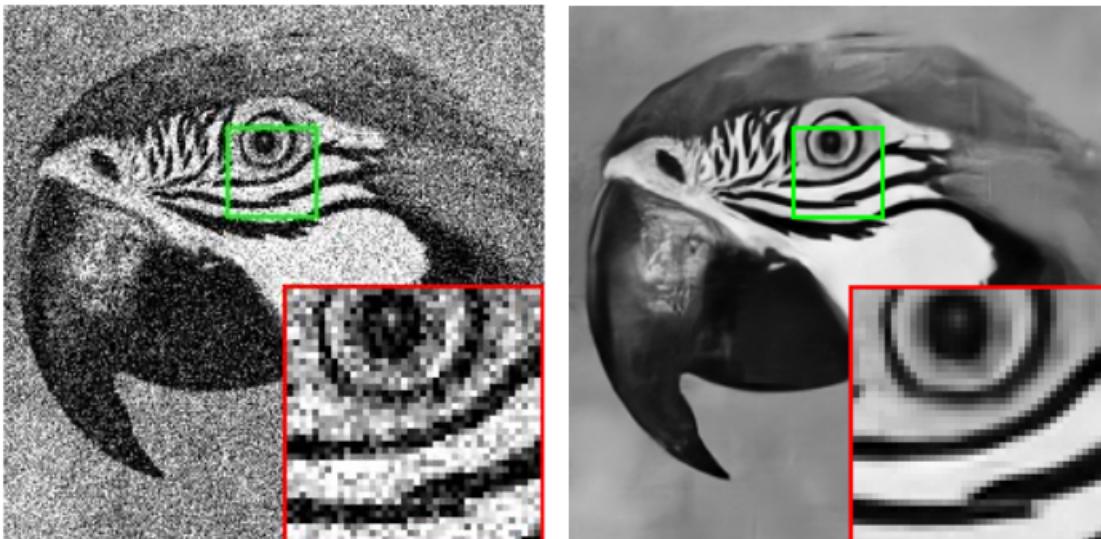
with depth d is $(2d + 1) \times (2d + 1)$

$\mathcal{F}(y) = x$

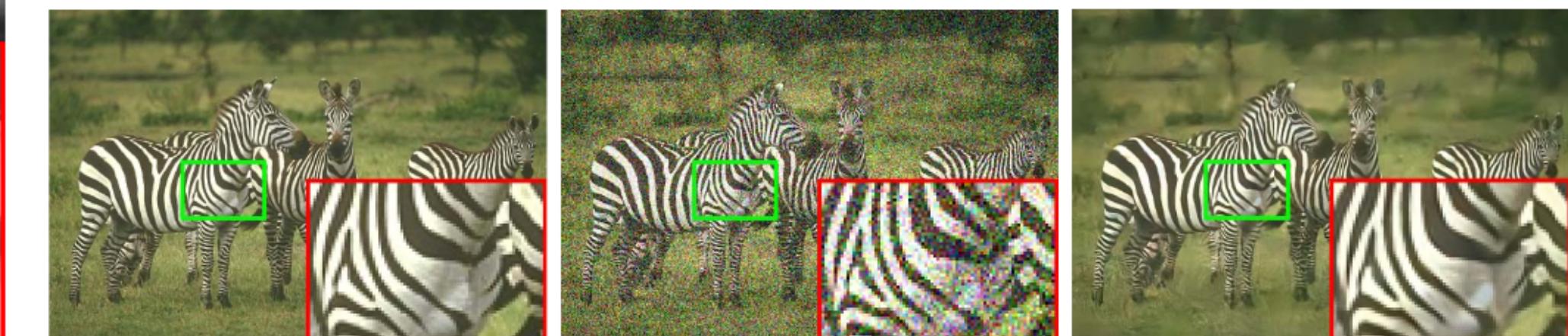
$\mathcal{R}(y) \approx v \rightarrow$ residual mapping

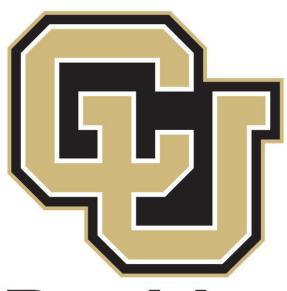
$\Rightarrow x = y - \mathcal{R}(y)$

$$\ell(\theta) = \frac{1}{2N} \sum_{i=1}^N \|R(y_i; \theta) - (y_i - x_i)\|_F^2$$



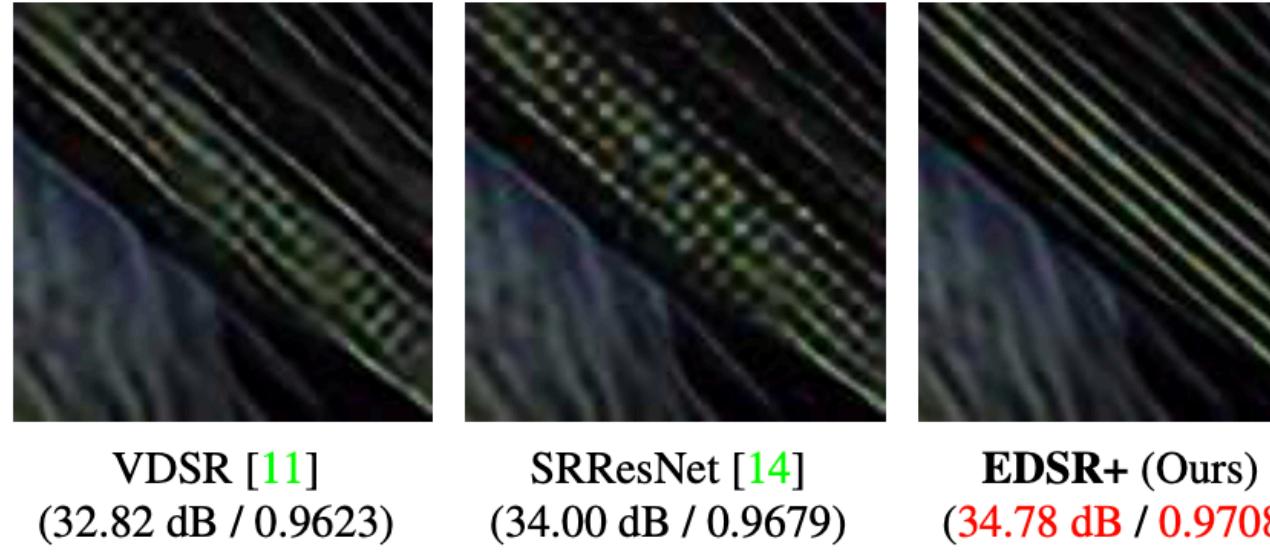
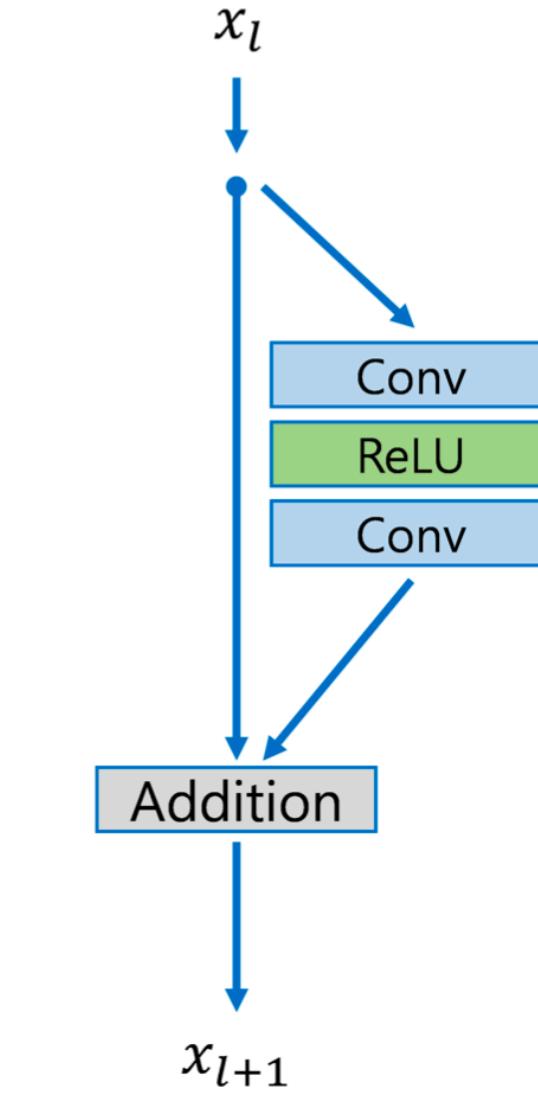
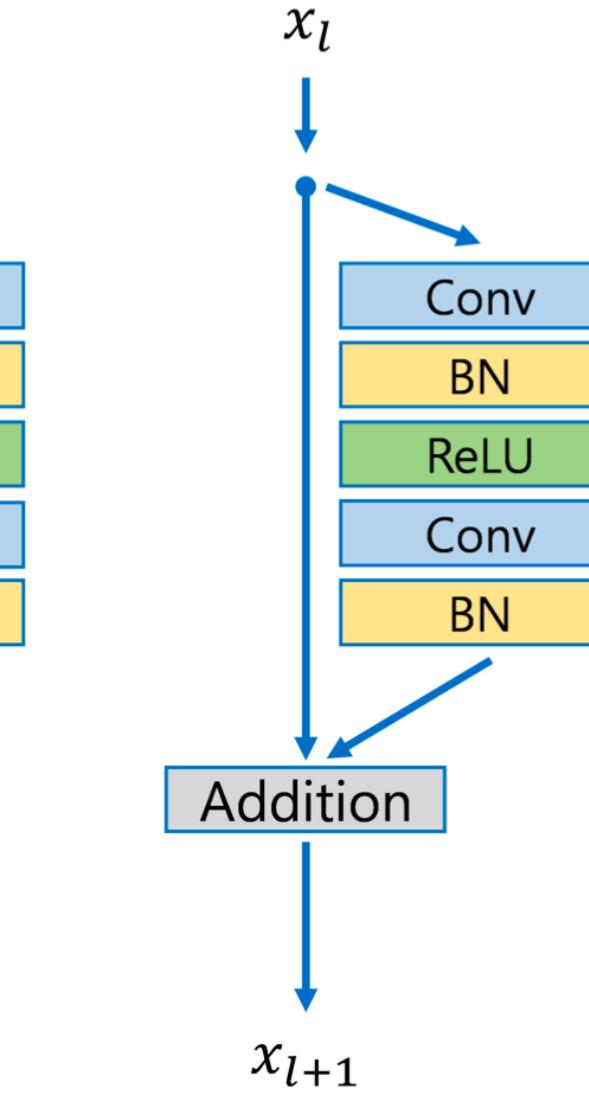
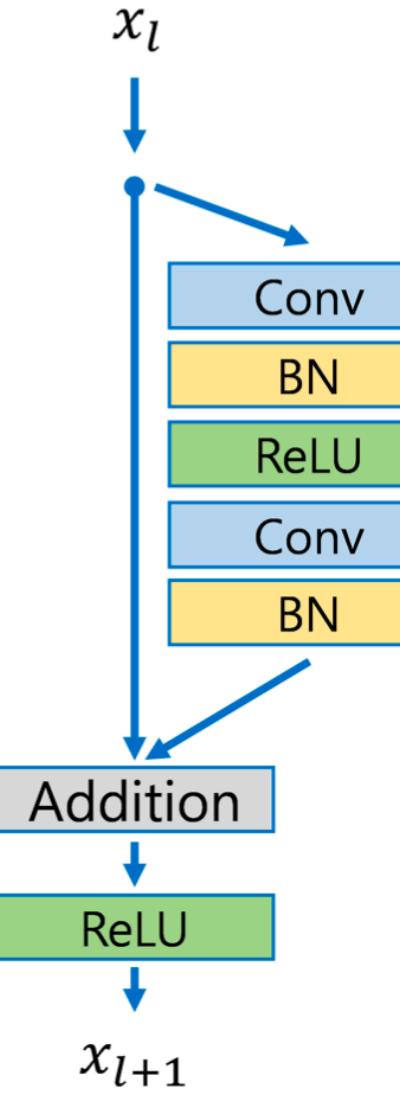
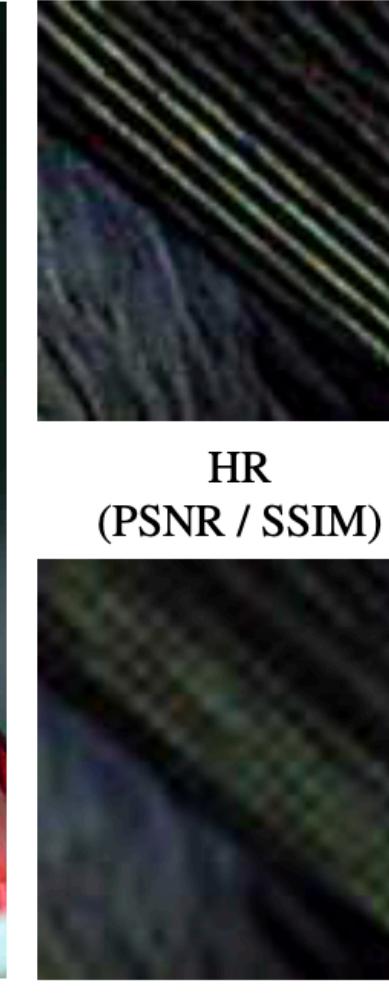
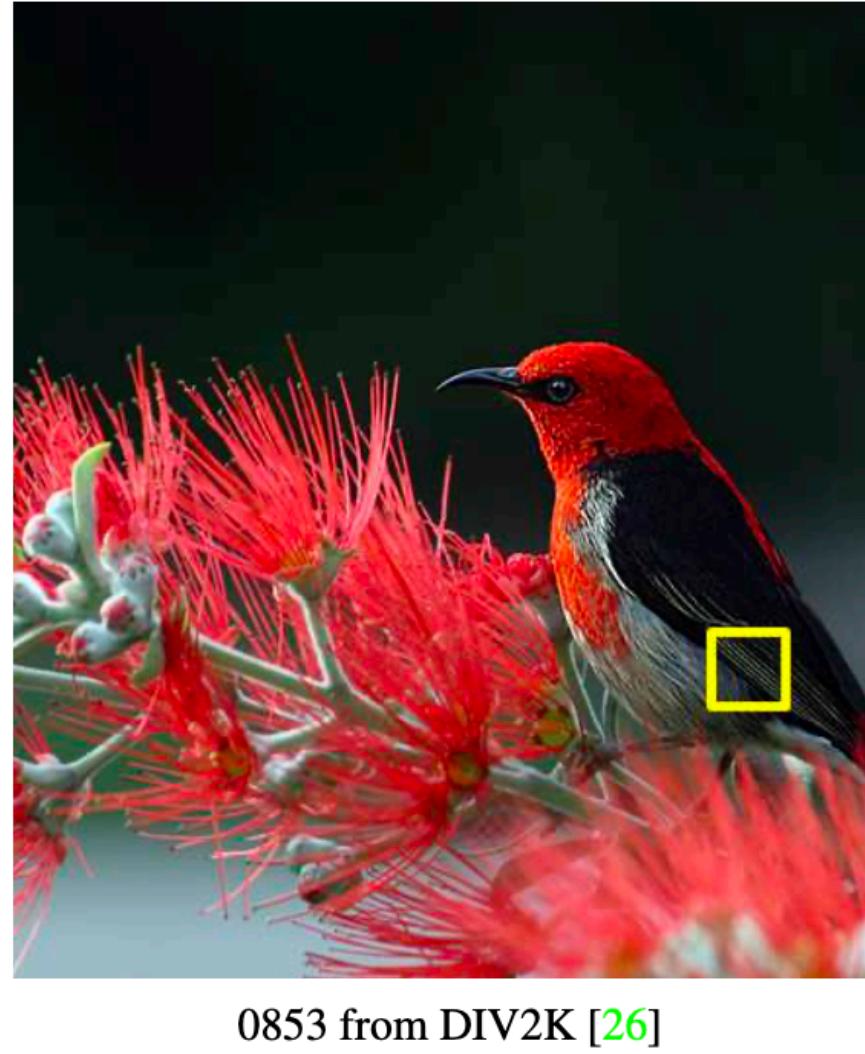
JPEG image de-blocking
Single Image Super-resolution
Gaussian Denoising





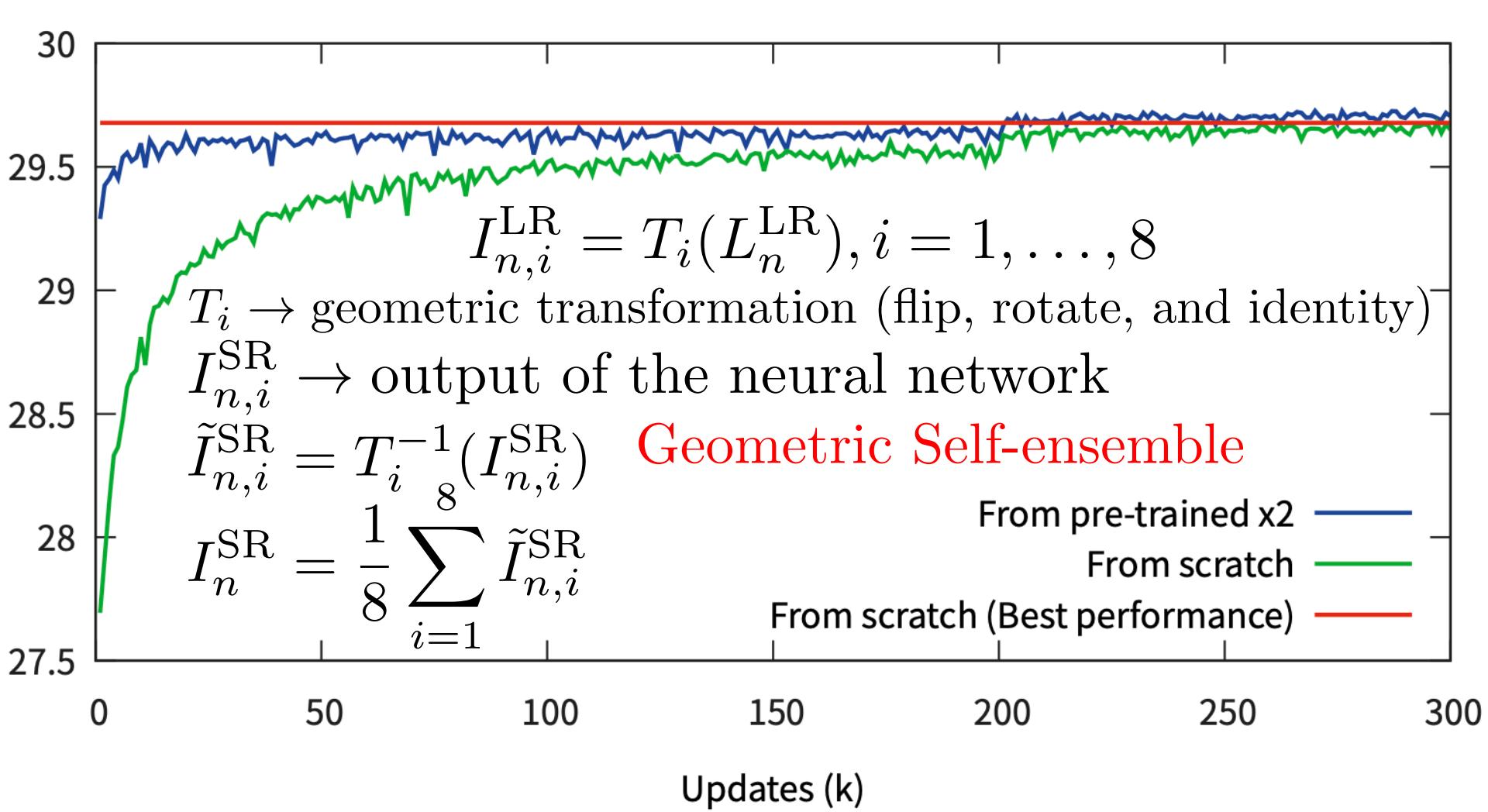
Boulder

Enhanced Deep Residual Networks for Single Image Super-Resolution

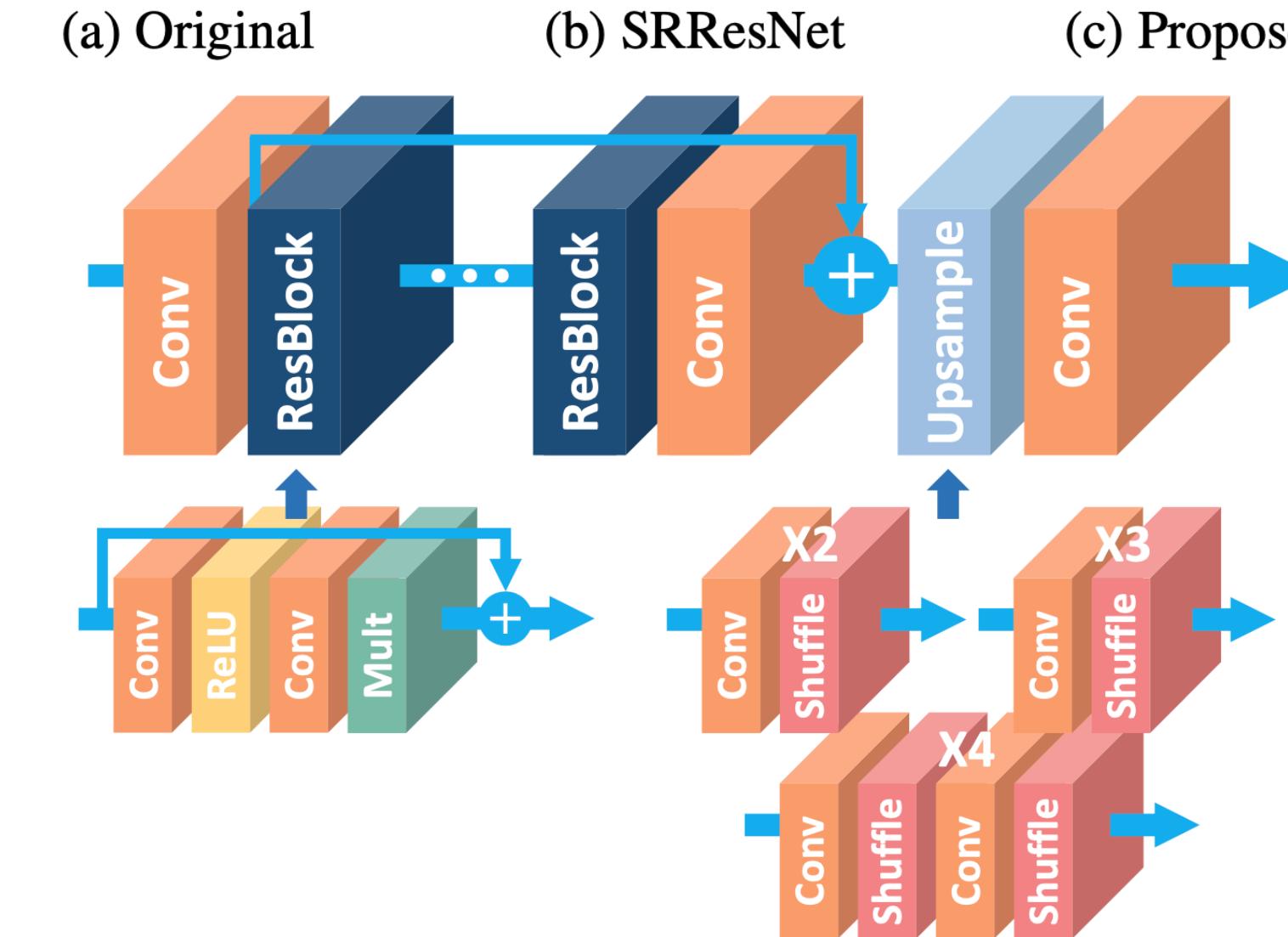


Options	SRResNet [14] (reproduced)	Baseline (Single / Multi)	EDSR	MDSR
# Residual blocks	16	16	32	80
# Filters	64	64	256	64
# Parameters	1.5M	1.5M / 3.2M	43M	8.0M
Residual scaling	-	-	0.1	-
Use BN	Yes	No	No	No
Loss function	L2	L1	L1	L1

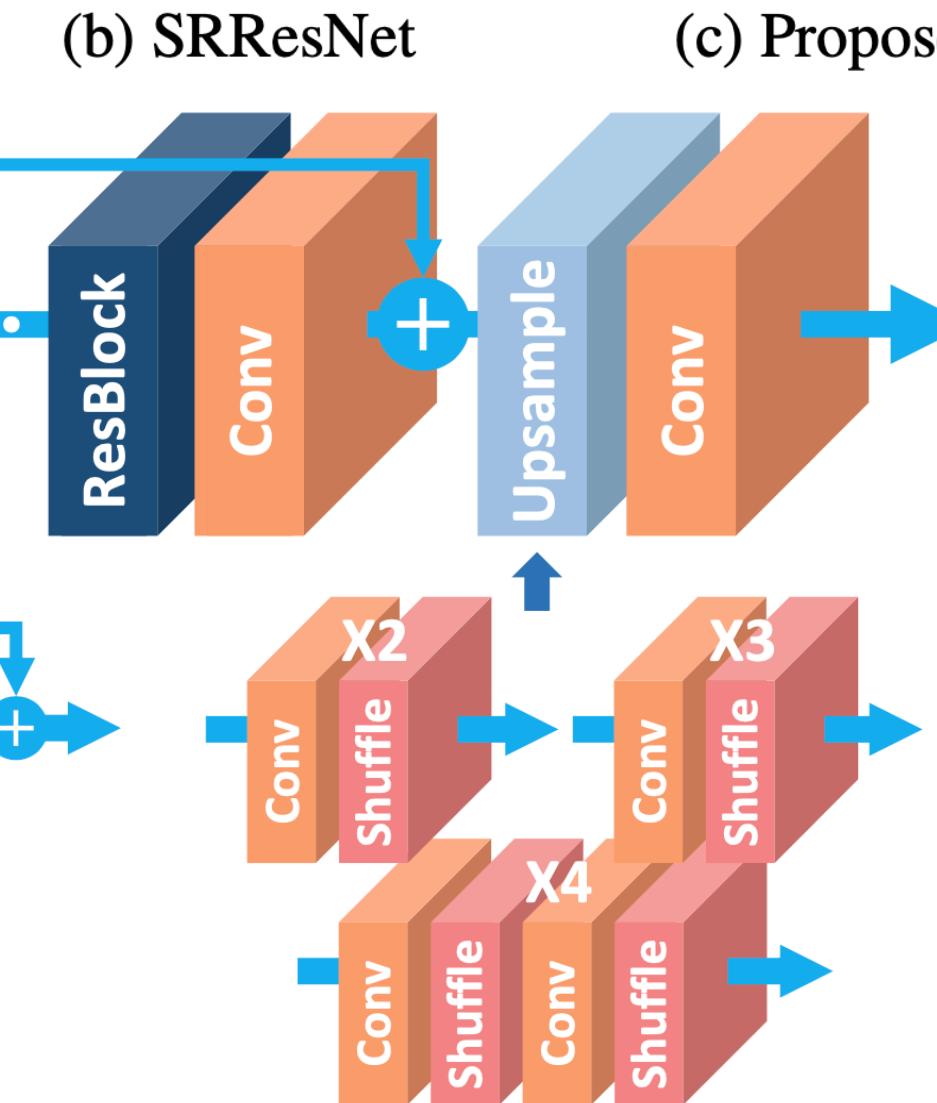
PSNR(dB) on DIV2K validation set (x4)



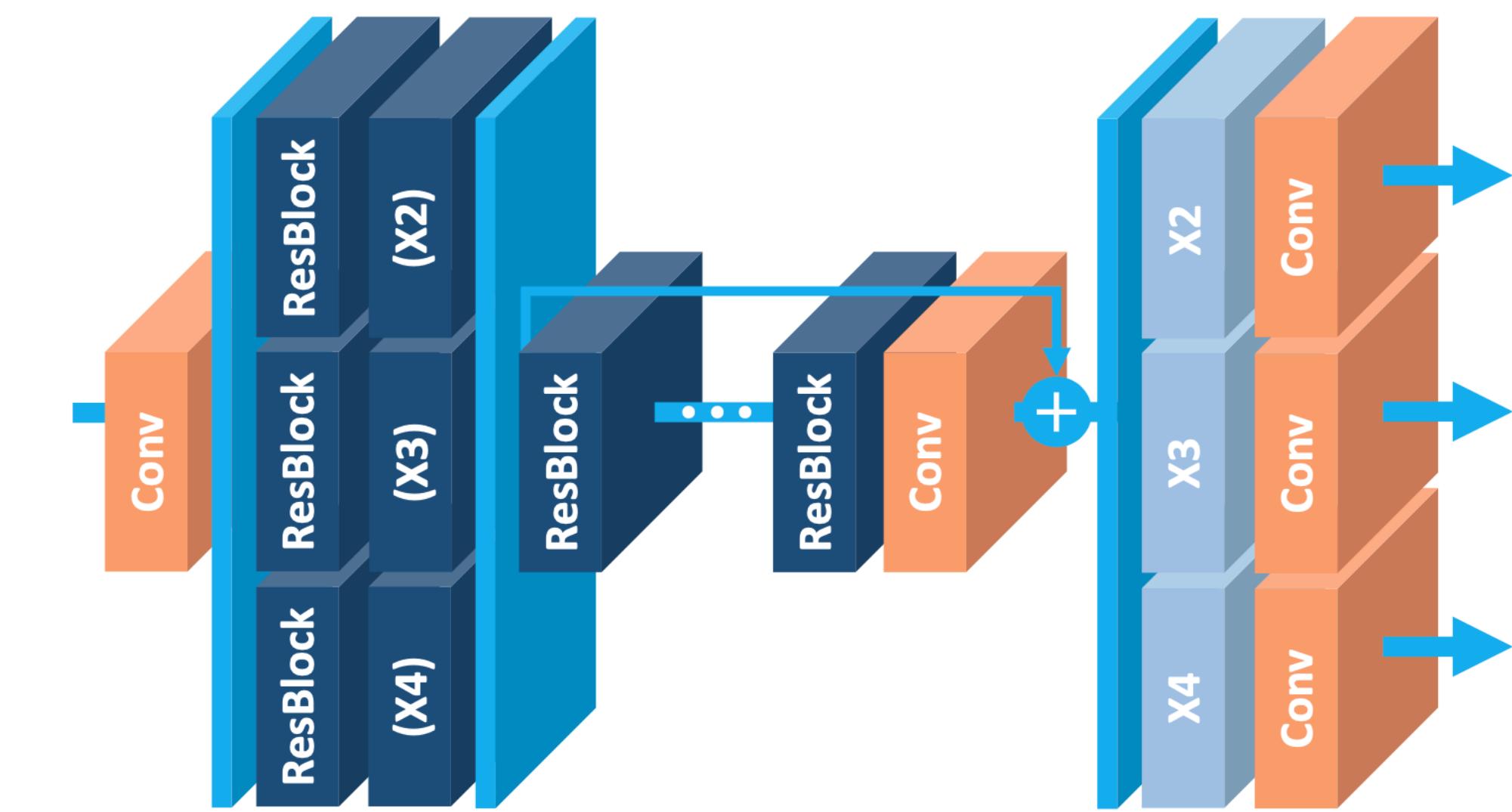
(a) Original

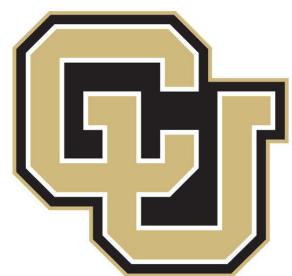


(b) SRResNet



(c) Proposed

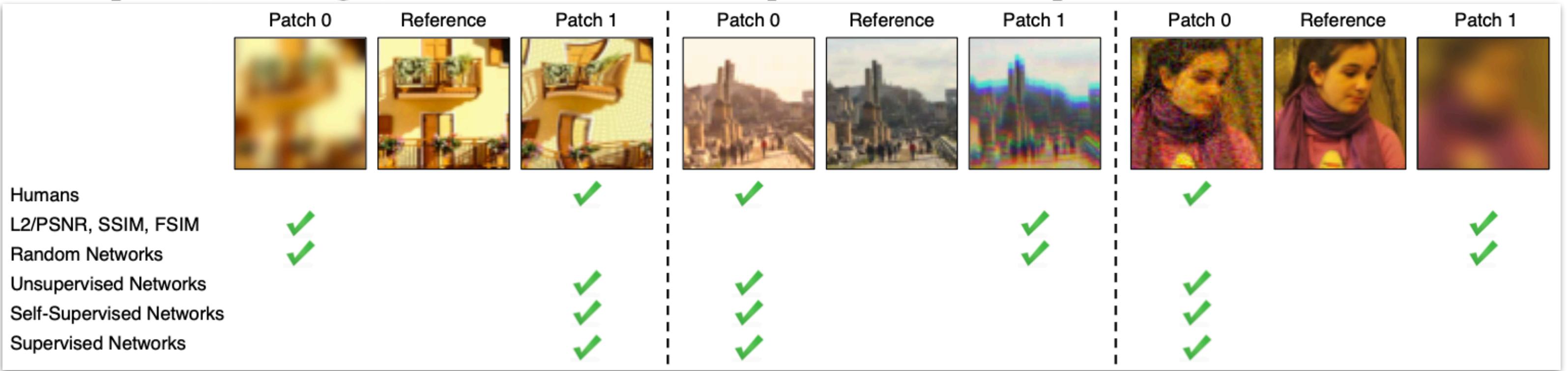




Boulder

The Unreasonable Effectiveness of Deep Features as a Perceptual Metric

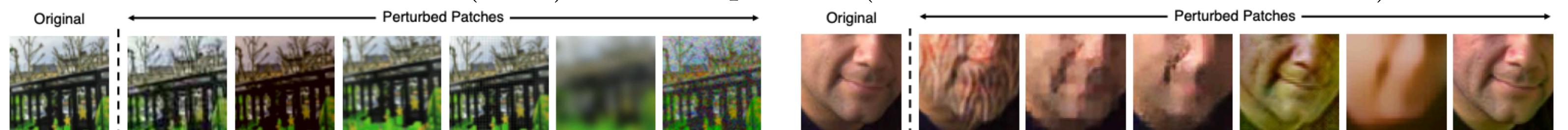
Which patch (left or right) is “closer” to the middle patch in these examples?



Perceptual Distance: Is a red circle more similar to a red square or to a blue circle?

Berkeley-Adobe Perceptual Patch Similarity (BAPPS) Dataset

- Two Alternative Forced Choice (2AFC): Which of two distortions is more similar to a reference?
 $x \rightarrow$ randomly selected image patch
 $x_0, x_1 \rightarrow$ distorted patches
 $h \in \{0, 1\} \rightarrow$ response of a human (which one is closer to the original patch?)
 $\mathcal{T} \rightarrow$ dataset of patch triplets (x, x_0, x_1, h)
 - Just Noticeable Difference (JND): Are two patches (one reference and one distorted) the same?



(a) Traditional

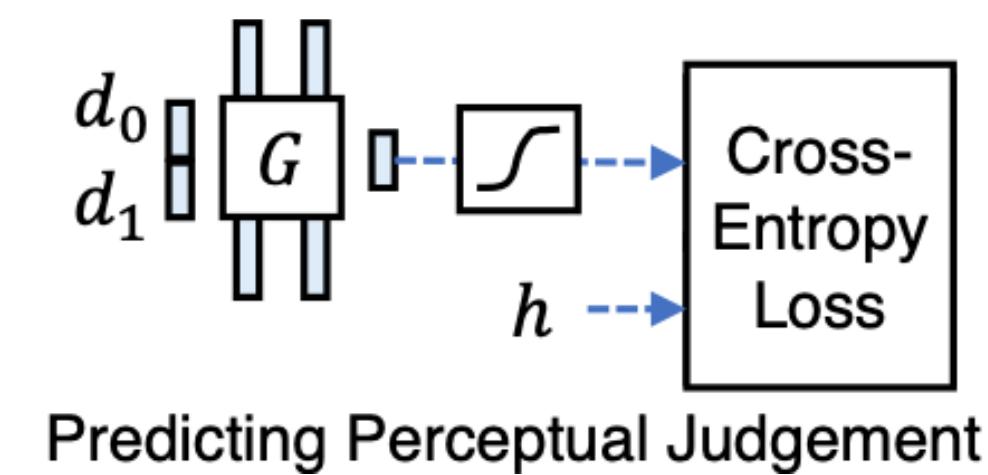
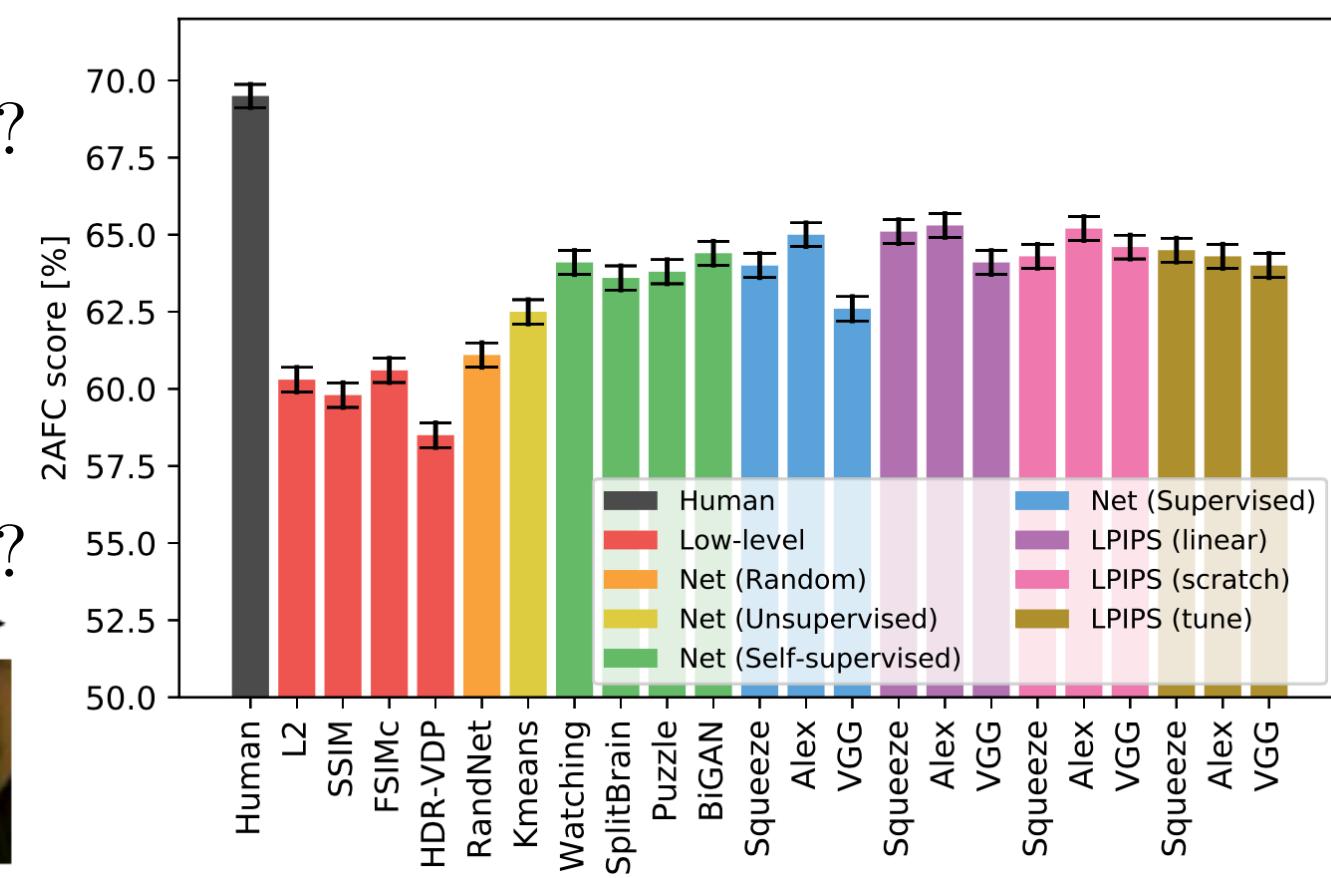
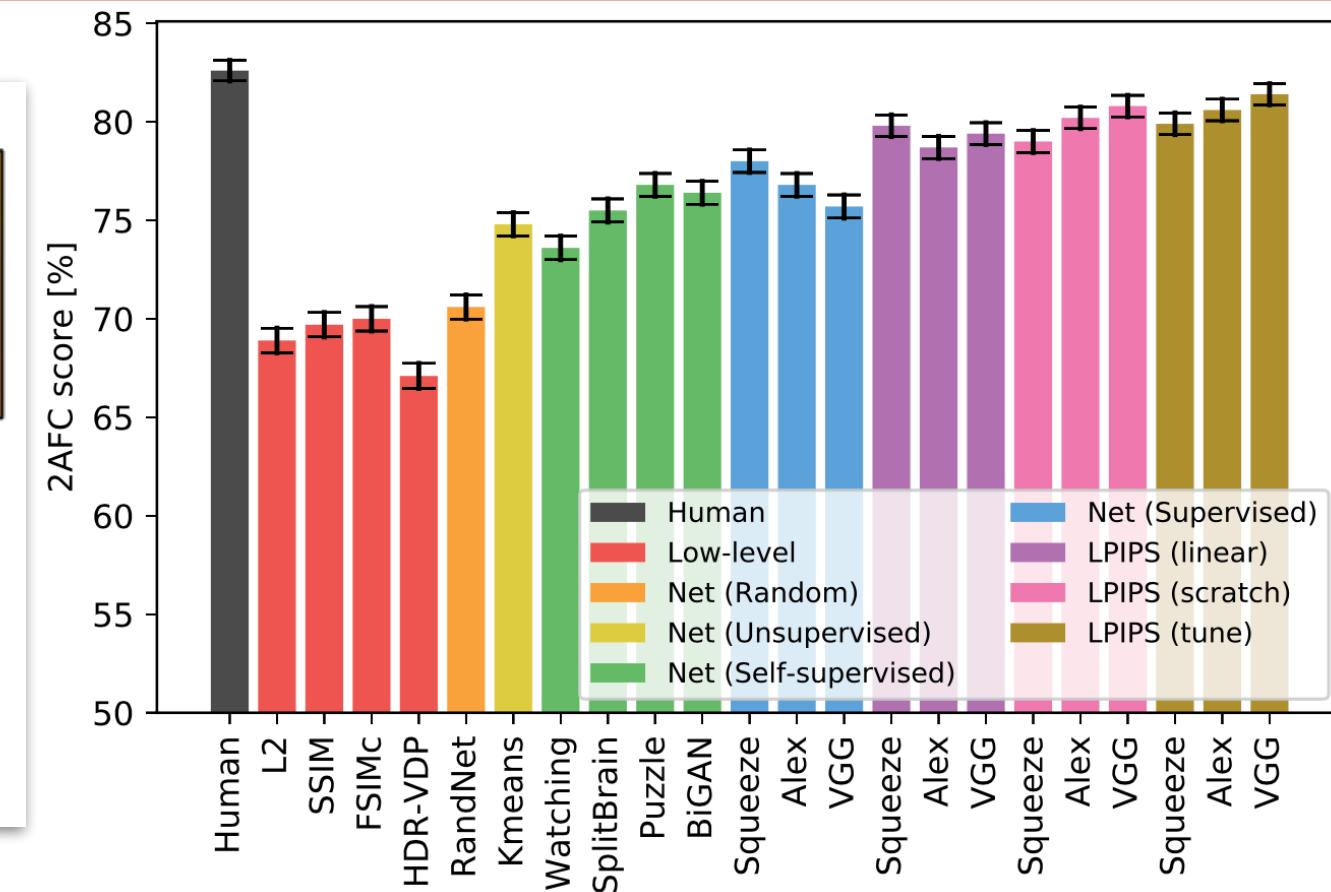
(b) CNN-based

Learned Perceptual Image Patch Similarity (LPIPS) metric

Distance between reference and distorted patches x, x_0 with network \mathcal{F} :

$$d(x, x_0) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} ||w_l \odot (\hat{y}_{hw}^l - \hat{y}_{0hw}^l)||_2^2 \quad \begin{aligned} \hat{y}^\ell, \hat{y}_0^\ell &\in \mathbb{R}^{H_\ell \times W_\ell \times C_\ell} \text{ for layer } \ell & w^\ell &\in \mathbb{R}^C \rightarrow \text{scaling factor} \\ \|\hat{y}^\ell\| &= \|\hat{y}_0^\ell\| = 1 \rightarrow \text{unit normalized in the channel dimension} \end{aligned}$$

Zhang, Richard, et al. "The unreasonable effectiveness of deep features as a perceptual metric." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.





Boulder



Questions?

[YouTube Playlist](#)
