Boulder

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

YouTube Playlist

Maziar Raissi

Assistant Professor

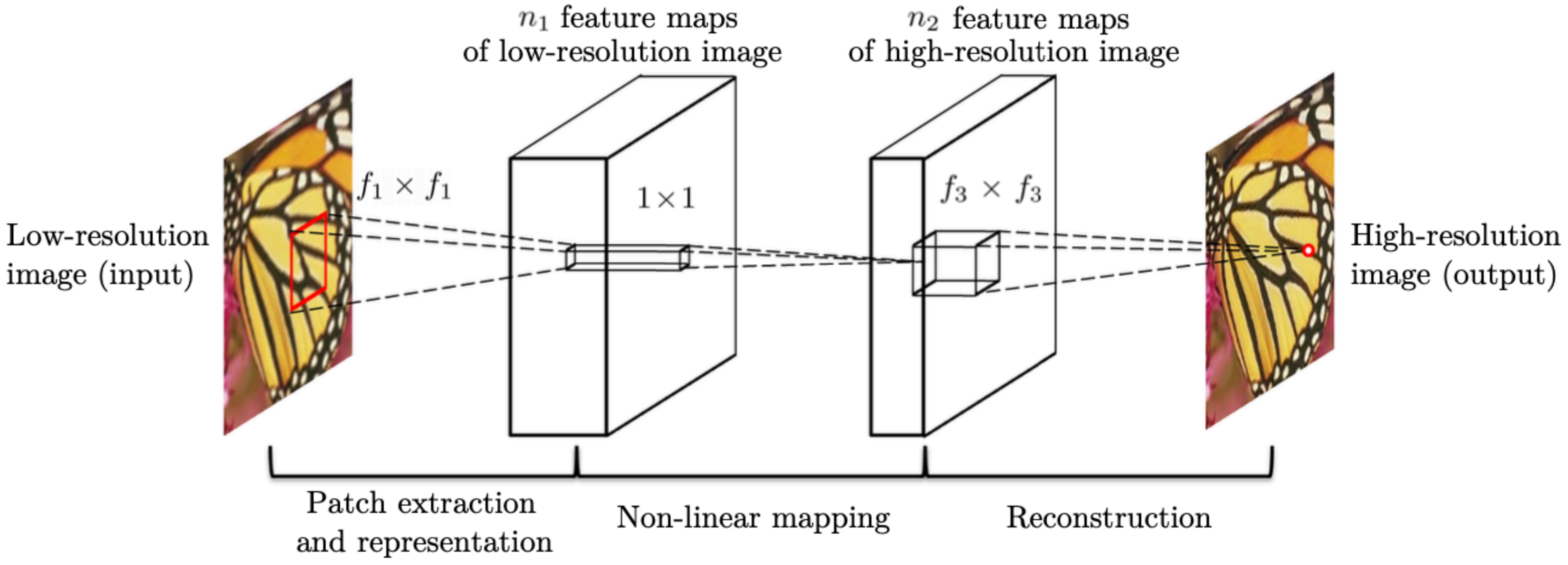
Department of Applied Mathematics

University of Colorado Boulder

maziar.raissi@colorado.edu

Learning a Deep Convolutional Network for Image Super-Resolution

YouTube Video



Super-Resolution Convolutional Neural Network (SRCNN)

Sparse Coding (SC) based method

Peak Signal-to-Noise Ratio (PSNR)

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

$$MAX_I = 255$$

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

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

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

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

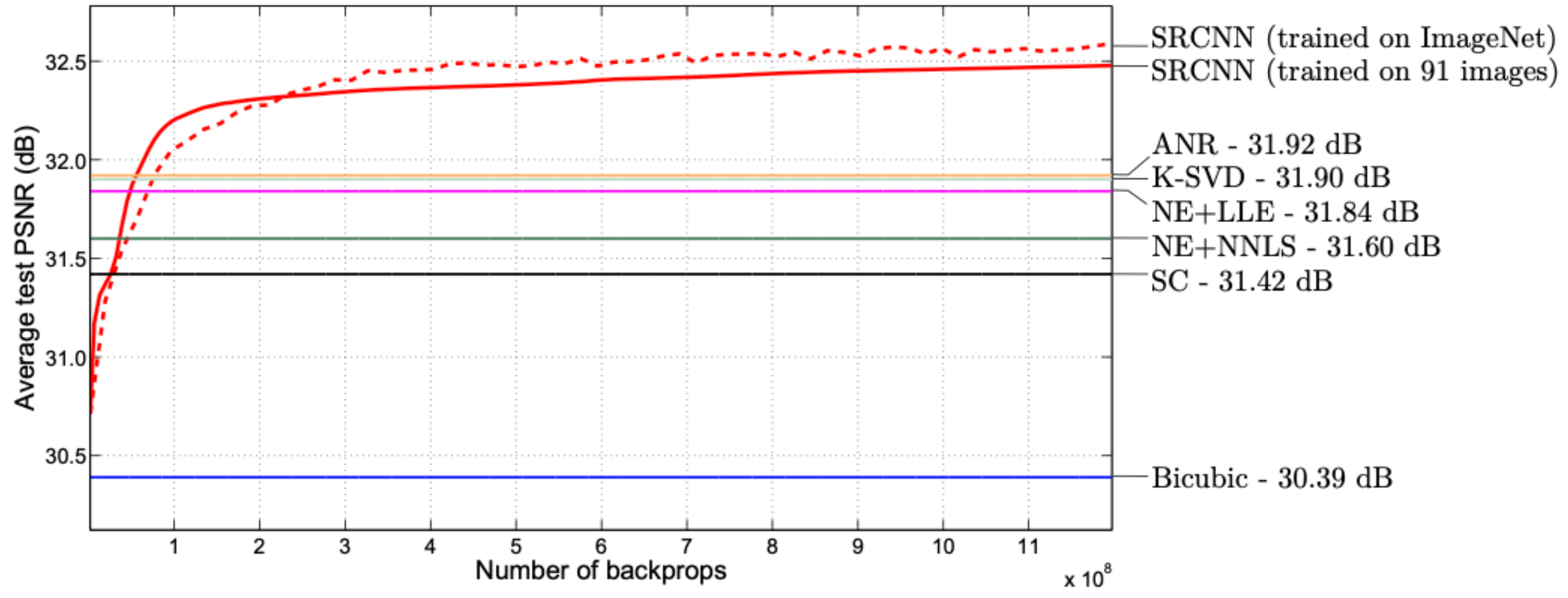
$$F_2(Y) = \max(0, W_2 * F_1(Y) + B_2)$$

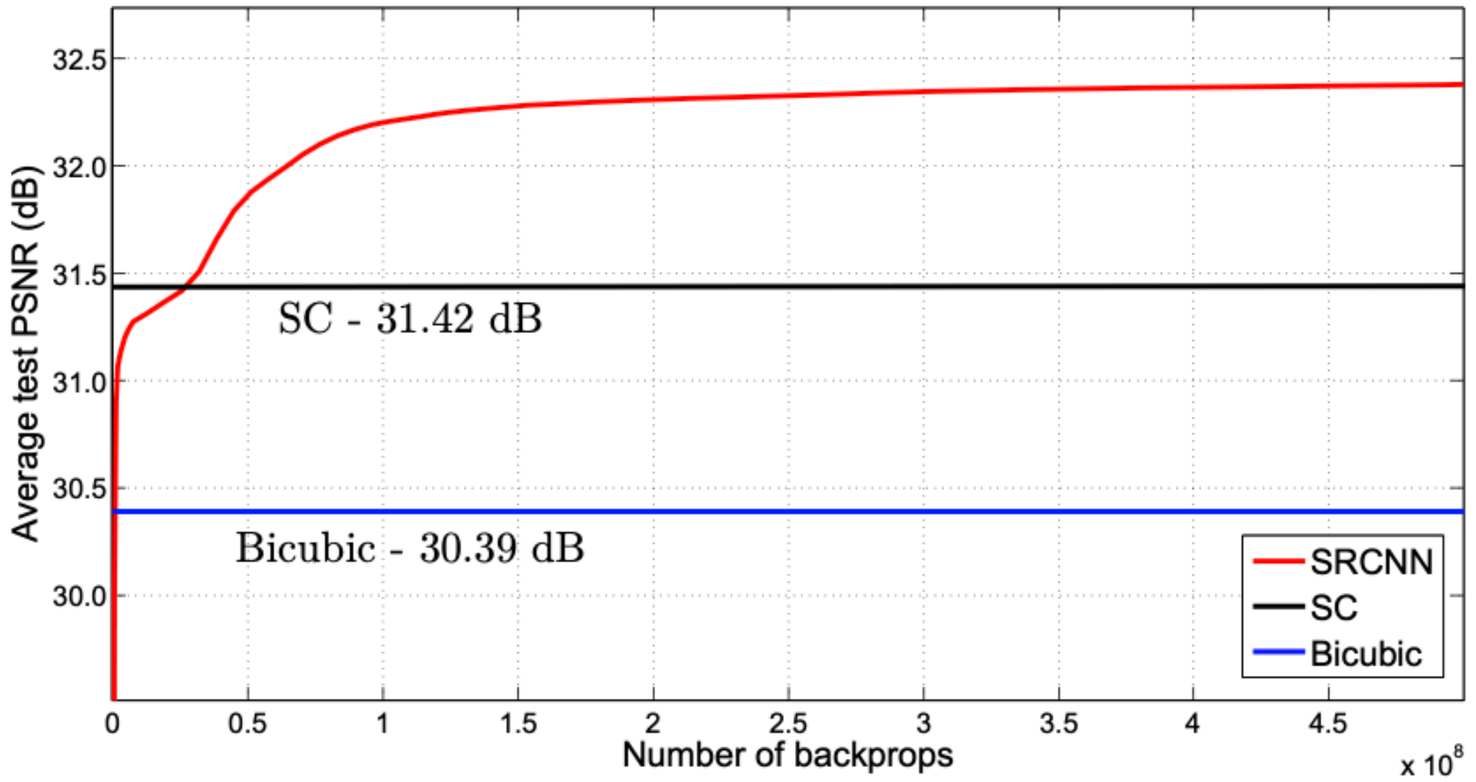
$$W_2 \in \mathbb{R}^{n_1 \times 1 \times 1 \times n_2} \qquad B_2 \in \mathbb{R}^{n_2}$$

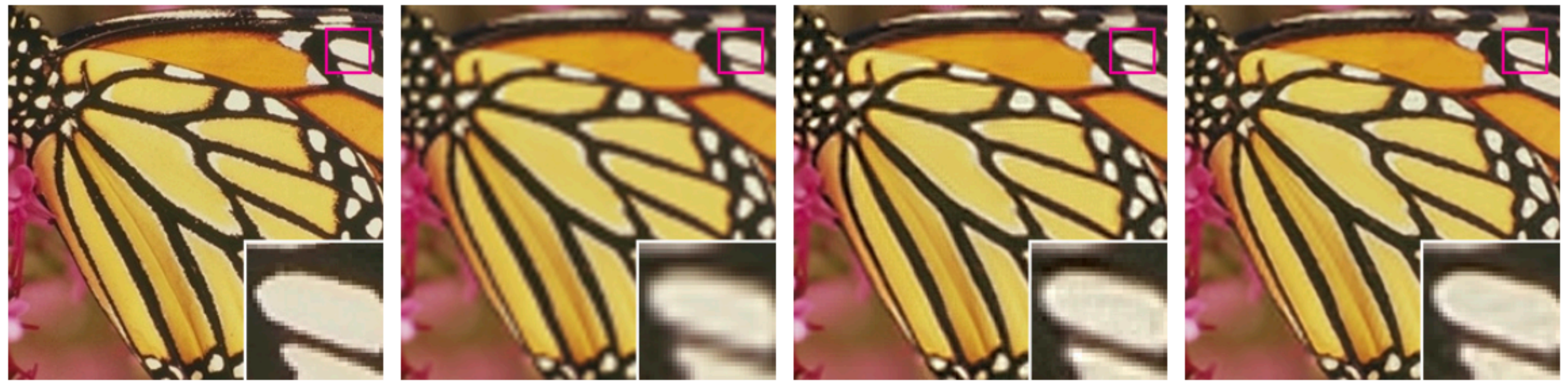
$$F_3(Y) = W_3 * F_2(Y) + B_3$$

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

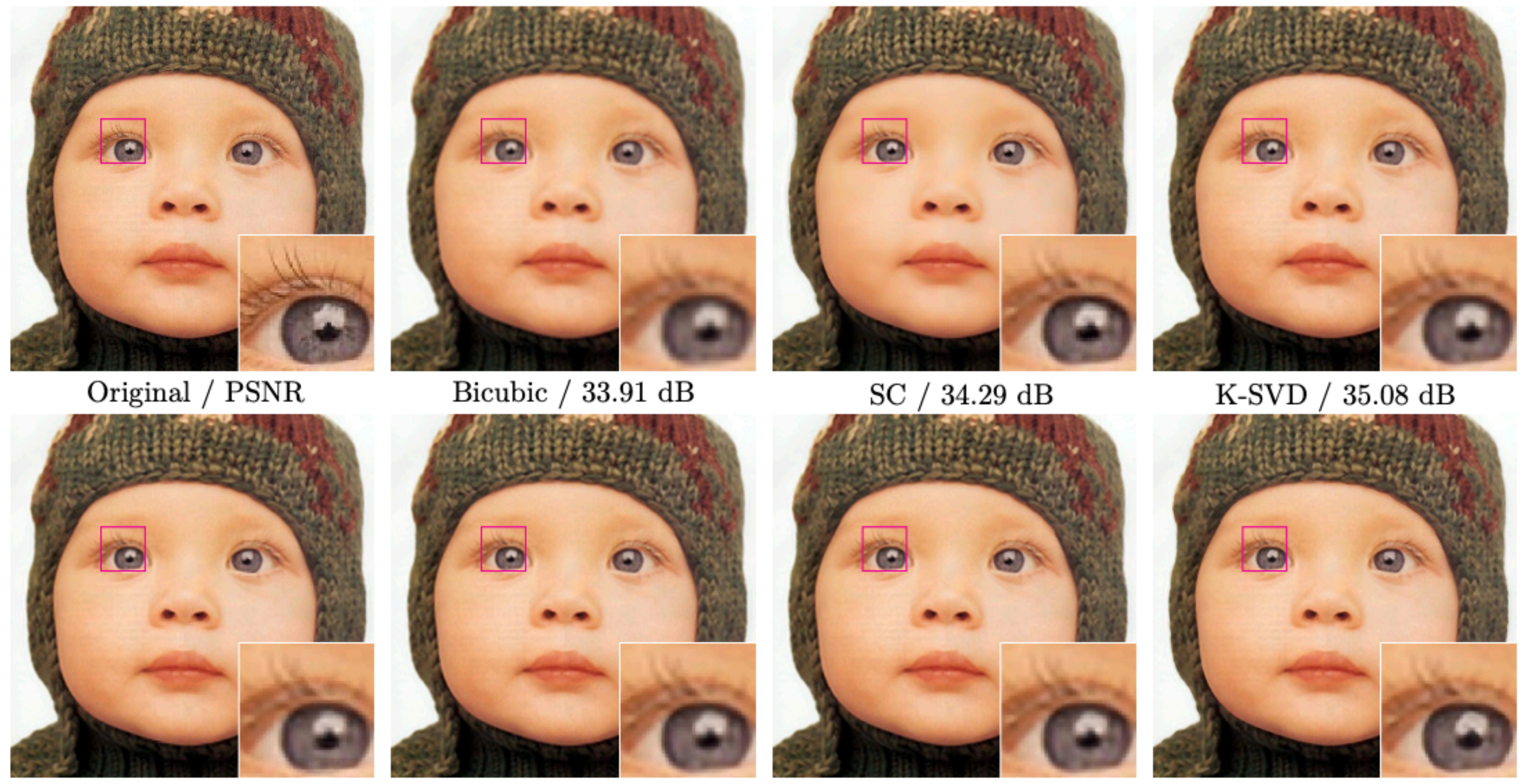
$$L(\Theta) = \frac{1}{n}\sum_{i=1}^{n} \|F(Y_i;\Theta) - X_i\|^2$$







Original / PSNR — Bicubic / 24.04 dB — SC / 25.58 dB — SRCNN / 27.58 dB



Original / PSNR — Bicubic / 33.91 dB — SC / 34.29 dB — K-SVD / 35.08 dB

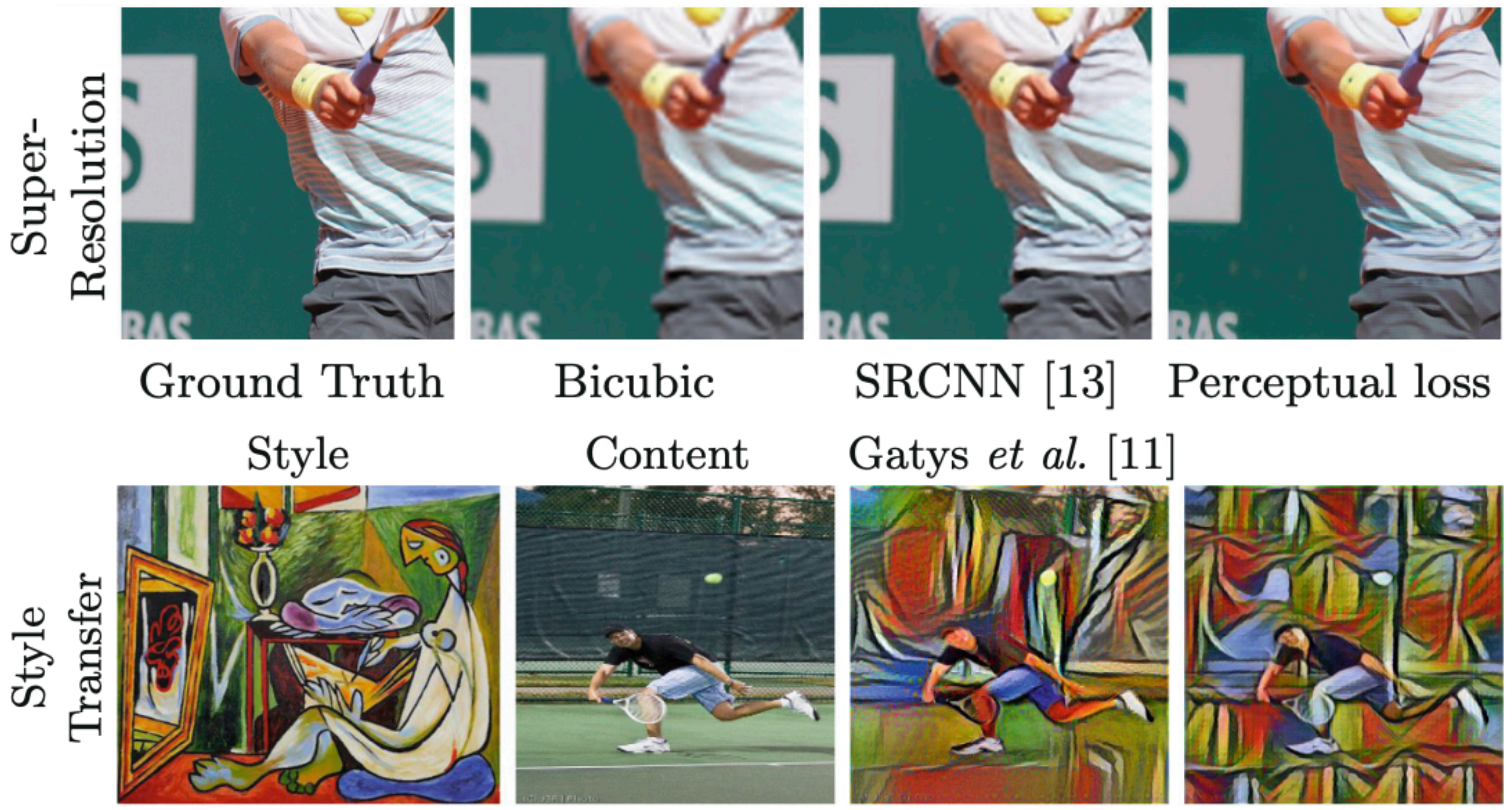
NE+NNLS / 34.77 dB — NE+LLE / 35.06 dB — ANR / **35.13** dB — SRCNN / 35.01 dB

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

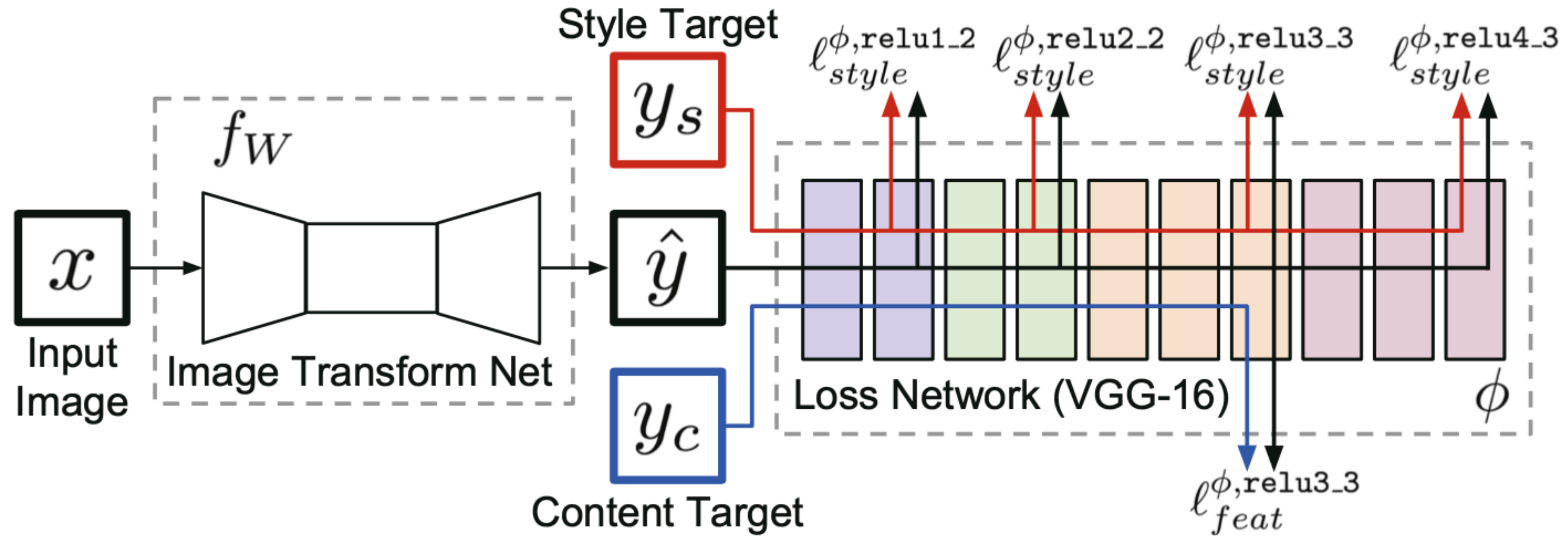
Dong, Chao, et al. "Image super-resolution using deep convolutional networks." *IEEE transactions on pattern analysis and machine intelligence* 38.2 (2015): 295-307.

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: $y_c =$ high-resolution image & style reconstruction loss is not used

Feature Reconstruction Loss

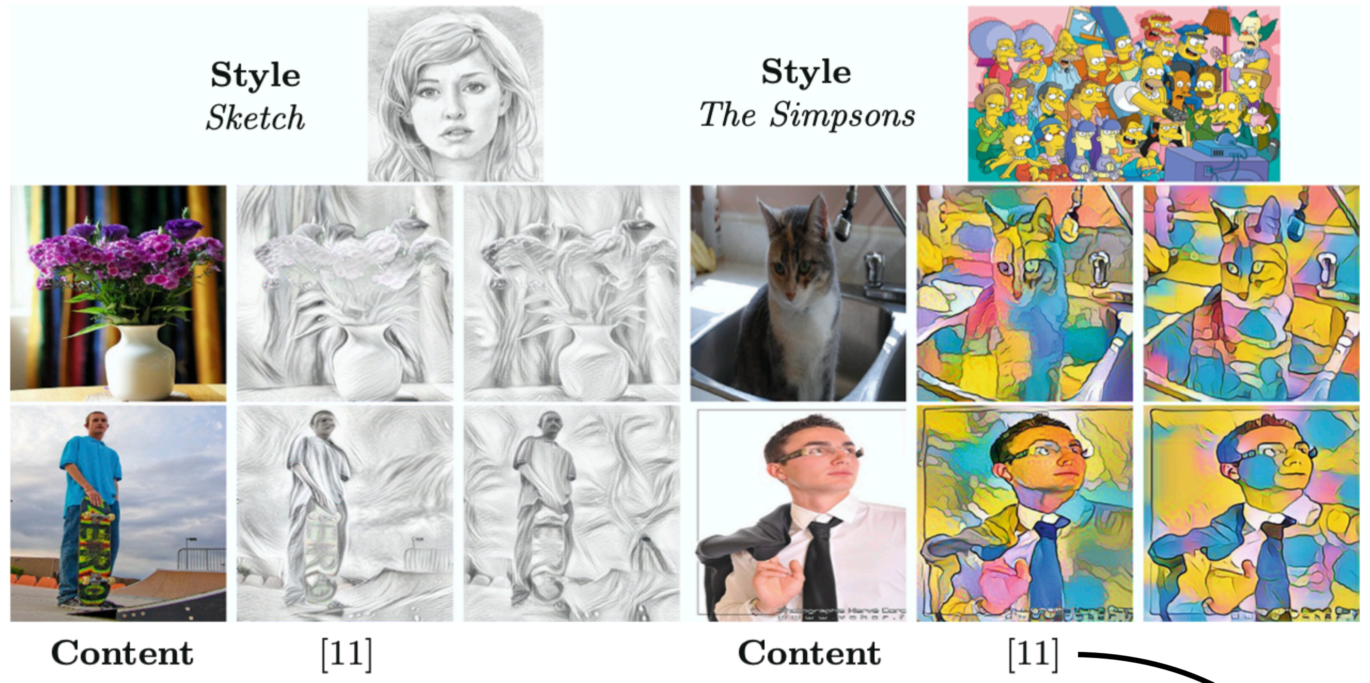
$\phi_j(x) \in \mathbb{R}^{C_j \times H_j \times W_j} \rightarrow$ activations of the j-th layer of network ϕ

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

Style Reconstruction Loss

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

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

Johnson, Justin, Alexandre Alahi, and Li Fei-Fei. "Perceptual losses for real-time style transfer and super-resolution." *European conference on computer vision*. Springer, Cham, 2016.

Image Style Transfer Using Convolutional Neural Networks

Boulder

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

$$\mathcal{L}_{\text{content}}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} \left(F^l_{ij} - P^l_{ij}\right)^2$$

$F^l_{i,j} \rightarrow$ 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

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

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

captures texture but not global arrangement

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

$G^l \rightarrow$ style representation of x

$$\mathcal{L}_{\text{style}}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l \qquad E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} \left(G^l_{ij} - A^l_{ij}\right)^2$$

Style Transfer

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

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

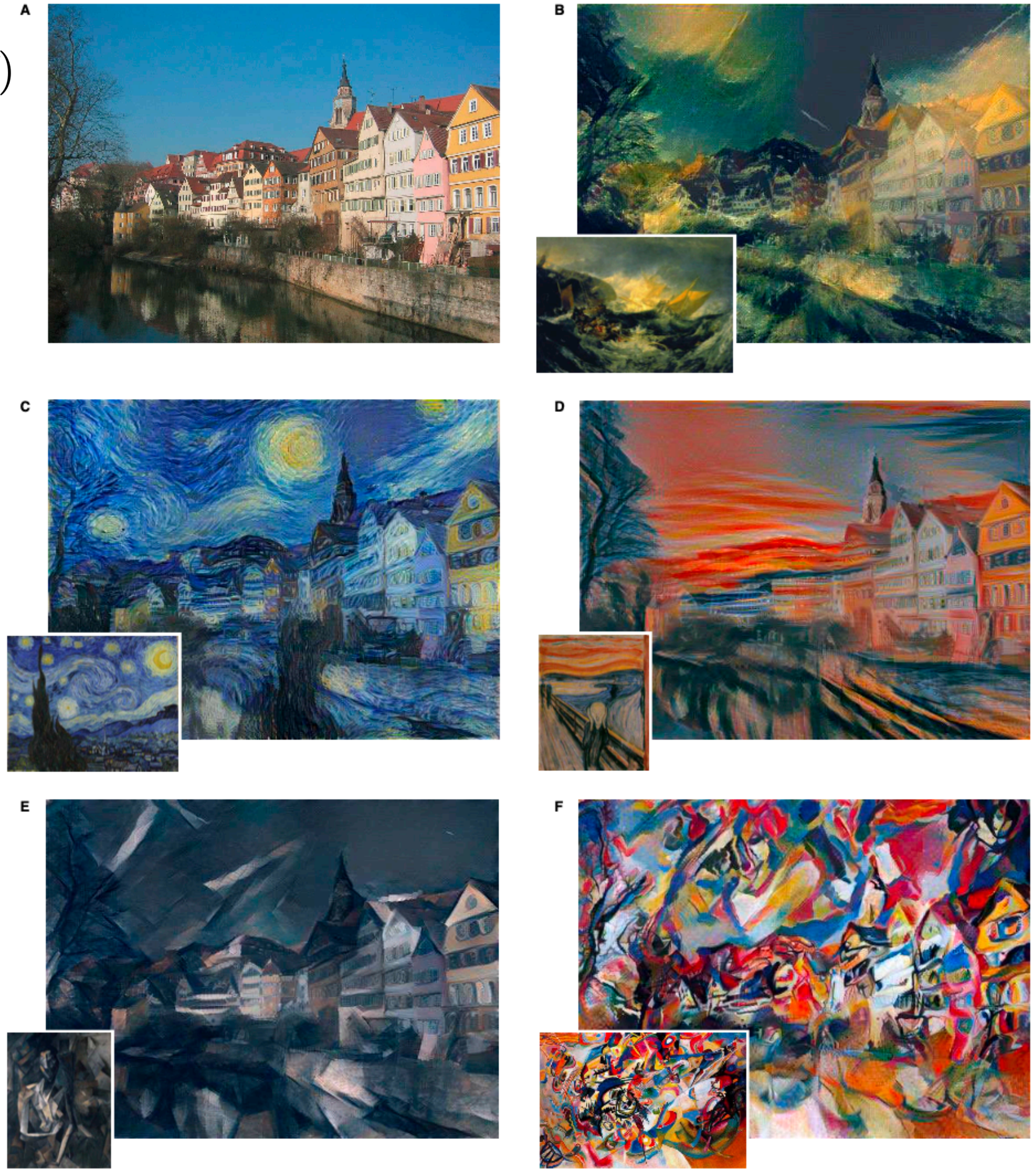
Style Image

Content Image

10^{-4}

10^{-1}

The ratio α/β



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.

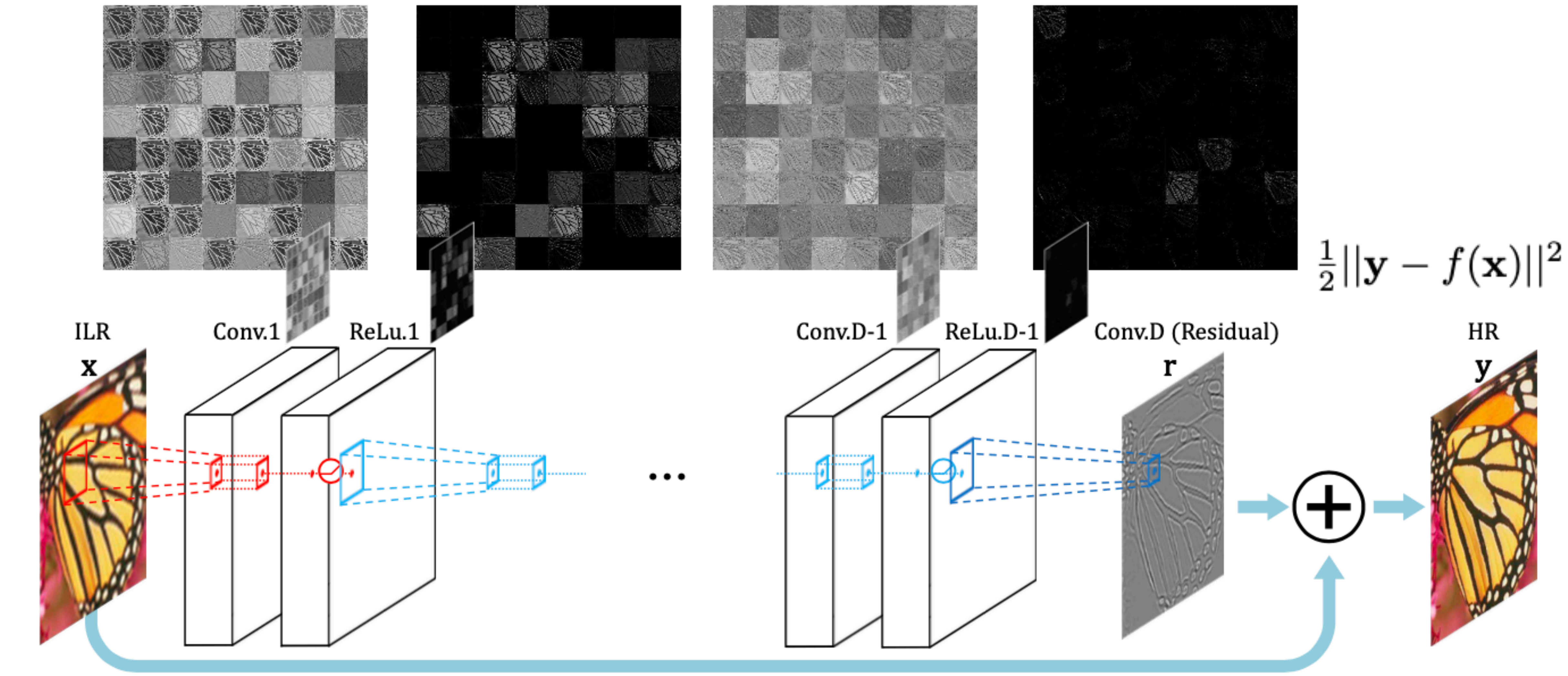
Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "A neural algorithm of artistic style." *arXiv preprint arXiv:1508.06576* (2015).

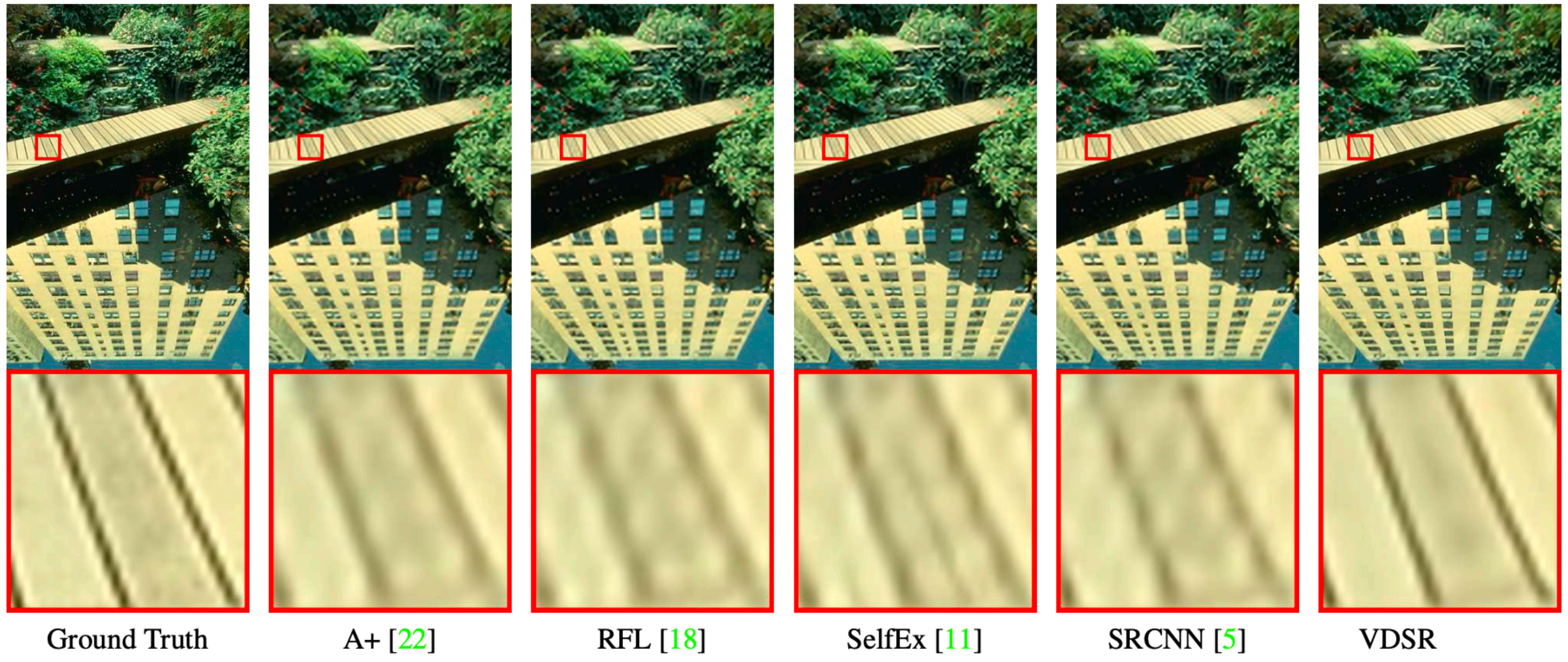
Boulder

Accurate Image Super-Resolution Using Very Deep Convolutional Networks

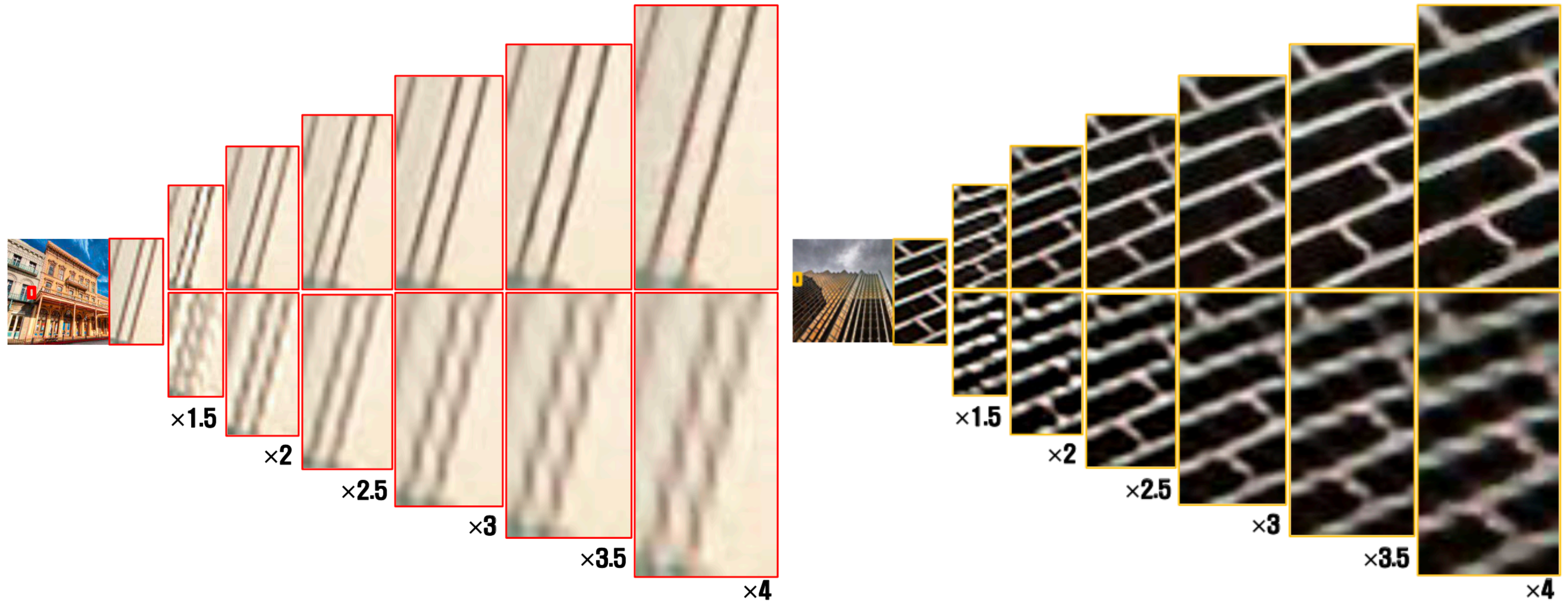
YouTube Video

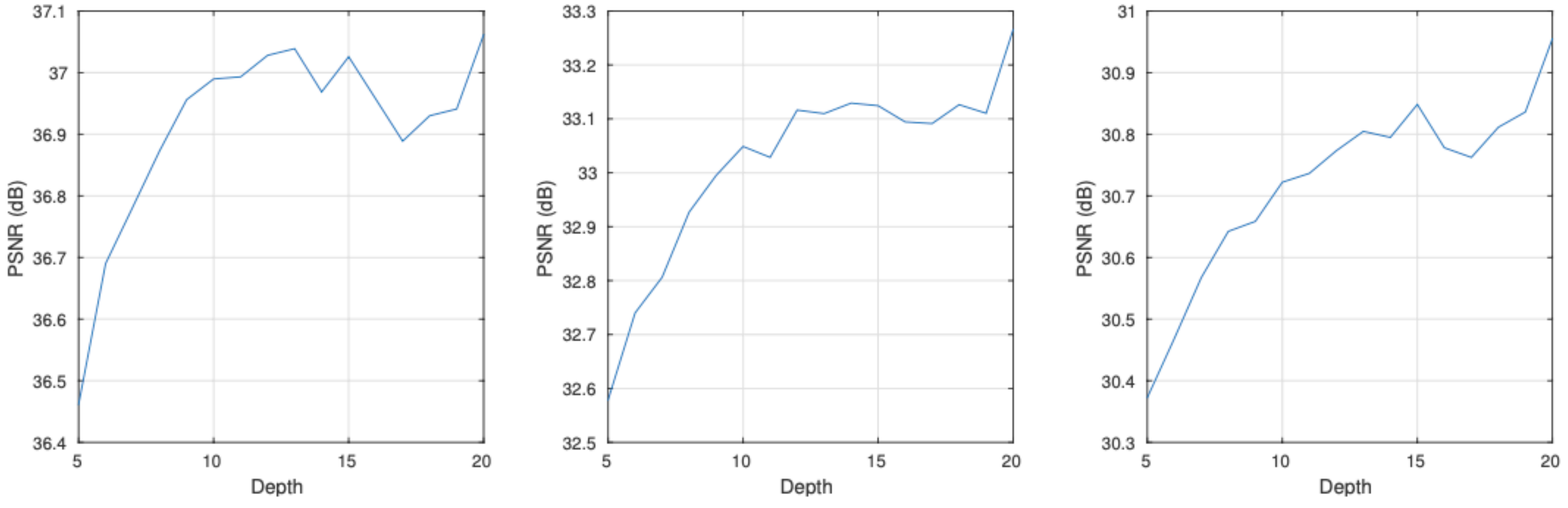
Single Image Super-Resolution (SISR)





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





(a) Test Scale Factor 2 **(b)** Test Scale Factor 3 **(c)** Test Scale Factor 4

Kim, Jiwon, Jung Kwon Lee, and Kyoung Mu Lee. "Accurate image super-resolution using very deep convolutional networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network

Boulder

$I^{\mathrm{HR}} \rightarrow$ high-resolution image

$I^{\mathrm{LR}} \rightarrow$ low-resolution image

$I^{\mathrm{LR}} \leftarrow I^{\mathrm{HR}}$ ▷ Gaussian filter

▷ Downsampling

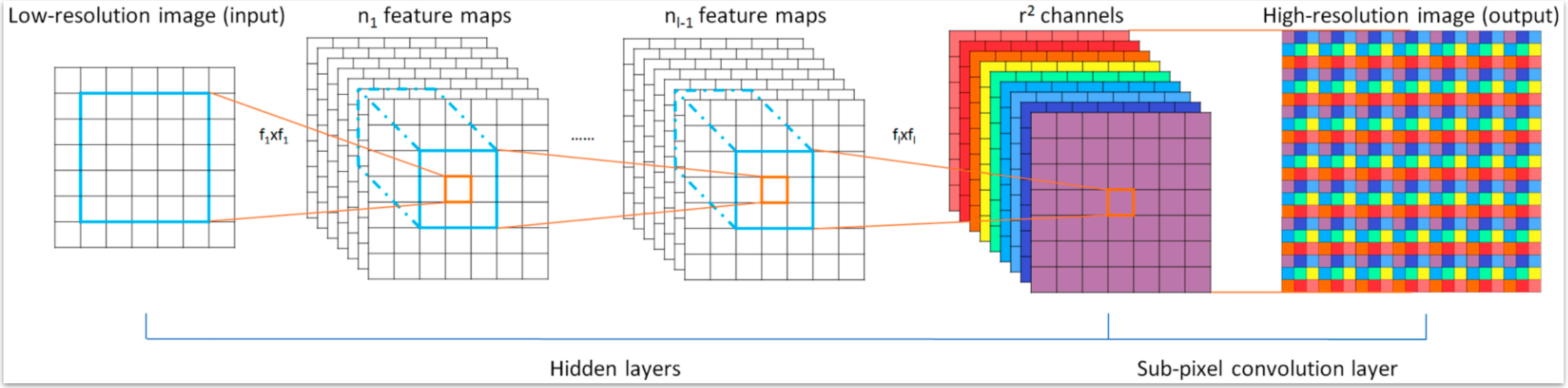
$I^{\mathrm{SR}} \rightarrow$ super-resolved image

$r \rightarrow$ upscaling ratio

$I^{\mathrm{LR}} \in \mathbb{R}^{H \times W \times C}$

$I^{\mathrm{HR}} \in \mathbb{R}^{rH \times rW \times C}$

avoid upscaling I^{LR} (i.e., bicubic interpolation) before feeding into the network



$$f^1(I^{\mathrm{LR}}; W_1, b_1) = \tanh(W_1 * I^{\mathrm{LR}} + b_1)$$

$$f^l(I^{\mathrm{LR}}; W_{1:l}, b_{1:l}) = \tanh(W_l * f^{l-1}(I^{\mathrm{LR}}) + b_l), l = 2, \ldots, L-1$$

$$W_l \in \mathbb{R}^{n_{l-1} \times n_l \times k_l \times k_l}, b_l \in \mathbb{R}^{n_l}$$

$n_l \rightarrow$ number of features at layer l

$n_0 = C, k_l \rightarrow$ filter size at layer l

Efficient sub-pixel convolution layer

$$\mathbf{I}^{SR} = f^L(\mathbf{I}^{LR}) = \mathcal{PS}\left(W_L * f^{L-1}(\mathbf{I}^{LR}) + b_L\right)$$

$\mathcal{PS} \rightarrow$ periodic shuffling operator $\quad W_L \in \mathbb{R}^{n_{L-1} \times r^2 C \times k_L \times k_L}$

$$H \times W \times C \cdot r^2 \mapsto rH \times rW \times C$$

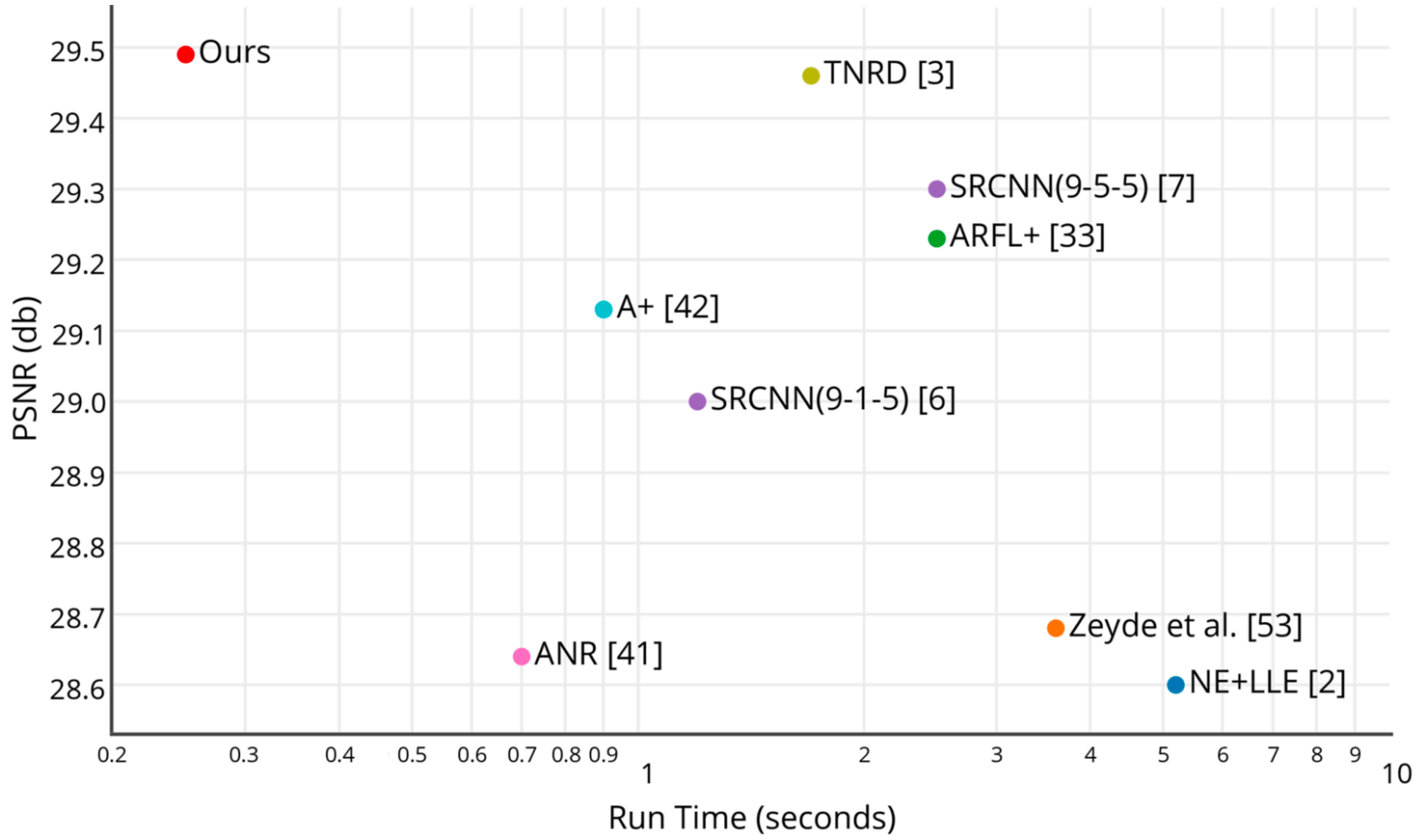
$$\mathcal{PS}(T)_{x,y,c} = T_{\lfloor x/r \rfloor, \lfloor y/r \rfloor, c \cdot r \cdot \mathrm{mod}(y,r) + c \cdot \mathrm{mod}(x,r)}$$

equivalent to a fractionally-strided convolution

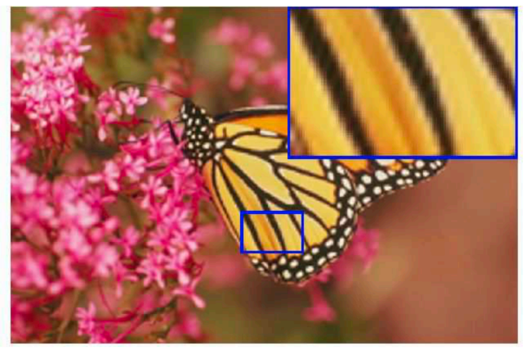
convolution with stride $1/r$ in the LR space

with a filter W_s of size $k_s = rk_L$, but more efficient!

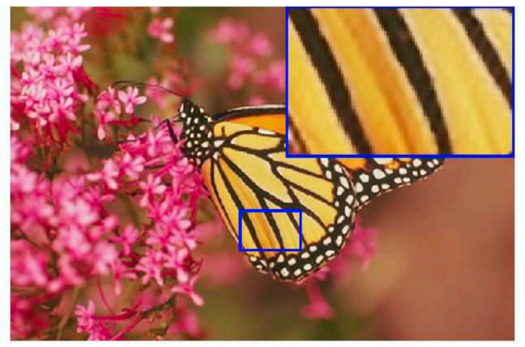
$$\ell(W_{1:L}, b_{1:L}) = \frac{1}{r^2 HW} \sum_{x=1}^{rH} \sum_{x=1}^{rW} \left(\mathbf{I}^{HR}_{x,y} - f^L_{x,y}(\mathbf{I}^{LR})\right)^2$$



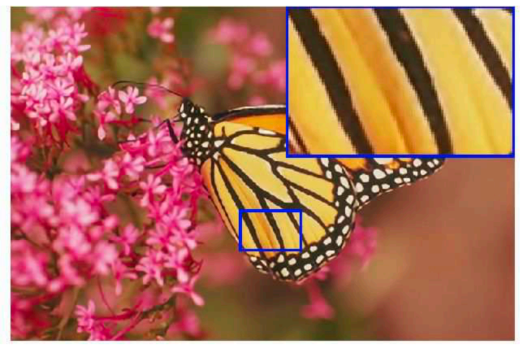
(k) Monarch Original



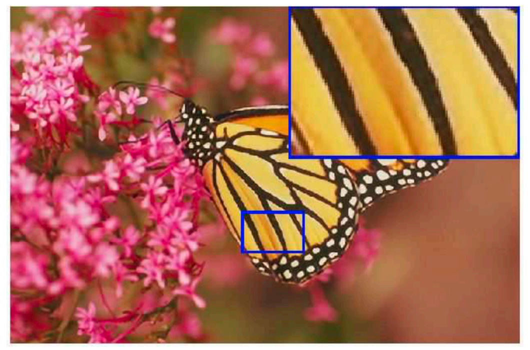
(l) Bicubic / 29.43db



(m) SRCNN [7] / 32.81db



(n) TNRD [3] / 33.62db



(o) ESPCN / **33.66db**

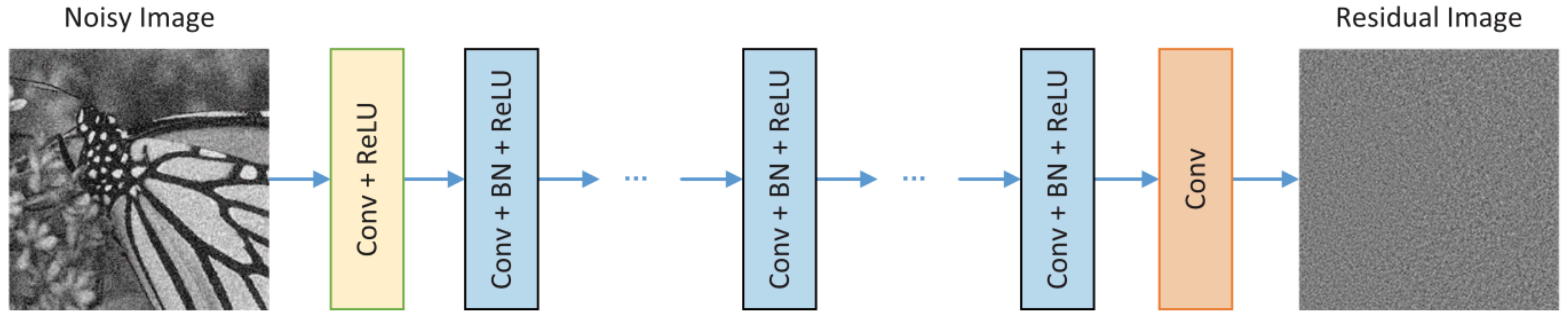
Shi, Wenzhe, et al. "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

Boulder

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

YouTube Video

Denoising Convolutional Neural Networks (DnCNNs)



$x \to$ clean image

$y \to$ noisy observation

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

AWGN (Additive White Gaussian Noise) with standard deviation σ

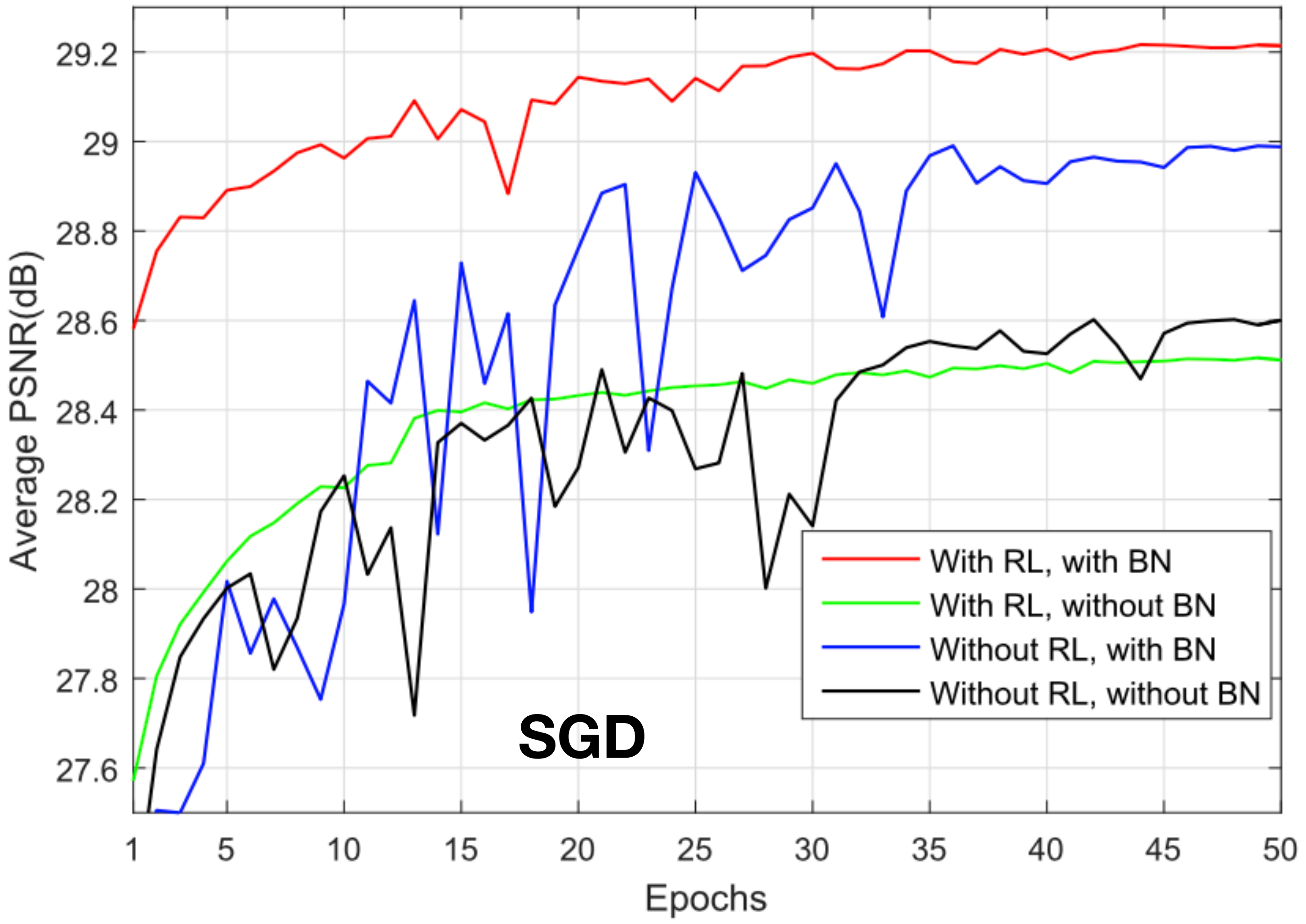
Receptive Field of DnCNN (3×3 conv) with depth d is $(2d+1) \times (2d+1)$

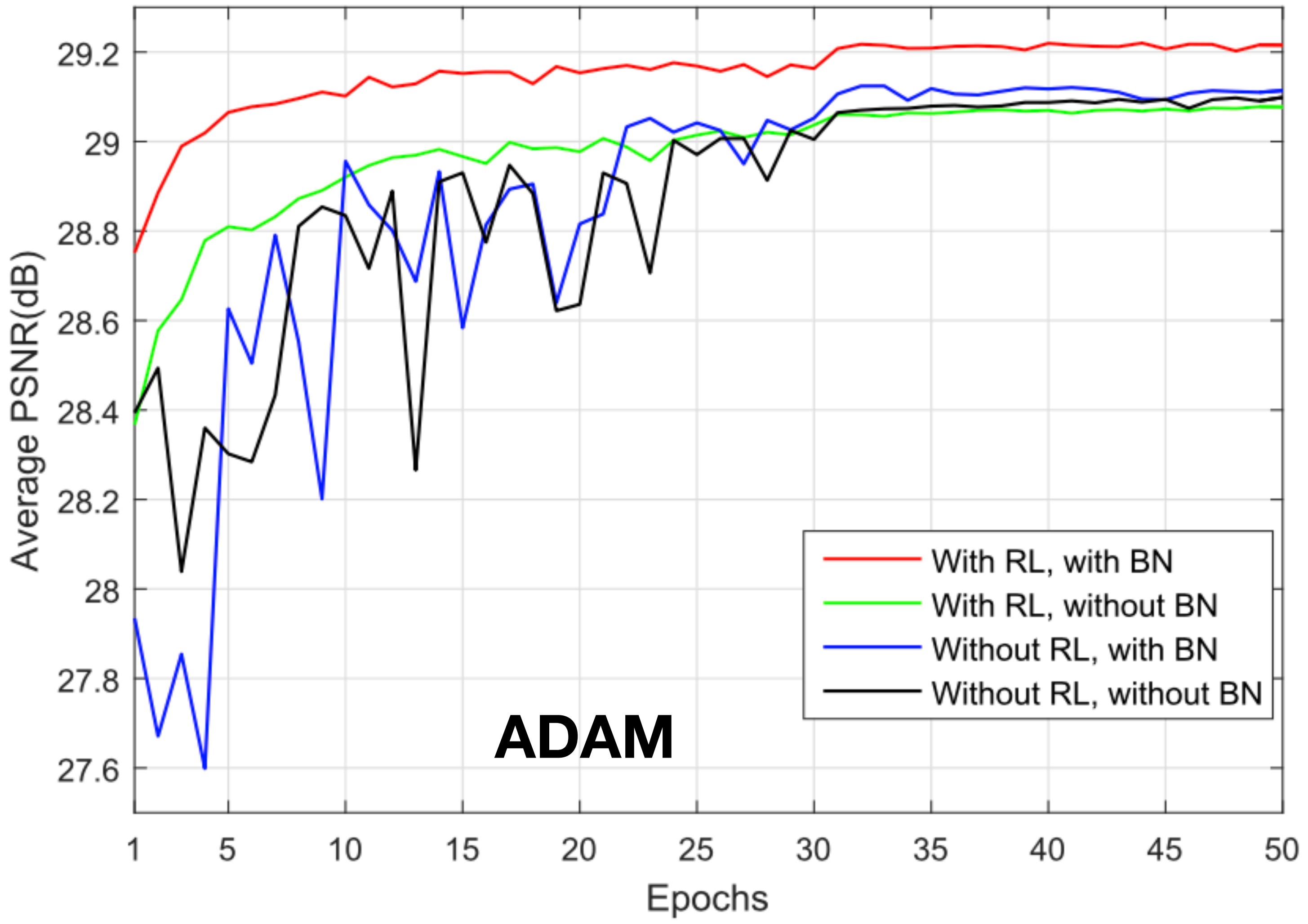
$\mathcal{F}(y) = x$

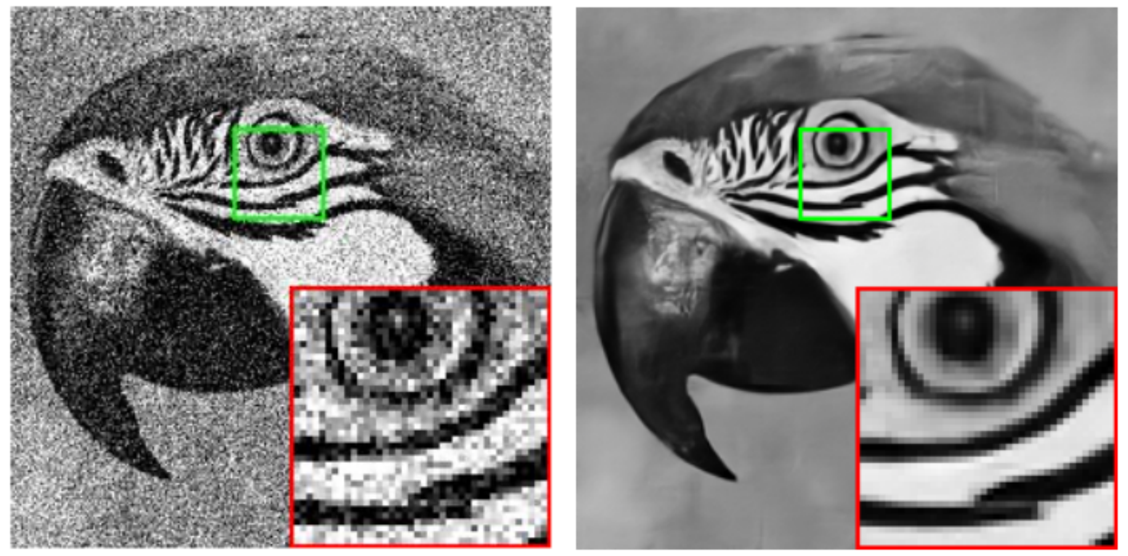
$\mathcal{R}(y) \approx v \to$ residual mapping

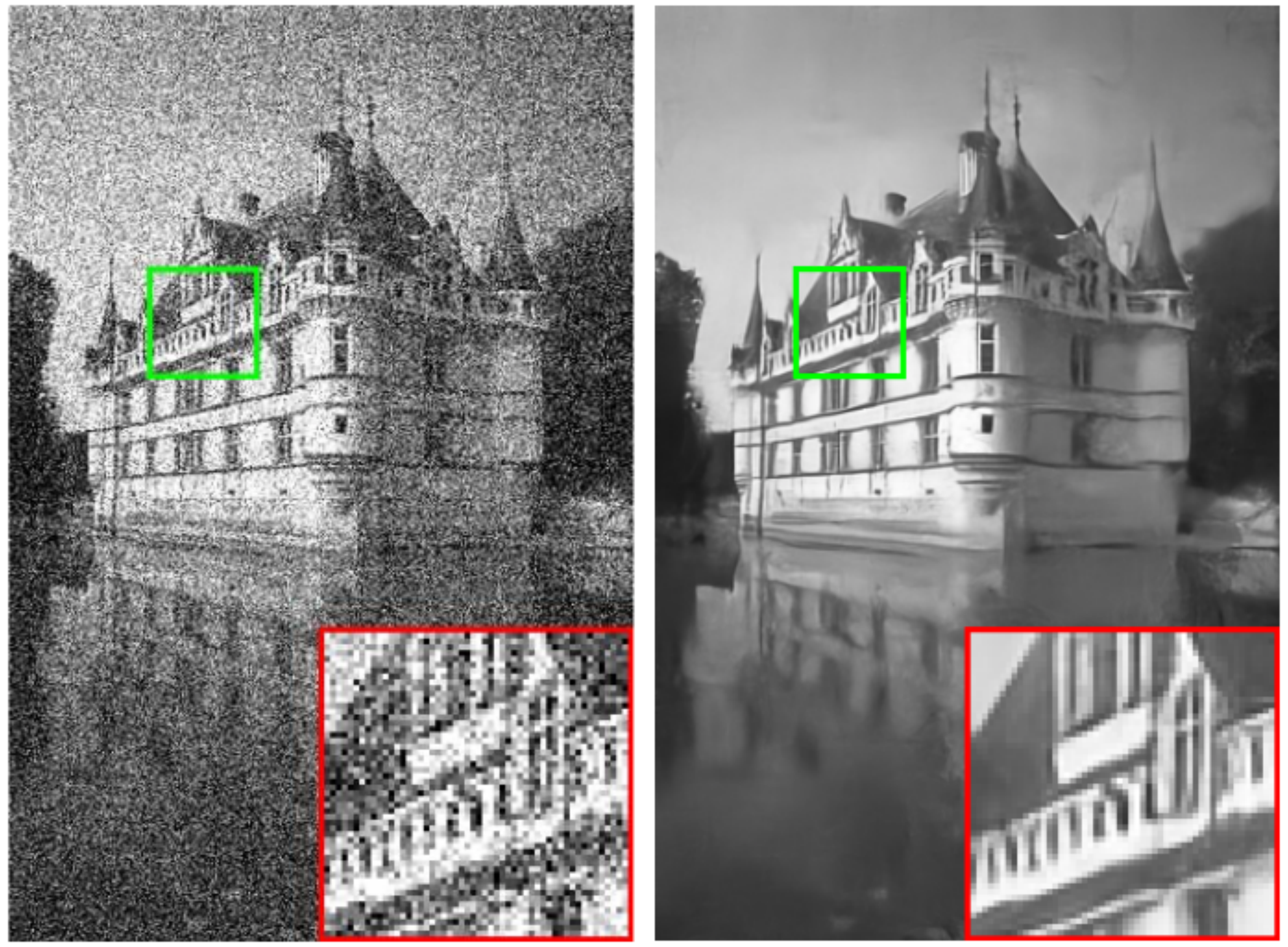
$\implies x = y - \mathcal{R}(y)$

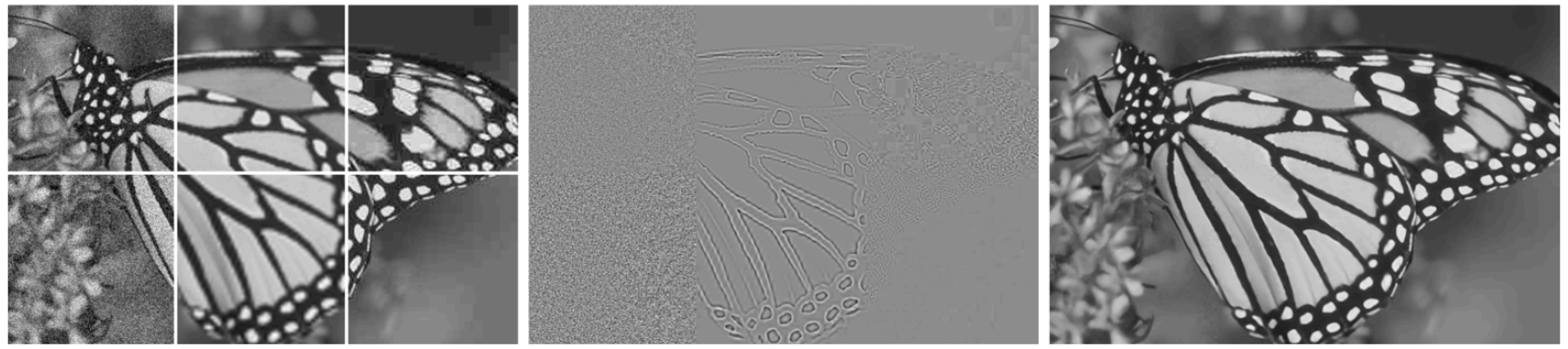
$$\ell(\theta) = \frac{1}{2N} \sum_{i=1}^{N} \|R(y_i; \theta) - (y_i - x_i)\|_F^2$$







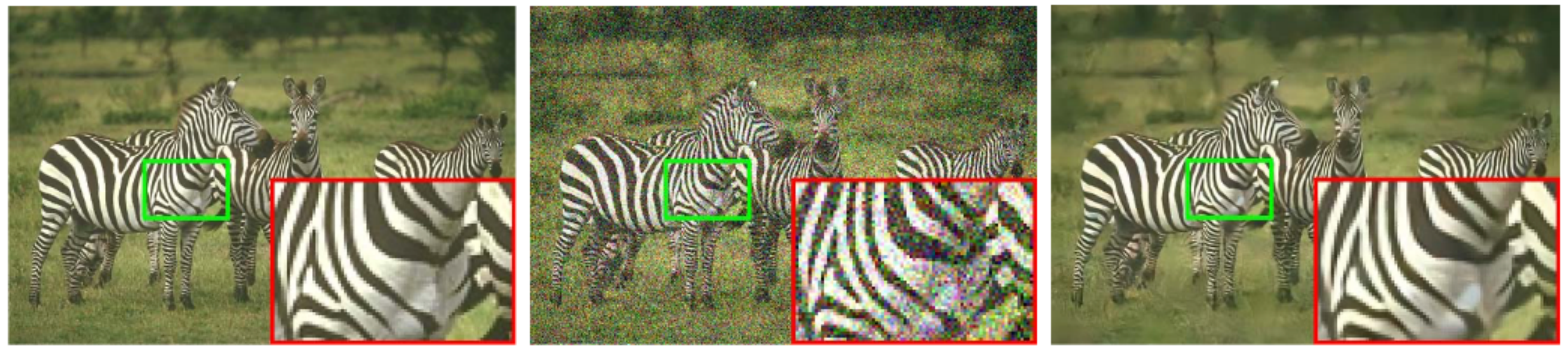




JPEG image de-blocking

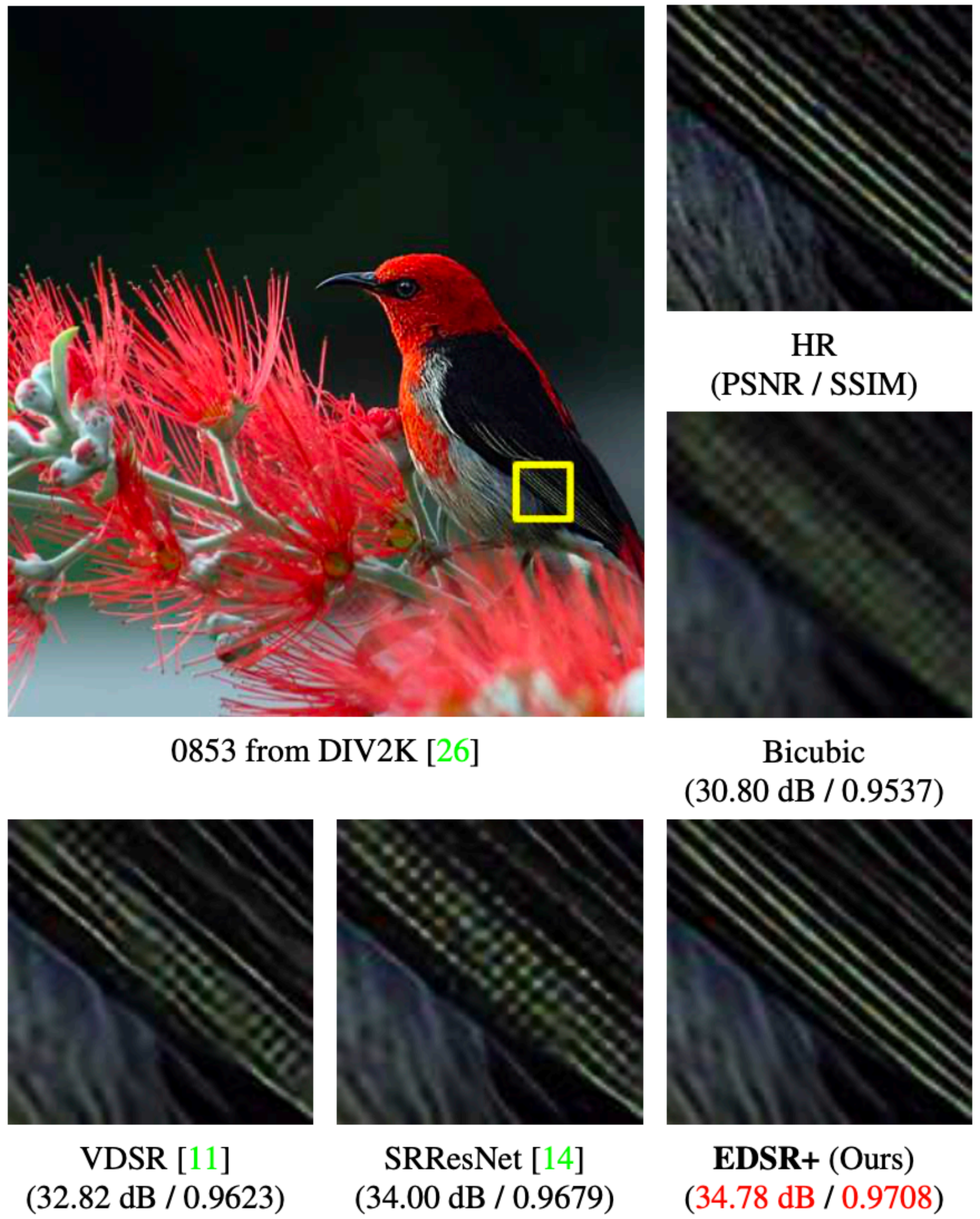
Single Image Super-resolution

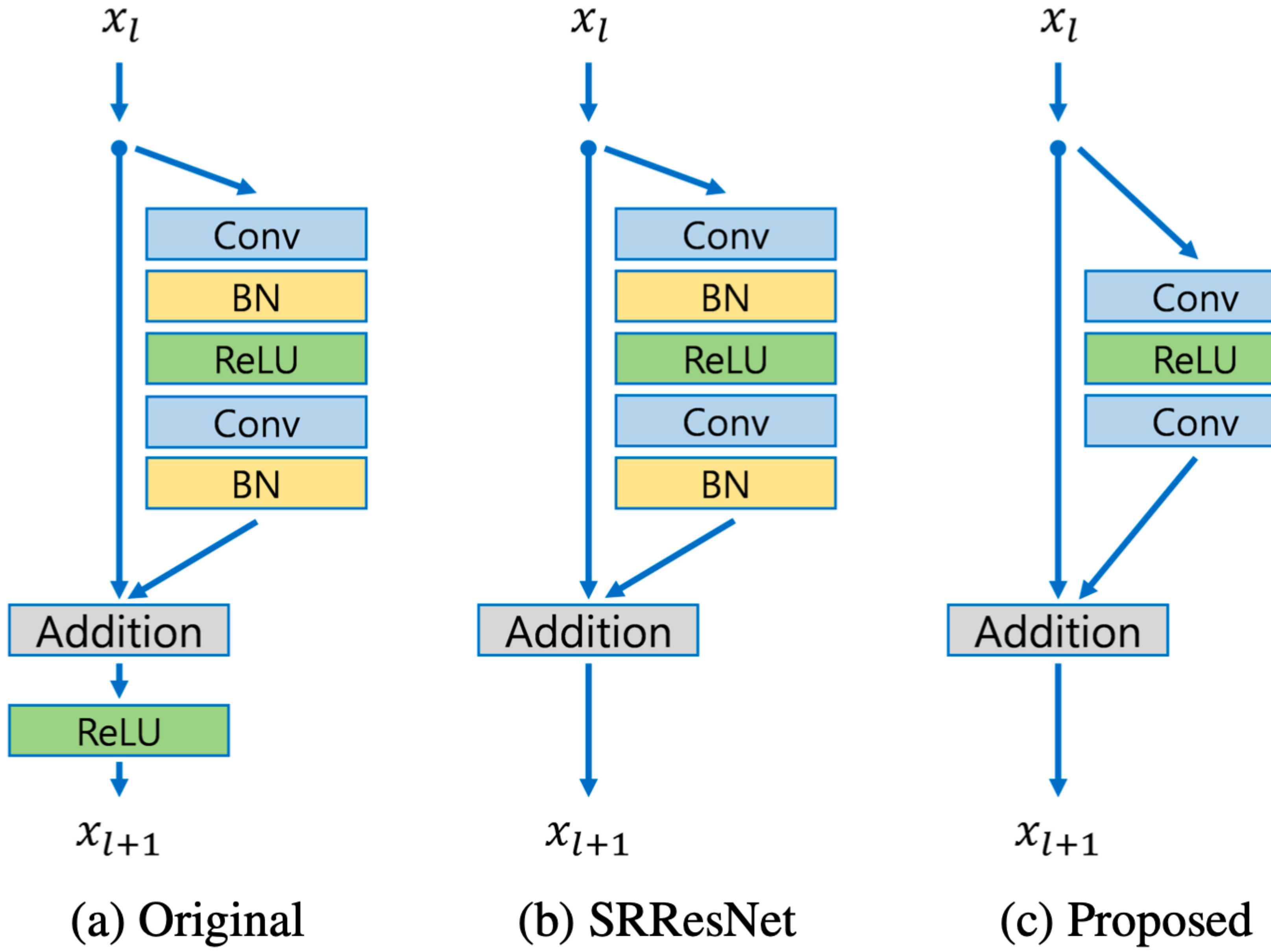
Gaussian Denoising

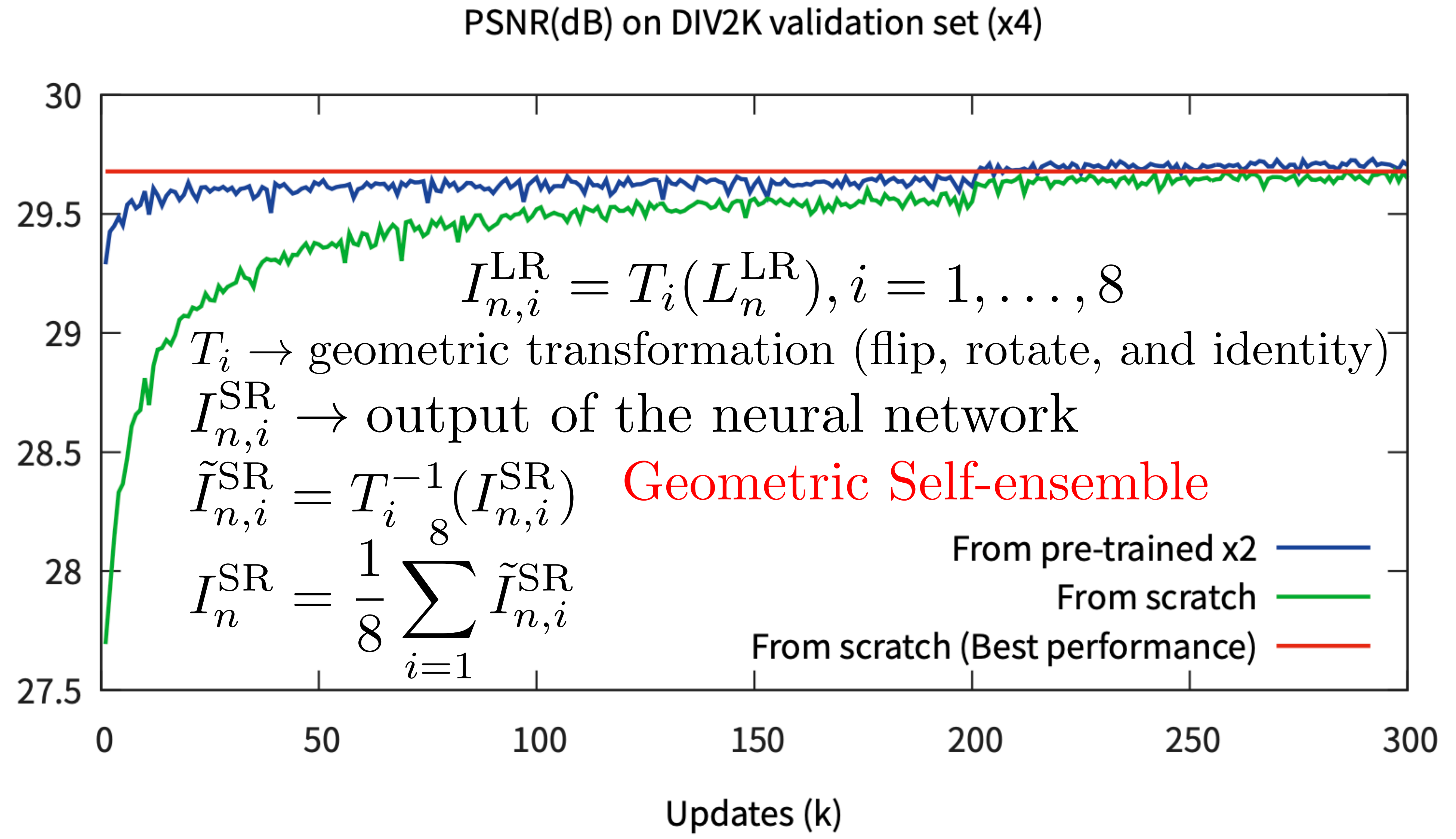


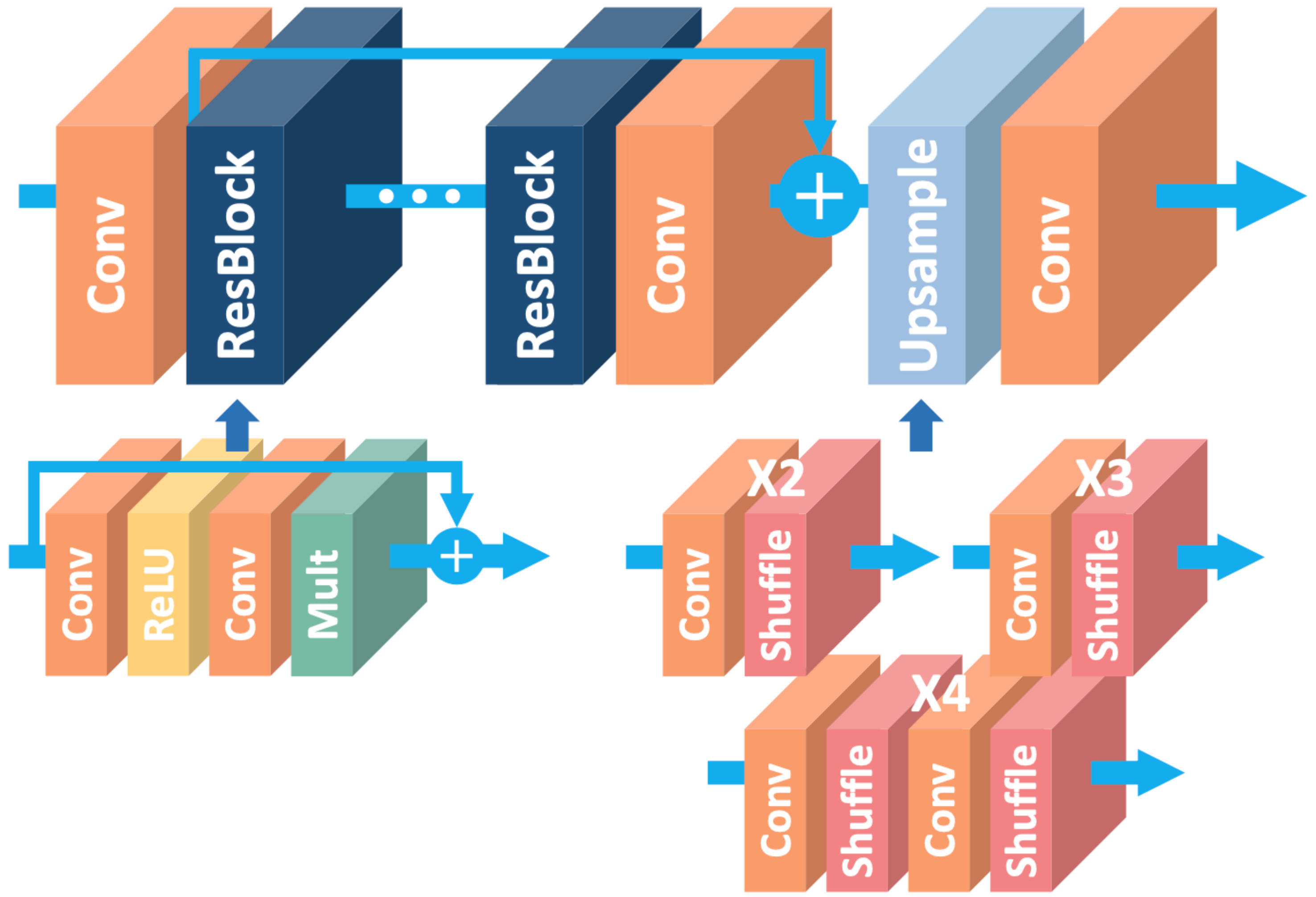
Zhang, Kai, et al. "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising." *IEEE Transactions on Image Processing* 26.7 (2017): 3142-3155.

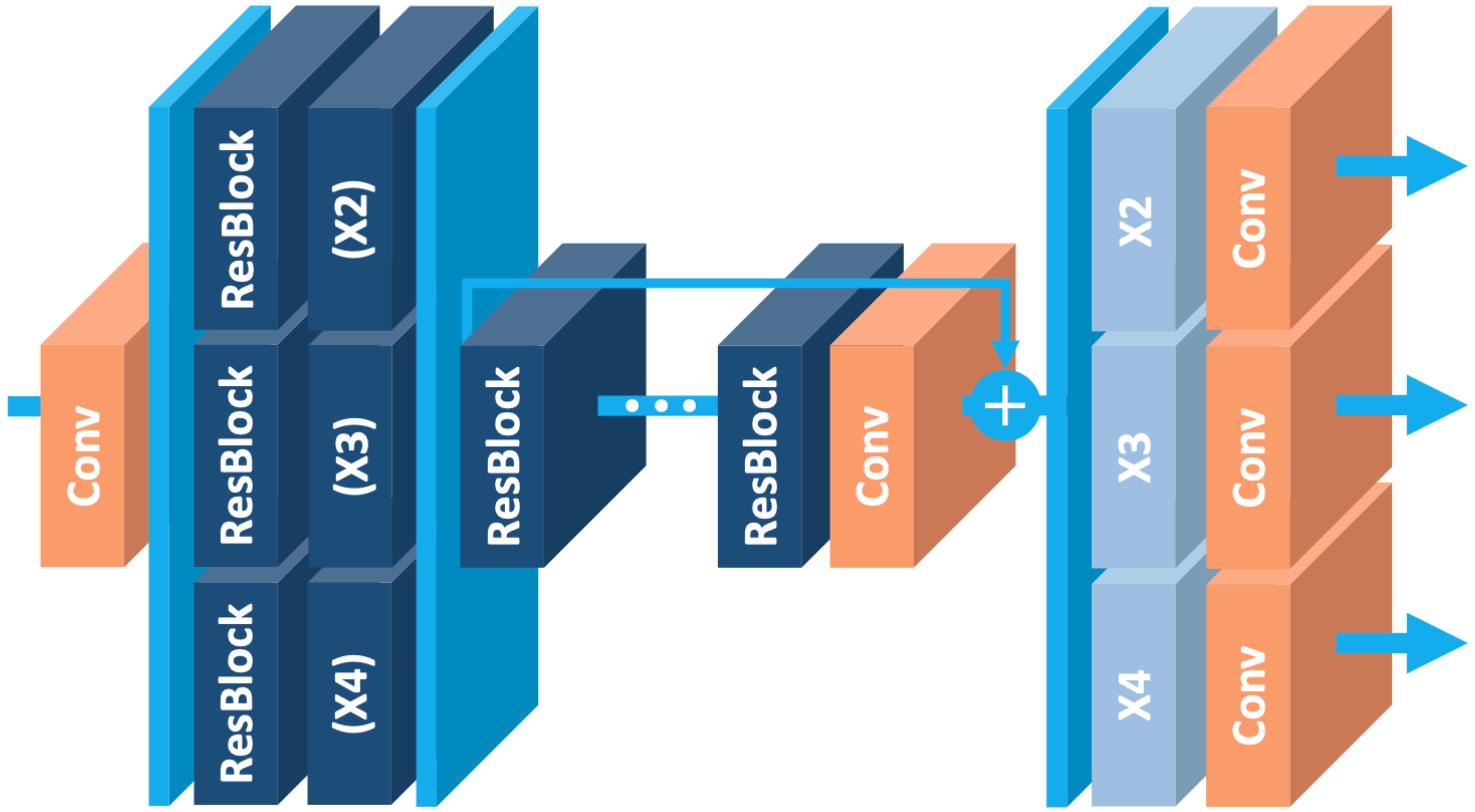
Enhanced Deep Residual Networks for Single Image Super-Resolution











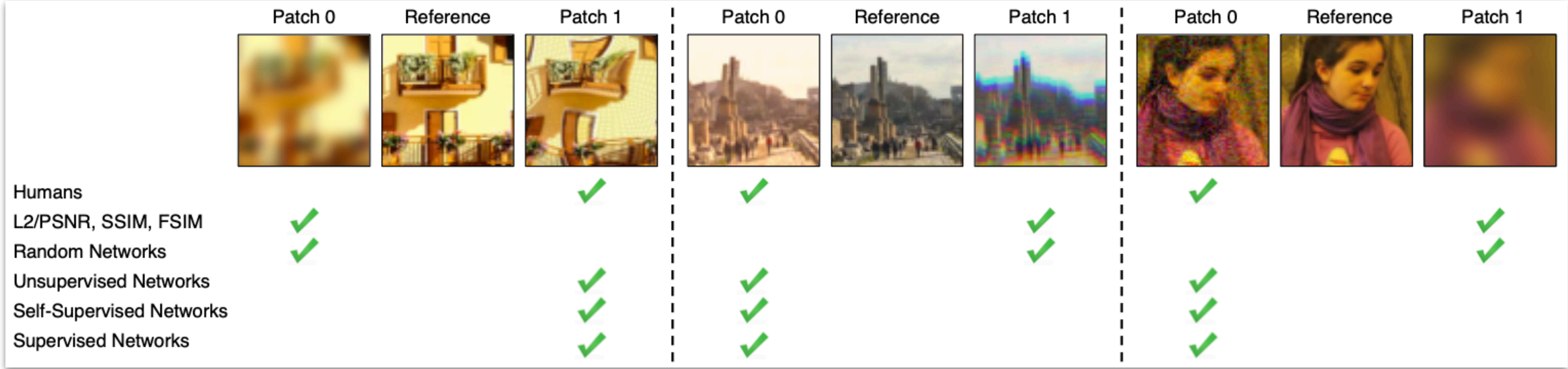
Options	SRResNet [14] (reproduced)	Baseline (Single / Multi)	EDSR	MDSR
# Residual blocks	16	16	32	80
# Filters	64	64	256	64
# Parameters	1.5M	1.5M / 3.2M	43M	8.0M
Residual scaling	-	-	0.1	-
Use BN	Yes	No	No	No
Loss function	L2	L1	L1	L1

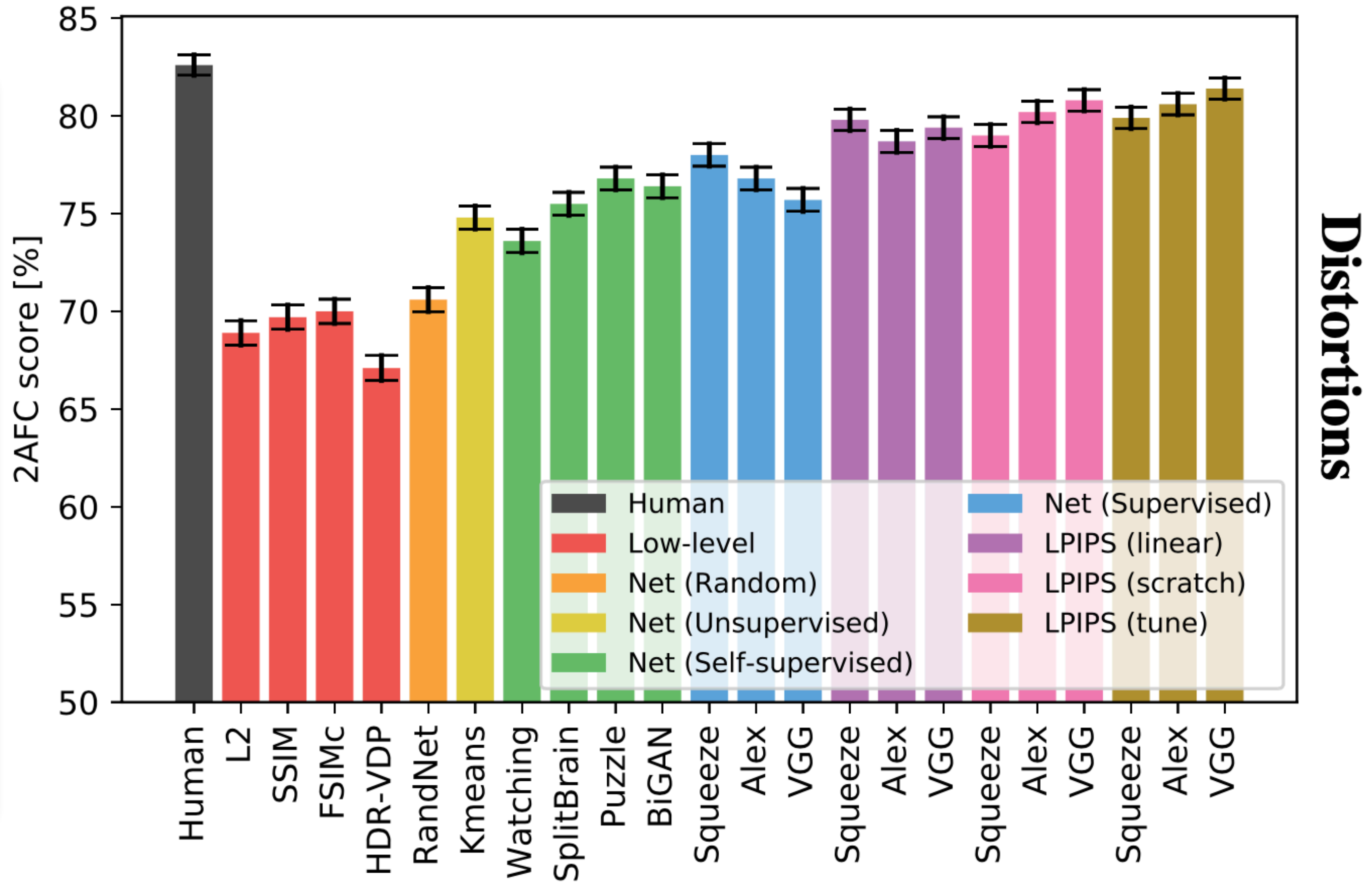
Lim, Bee, et al. "Enhanced deep residual networks for single image super-resolution." Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2017.

The Unreasonable Effectiveness of Deep Features as a Perceptual Metric

Boulder

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?

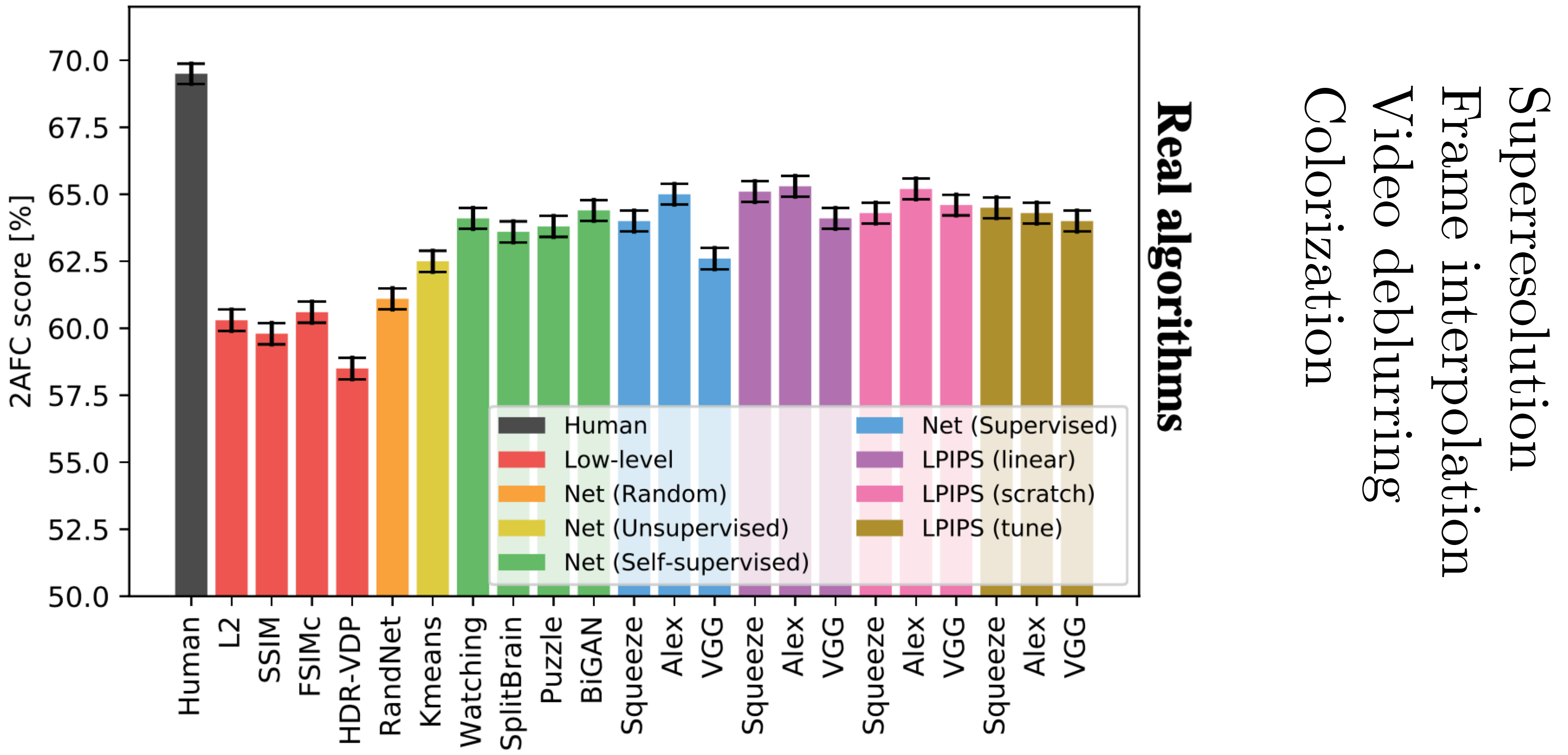
$x \rightarrow$ randomly selected image patch

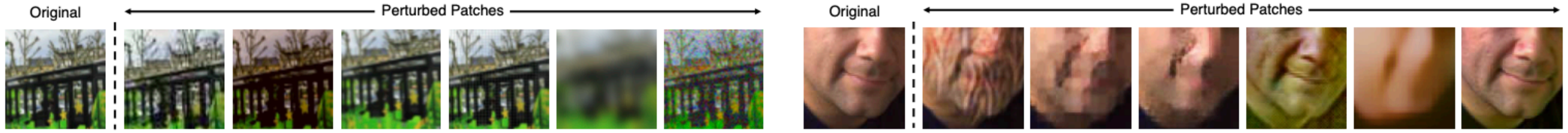
$x_0, x_1 \rightarrow$ distorted patches

$h \in \{0, 1\} \rightarrow$ 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**

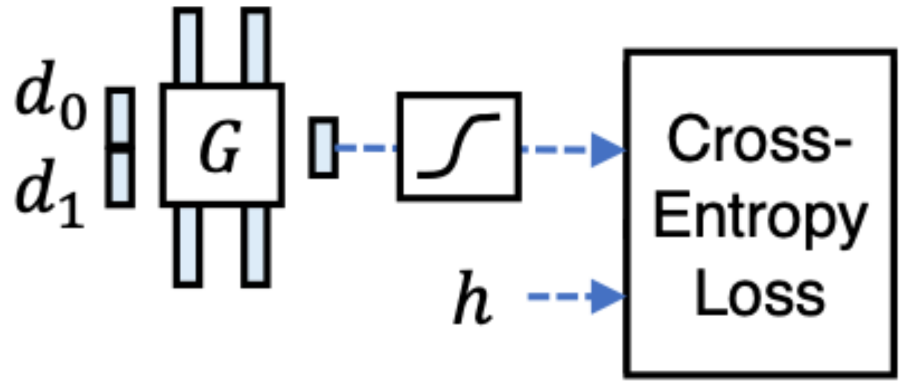
Learned Perceptual Image Patch Similarity (LPIPS) metric

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

$$d(x, x_0) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} ||w_l \odot (\hat{y}^l_{hw} - \hat{y}^l_{0hw})||^2_2$$

$\hat{y}^\ell, \hat{y}^\ell_0 \in \mathbb{R}^{H_\ell \times W_\ell \times C_\ell}$ for layer ℓ $\quad w^\ell \in \mathbb{R}^C \rightarrow$ scaling factor

$||\hat{y}^\ell|| = ||\hat{y}^\ell_0|| = 1 \rightarrow$ unit normalized in the channel dimension



Predicting Perceptual Judgement

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.

Questions?