Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network

Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, Wenzhe Shi Twitter

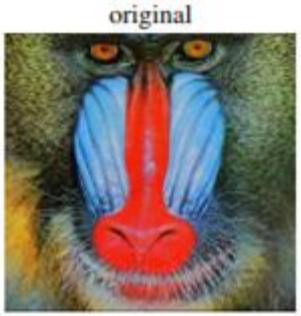
Introduction

- “How do we recover the finer texture details when we super-resolve at large upscaling factors?”
- Enhance the resolution of an imaging system
- Super-Resolution (SR): LR -> HR
- Aim: Increase facial resolution with a given upscaling factor and add fine texture details during the process

Image Super Resolution

Applications

- Satellite Imaging
- Medical Imaging
- Media Content
- Face Recognition
- Surveillance
- And many more...



Introduction (Conti…)

- Optimization target (supervised SR algo): Minimization of the mean squared error (MSE) between the recovered HR image and the ground truth
  - Minimizing MSE maximizes the peak signal-to-noise ratio (PSNR)
  - High PSNR => High Quality
- Ability to capture perceptually relevant differences, (high texture detail), is **very limited**
  - As they are defined based on **pixel-wise** image differences
  - Illustrated in Figure 2
  - Highest PSNR does not necessarily reflect the perceptually better SR result
  - Perceptual difference between the super-resolved and original image
    - Means that the recovered image is not photorealistic (Ferwerda)

Introduction (Conti...)

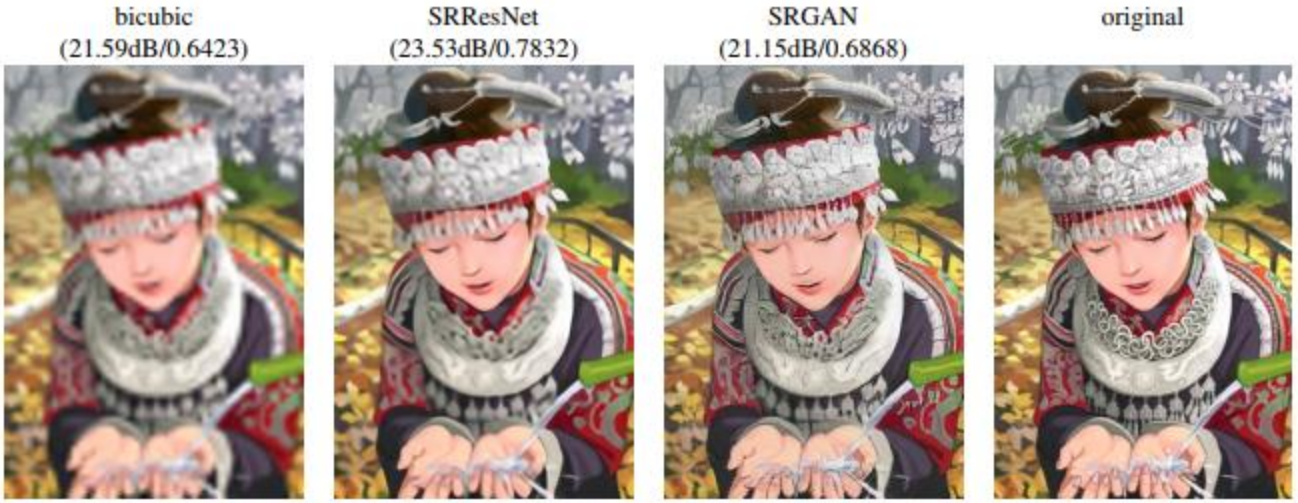


Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

Introduction (Conti...)

- Define a **super-resolution generative adversarial network (SRGAN)** for which we employ a **deep residual network (ResNet)** with skip-connection and diverge from MSE as the sole optimization target.
- Different from previous works, we define a novel perceptual loss using high-level feature maps of the VGG network combined with a discriminator that encourages solutions perceptually hard to distinguish from the HR reference images.
- Example photo-realistic image: Super resolved with a 4× upscaling factor is shown in Figure 1.

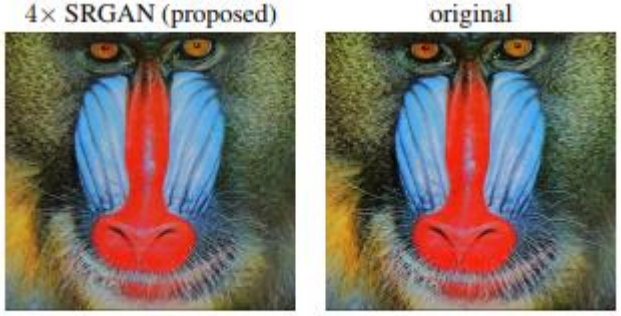


Figure 1: Super-resolved image (left) is almost indistinguishable from original (right). [4× upscaling]

Related Work

1. Image super-resolution
    1. Prediction-based methods
    2. Filtering approaches (e.g. linear, bicubic or Lanczos): very fast | over simplify
    3. Powerful approaches: establish a complex mapping between low and high-resolution image information - rely on training data
    4. Patch redundancies across scales within the image to drive the SR (self-similarity) - self dictionaries are extended by further allowing for small transformations and shape variations
    5. A convolutional sparse coding approach - improves consistency - processing the whole image rather than overlapping patches
    6. Combined edge-directed SR algorithm with learning-based detail synthesis
    7. Structure-aware matching criterion for alignment - retrieve correlating HR images with similar content from the web - for landmark images
    8. Neighborhood embedding approaches - upsample a LR image patch by finding similar LR training patches in a low dimensional manifold and combining their corresponding HR patches for reconstruction - neighborhood approaches tend to overfit
    9. Regression
    10. CNN

Related Work

Image super-resolution (CNN)

- Encode a sparse representation prior into their feed-forward network architecture based on the learned iterative shrinkage and thresholding algorithm (LISTA)
- Bicubic interpolation: Upscale an input image and trained a *three layer deep* fully convolutional network end-to-end to achieve state of-the-art SR performance
- Enabling the network to learn the upscaling filters directly: increased accuracy and speed
- Deeply-recursive convolutional network (DRCN) - long-range pixel dependencies while keeping the number of model parameters small
- **The works by Johnson et al. and Bruna et al., who rely on a loss function closer to perceptual similarity to recover visually more convincing HR images**

Design of CNN

"Deeper network architectures can be difficult to train but have the potential to substantially increase the network's accuracy as they allow modeling mappings of very high complexity"

- Batch-normalization
- recursive CNN
- residual blocks
- Skip connections

Learning upscaling filters is improvement from bicubic interpolation employed to upscale the LR observation before feeding the image to the CNN

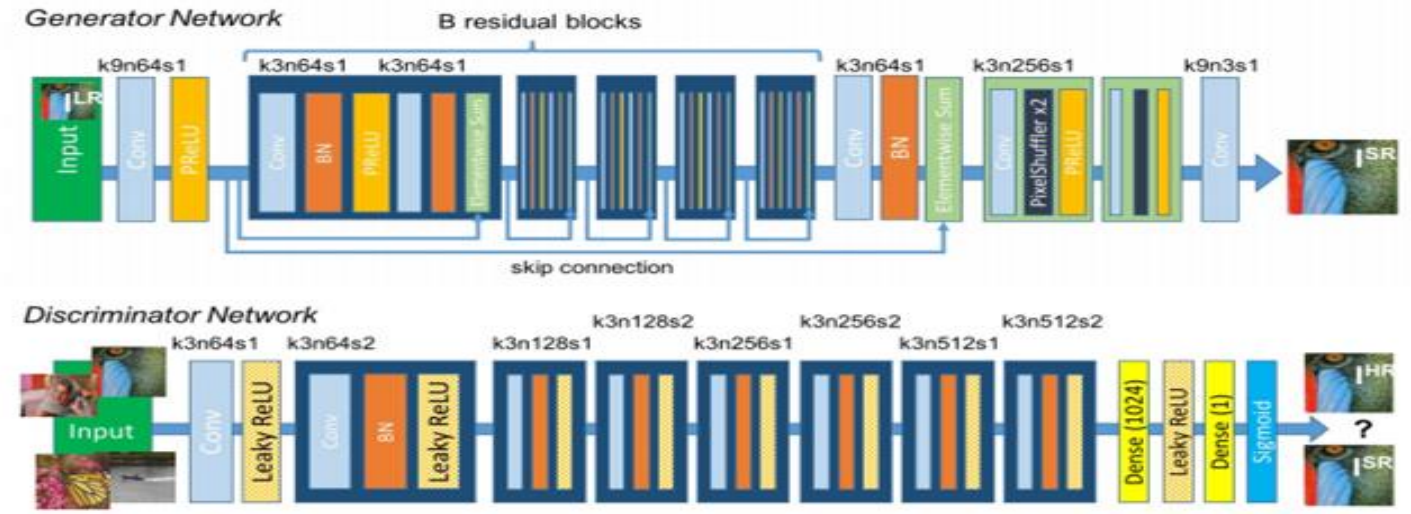
SRResNet

SRResNet

- Deep residual CNN for image SR
  - Trained on ImageNet (350k images)
  - Optimized for MSE matrix

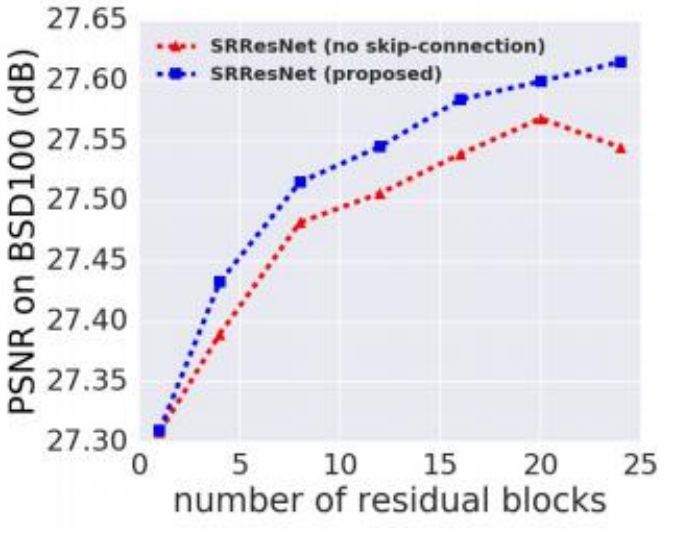
$$l_{MSE}^{SR} = \frac{1}{r^2WH}\sum_{x=1}^{rW}\sum_{y=1}^{rH}(I_{x,y}^{HR} - G_{\theta_G}(I^{LR})_{x,y})^2$$

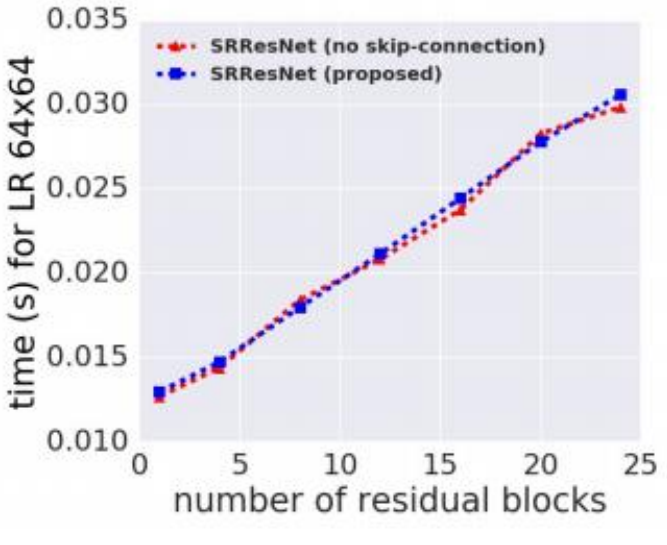
Architecture



- LR input images to [0, 1]
- HR images to [−1, 1].
- Intensity range [−1, 1]
- VGG feature maps rescaled by a factor of 1/12.75
  - To obtain VGG losses of a scale that is comparable to the MSE loss
- Use two convolutional layers with small 3×3 kernels and 64 feature maps followed by batch-normalization layers and ParametricReLU as activation function

Influence of Neural Depth





Example

original

bicubic

SRResNet

SRGAN

Loss Functions

- Problem of minimizing MSE
  - Multiple potential solutions with high texture details are averaged to create a smooth reconstruction
  - illustrate the problem of minimizing MSE in figure where multiple potential solutions with high texture details are averaged to create a smooth reconstruction
- Tackled this problem by employing generative adversarial networks (GANs)

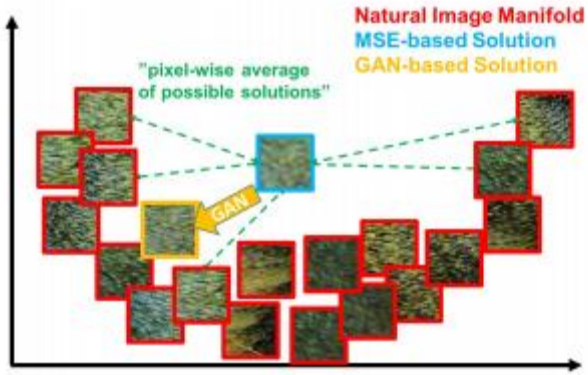


Figure 3: Illustration of patches from the natural image manifold (red) and super-resolved patches obtained with MSE (blue) and GAN (orange). The MSE-based solution appears overly smooth due to the pixel-wise average of possible solutions in the pixel space, while GAN drives the reconstruction towards the natural image manifold producing perceptually more convincing solutions.

Limitations of MSE

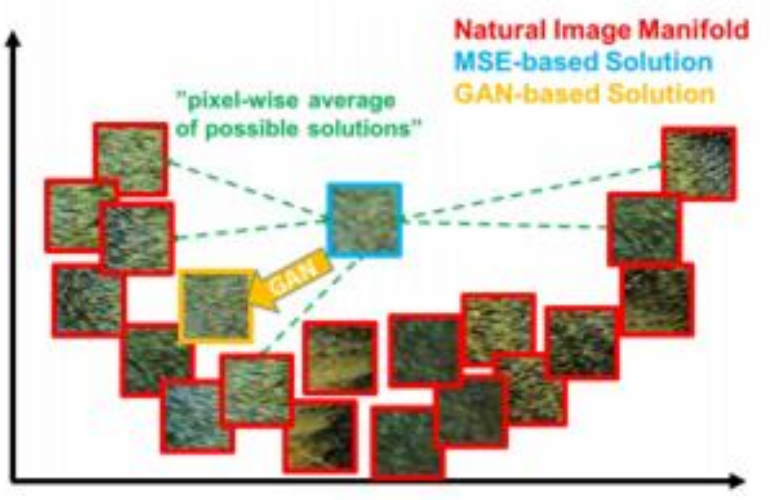
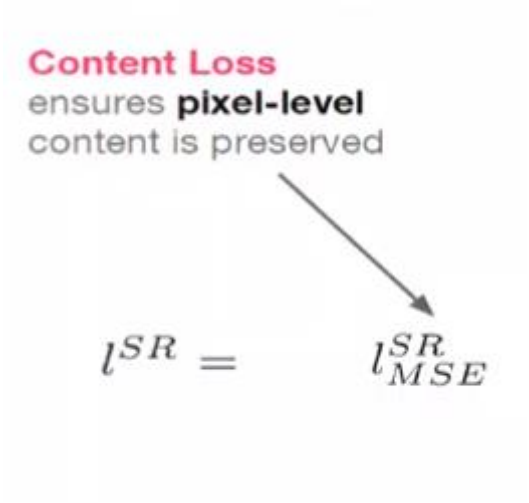
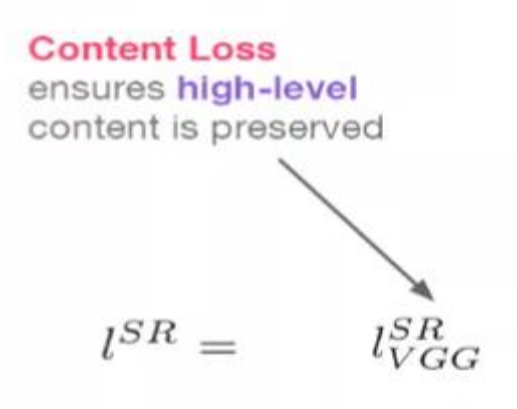


Figure 3: Illustration of patches from the natural image manifold (red) and super-resolved patches obtained with MSE (blue) and GAN (orange). The MSE-based solution appears overly smooth due to the pixel-wise average of possible solutions in the pixel space, while GAN drives the reconstruction towards the natural image manifold producing perceptually more convincing solutions.

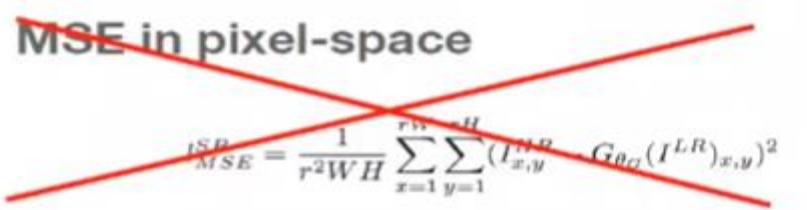
From MSE to Perceptual Loss



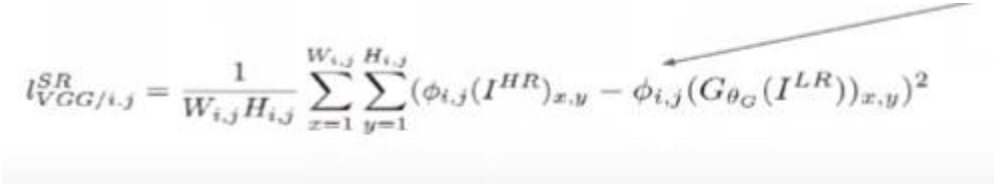


From MSE to Perceptual Loss

- MSE in pixel space



- MSE in VGG Feature Space

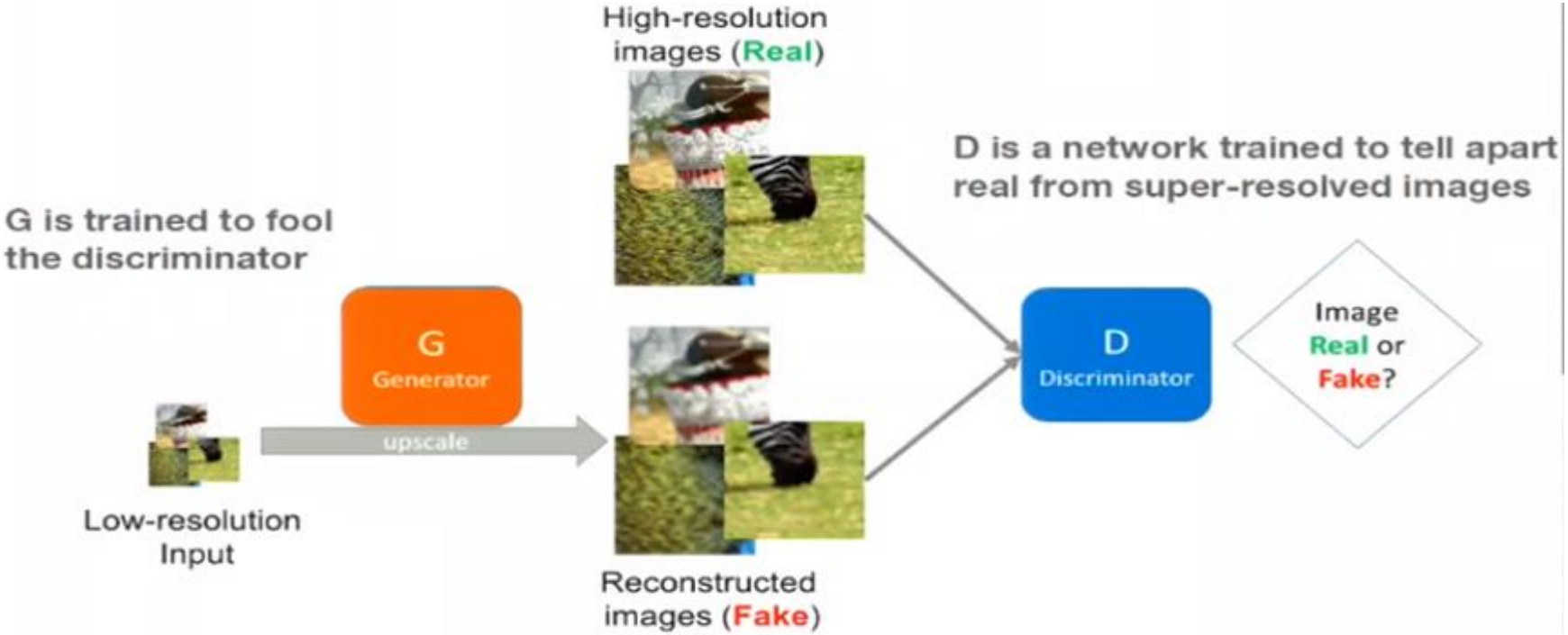


Intuition

“While regression to mean problem in pixel space gives average like solution, finding the mean solution in feature space **does not necessarily** penalize the reconstruction fine texture details in pixel space**”**

Perceptual Loss Components

GAN


High-resolution images (Real)
D is a network trained to tell apart real from super-resolved images
G is trained to fool the discriminator
G
Generator
upscale
D
Discriminator
Image Real or Fake?
Low-resolution Input
Reconstructed images (Fake)

Adversarial Loss

Train a generative adversarial network (GAN)

- Discriminator discriminates real from super-resolved images
- **Generate solutions that lie on the natural image manifold**

$$\min_{\theta_G} \max_{\theta_D} \mathbb{E}_{I^{HR} \sim p_{\text{train}}(I^{HR})}[\log D_{\theta_D}(I^{HR})] + \mathbb{E}_{I^{LR} \sim p_G(I^{LR})}[\log(1 - D_{\theta_D}(G_{\theta_G}(I^{LR}))]$$

$$l^{SR}_{Gen} = \sum_{n=1}^{N} -\log D_{\theta_D}(G_{\theta_G}(I^{LR}))$$

$$l^{SR} = l^{SR}_{VGG} + 10^{-3} l^{SR}_{Gen}$$

Conclusion and Results

Limitations of PSNR/SSIM

Problem with evaluating this approach lies in limitation of commonly used evaluation methods like PSNR to actually capture the perceptual quality of the reconstruction

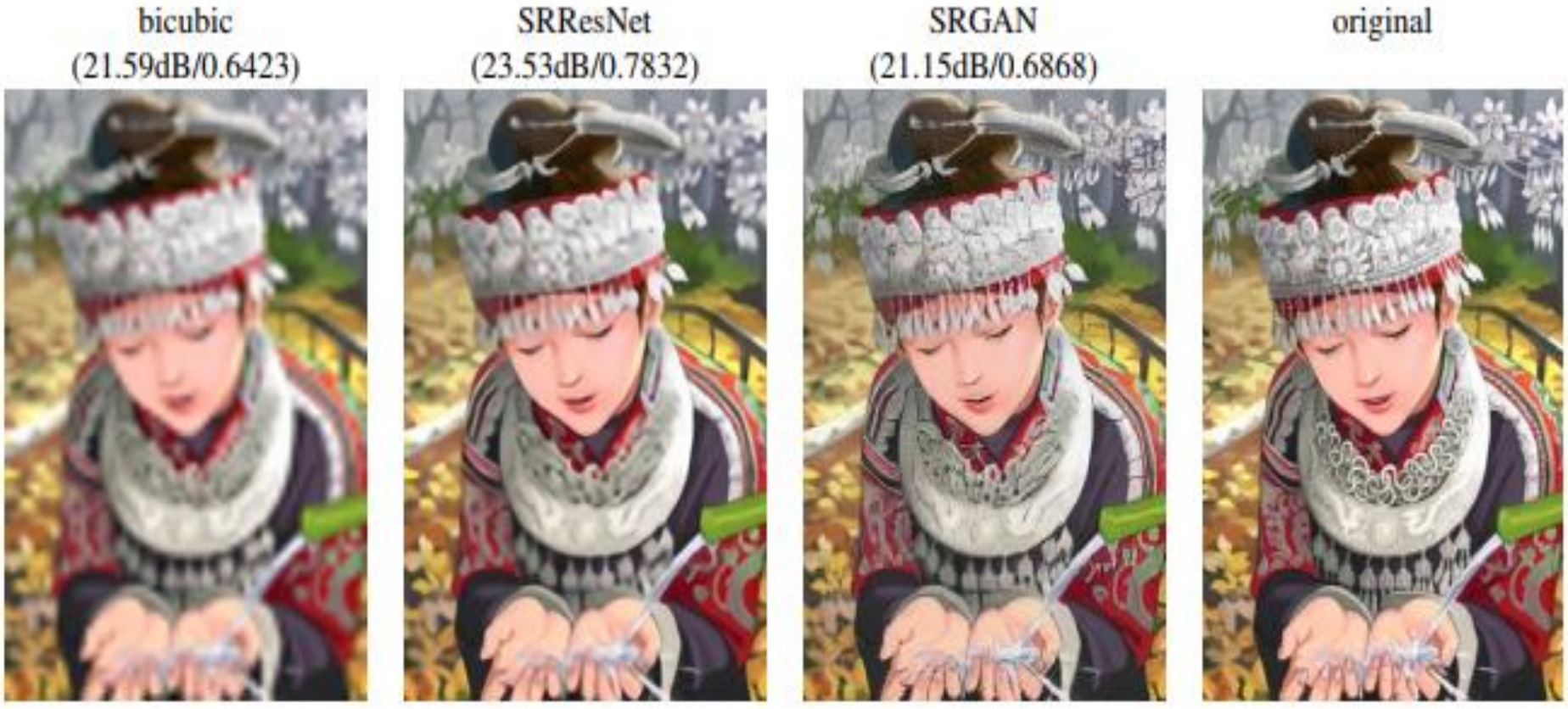


Figure 2: From left to right: bicubic interpolation, deep residual network optimized for MSE, deep residual generative adversarial network optimized for a loss more sensitive to human perception, original HR image. Corresponding PSNR and SSIM are shown in brackets. [4× upscaling]

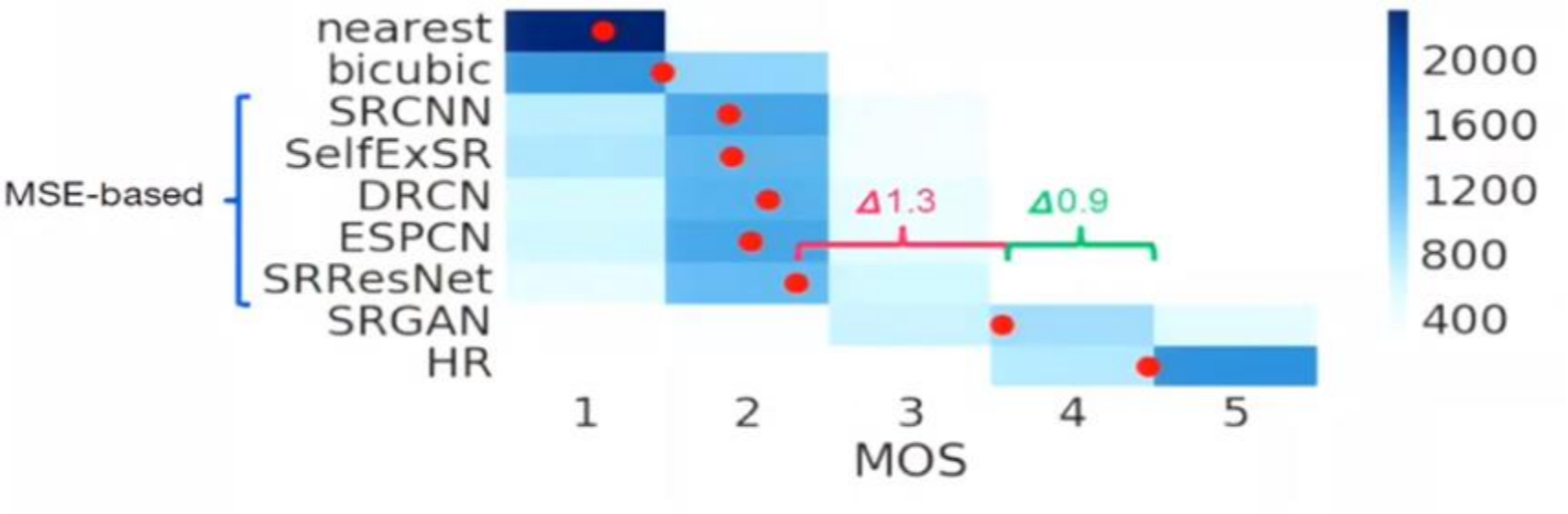
Mean-Opinion-Score (MOS) Testing

PSNR and SSIM fail to assess perceptual quality

26 human raters

- Give scores 1 (bad) to 5 (excellent)
- Each rater rated more than 1000 images

MOS Test Results (4x): Outperforms



[SRCNN] **Dong, et al.** Learning a deep convolutional network for image super-resolution. **ECCV 2014.**
[SelfExSR] **Huang, et al. Shi, et al.**, "Single image super-resolution from transformed self-exemplars", **CVPR 2015**
[DRCN] **Kim, et al.**, "Deeply-recursive convolutional network for image super-resolution", **CVPR 2016**
[ESPCN] **Shi, et al.**, "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network",

Results: Comparison to state of art methods

Table 1: Performance of different loss functions for SRResNet and the adversarial networks on Set5 and Set14 benchmark data. MOS score significantly higher ($p < 0.05$) than with other losses in that category*. [4× upscaling]

	SRResNet-		SRGAN-		
Set5	MSE	VGG22	MSE	VGG22	VGG54
PSNR	32.05	30.51	30.64	29.84	29.40
SSIM	0.9019	0.8803	0.8701	0.8468	0.8472
MOS	3.37	3.46	3.77	3.78	3.58
Set14					
PSNR	28.49	27.19	26.92	26.44	26.02
SSIM	0.8184	0.7807	0.7611	0.7518	0.7397
MOS	2.98	3.15*	3.43	3.57	3.72*

Table 2: Comparison of NN, bicubic, SRCNN [9], SelfExSR [31], DRCN [34], ESPCN [48], **SRResNet**, **SRGAN**-VGG54 and the original HR on benchmark data. Highest measures (PSNR [dB], SSIM, MOS) in bold. [4× upscaling]

Set5	nearest	bicubic	SRCNN	SelfExSR	DRCN	ESPCN	**SRResNet**	**SRGAN**	HR
PSNR	26.26	28.43	30.07	30.33	31.52	30.76	**32.05**	29.40	∞
SSIM	0.7552	0.8211	0.8627	0.872	0.8938	0.8784	**0.9019**	0.8472	1
MOS	1.28	1.97	2.57	2.65	3.26	2.89	3.37	**3.58**	4.32
Set14									
PSNR	24.64	25.99	27.18	27.45	28.02	27.66	**28.49**	26.02	∞
SSIM	0.7100	0.7486	0.7861	0.7972	0.8074	0.8004	**0.8184**	0.7397	1
MOS	1.20	1.80	2.26	2.34	2.84	2.52	2.98	**3.72**	4.32
BSD100									
PSNR	25.02	25.94	26.68	26.83	27.21	27.02	**27.58**	25.16	∞
SSIM	0.6606	0.6935	0.7291	0.7387	0.7493	0.7442	**0.7620**	0.6688	1
MOS	1.11	1.47	1.87	1.89	2.12	2.01	2.29	**3.56**	4.46

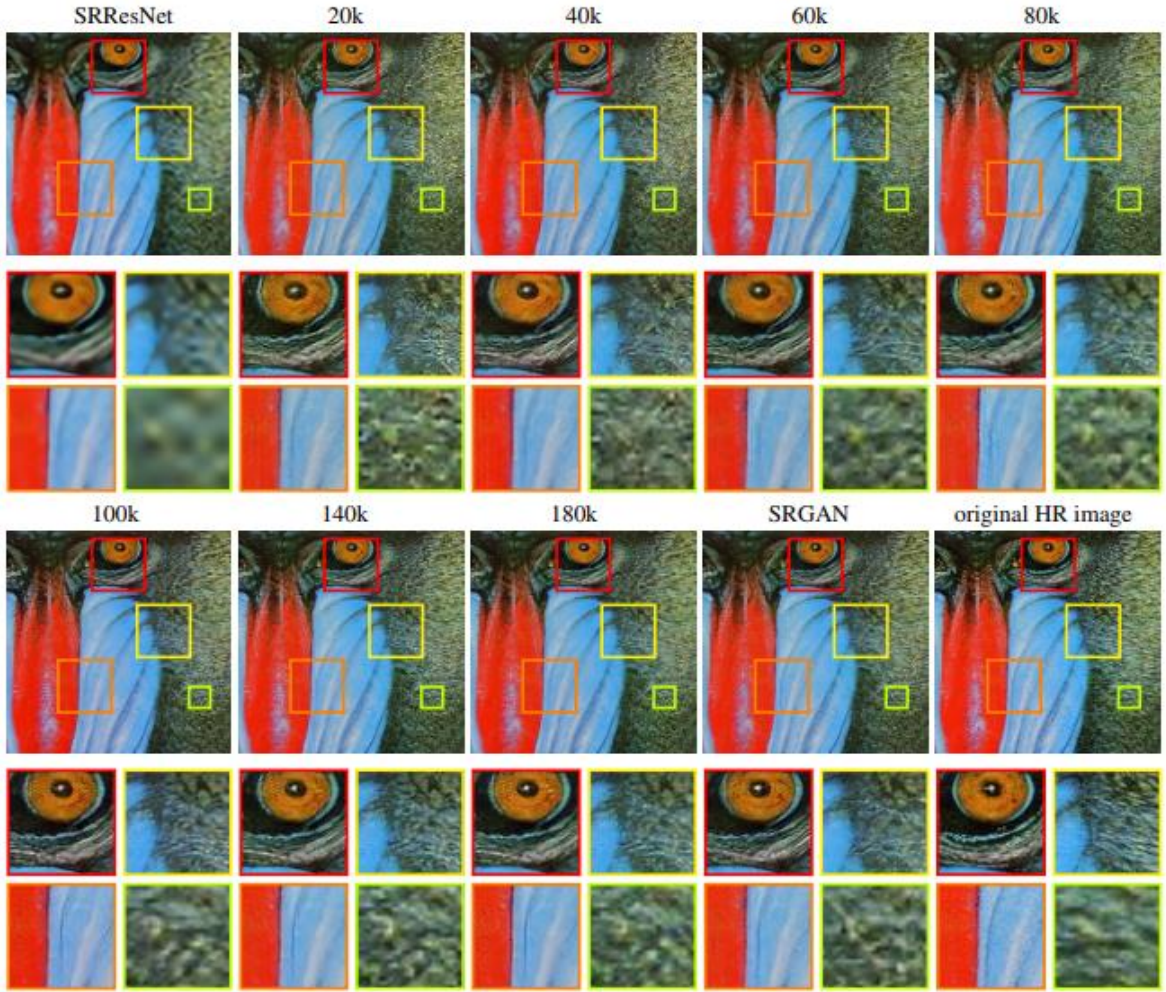


Figure 8: Evolution of SRGAN generator network during training progress. Note: Generator initialized with SRResNet weights; learning rate set to 10^{-4} for first 100k iterations, then reduced to 10^{-5} for another 100k iterations. [4× upscaling]

Thank you