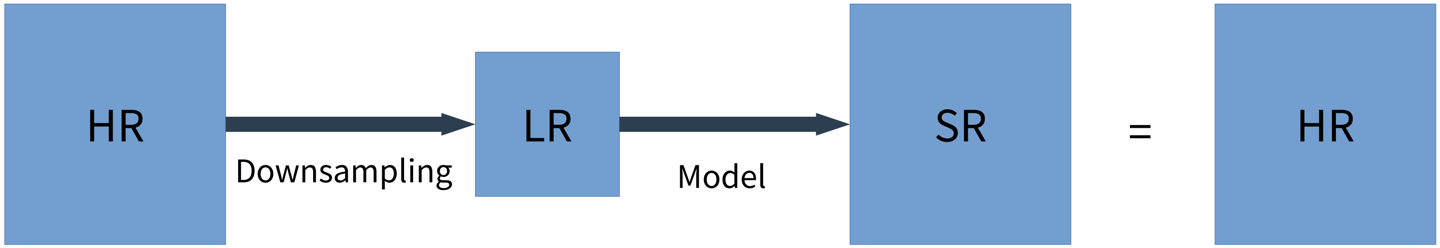
Multi-Scale Super Resolution With Blind De-noising Using Residual Learning For Digital Art

Student: Santiago Mazagatos Pérez
Supervisor: Sergio Escalera Guerrero

Super Resolution

- Ideally: increasing the size of an image without losing fidelity.
- Restore a hypothetical High Resolution image from a Low Resolution image.
- Given an actual LR and HR pair, the restored Super Resolution image given by LR should be identical to the HR image.



Classic SR Methods: Dictionary methods

- Create a database of low resolution patches and high resolution patches.
- Input is processed and defined in terms of the low resolution patches.
- The coefficients used to define the input using the low resolution patches are applied to their corresponding high resolution patches and the image is reconstructed.

New Approach: SRCNN

- Using example LR and HR pairs, let the network learn the desired upsampling function on its own.
- Much less handcrafted than dictionary approaches.
- Most of current SR work is based on CNNs.

SRCNN based application: Waifu2x

- Trained with digital art (mainly anime).
- Input is upscaled before processing and then deblurred by the network, making it capable of using any scale (with diminishing results as the scale increases and thus the blurriness of the input increases).
- Requires different models for different noise levels.

Residual Learning

- Used to mitigate the vanishing/exploding gradient problem.
- Skip connections are added to the network to allow outputs from earlier layers to affect deeper layers.
- Allows for very deep architectures and thus highly complex models.

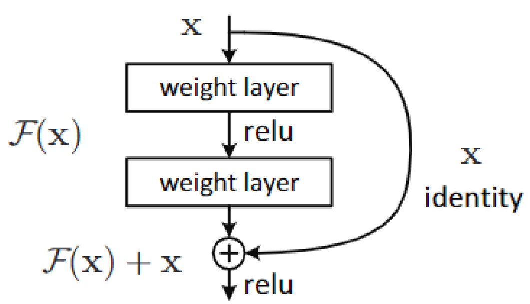
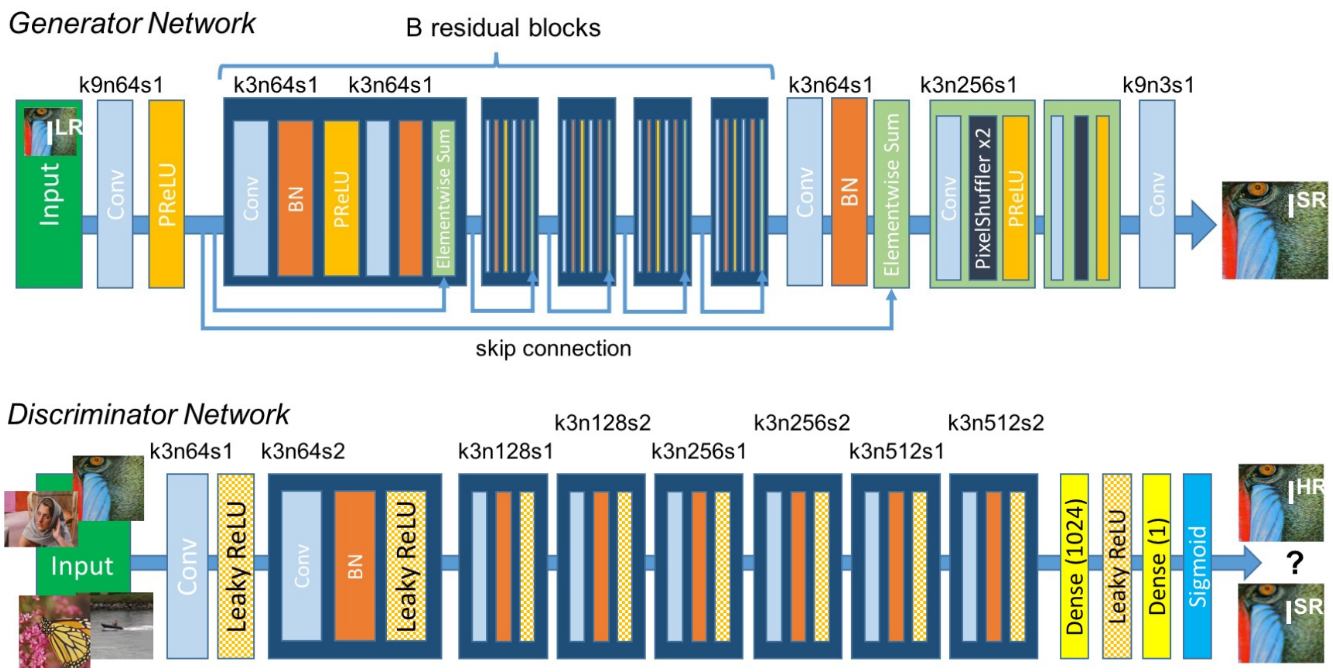


Figure 2. Residual learning: a building block.

[K. He, 2016] Deep Residual Learning for Image Recognition

SRResNet and SRGAN

- SRResNet: modified ResNet model for Super Resolution tasks.
- SRGAN: Generative Adversarial Network using SRResNet as a generator, and a discriminator to determine whether or not a generated SR image looks “natural”.



[C. Ledig, 2017] Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network

Starting point: EDSR

- Modification of SRResNet.
- Batch Normalization layers are removed.
- Thanks to reduced memory and computational requirements, the network can be deeper, and convolutional layers can have a higher number of filters.

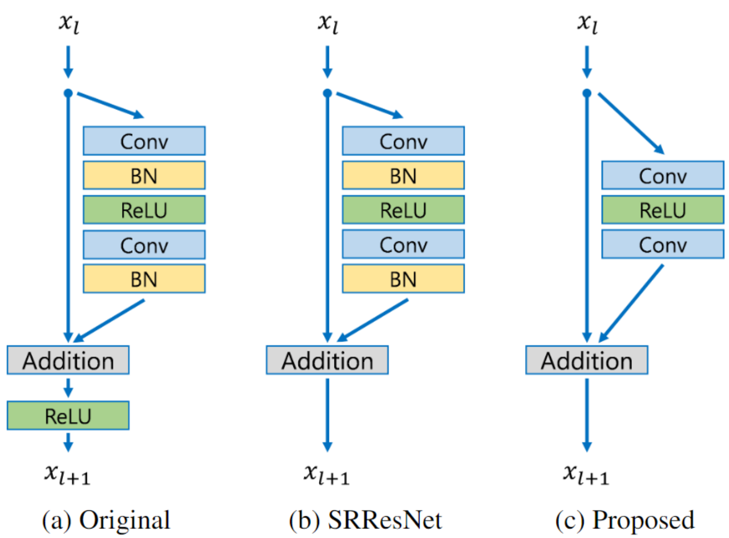


Figure 2: Comparison of residual blocks in original ResNet, SRResNet, and ours.

[B. Lim, 2017] Enhanced Deep Residual Networks for Single Image Super-Resolution

Starting point: CinCGAN

- Model that uses two CycleGANs along with a SR network (EDSR in the case of the paper).
- Unsupervised super-resolution (no HR-LR pairs needed in training).
- Unsupervised denoising.

Ground Truth	Bicubic	EDSR [17]	BM3D+EDSR	**CinCGAN**
PSNR/SSIM	29.42/0.82	28.95/0.76	30.94/0.91	31.01/0.92

[Y. Yuan, 2018] Unsupervised Image Super-Resolution using Cycle-in-Cycle Generative Adversarial Networks

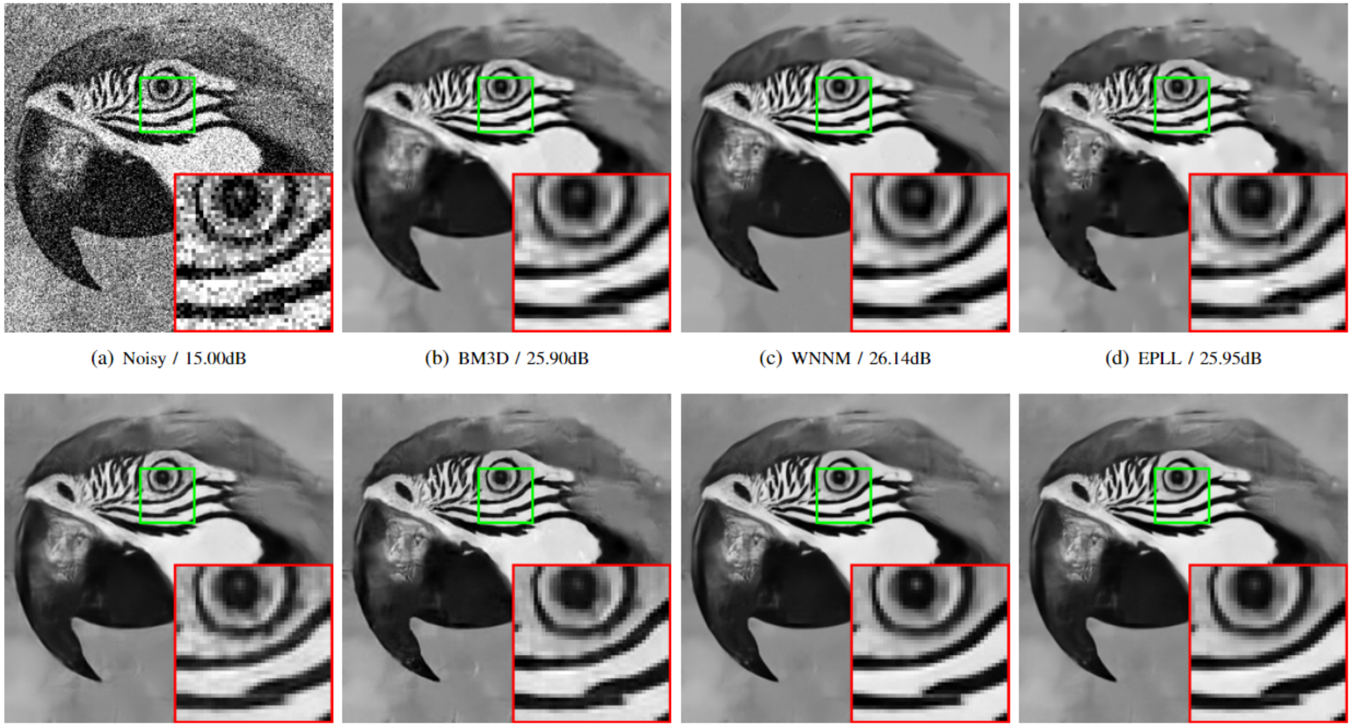
Denoising: BM3D

- BM3D is a method proposed in a 2007 paper based on collaborative filtering.
- The method processes an image in blocks by matching them with similar blocks and using collaborative filtering, in other words, estimating a clean block from all the similar blocks.
- Still popular despite its age, it's used as a reference in the CinCGAN paper when paired with EDSR for denoising tasks.

Denoising: DnCNN

An adaptation of the VGG network for denoising.

- Deep network
- Residual learning
- Blind denoising
- Deblurring as a means of SR



(a) Noisy / 15.00dB (b) BM3D / 25.90dB (c) WNNM / 26.14dB (d) EPLL / 25.95dB

(e) MLP / 26.12dB (f) TNRD / 26.16dB (g) DnCNN-S / 26.48dB (h) DnCNN-B / 26.48dB

Fig. 5. Denoising results of the image “parrot” with noise level 50.

[K. Zhang, 2016] Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising

Denoising: DnCNN

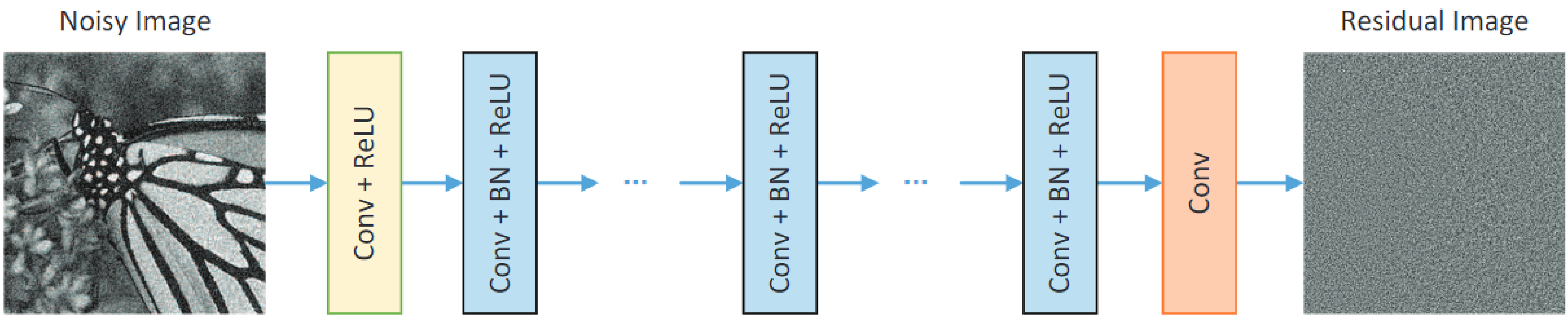


Fig. 1. The architecture of the proposed DnCNN network.

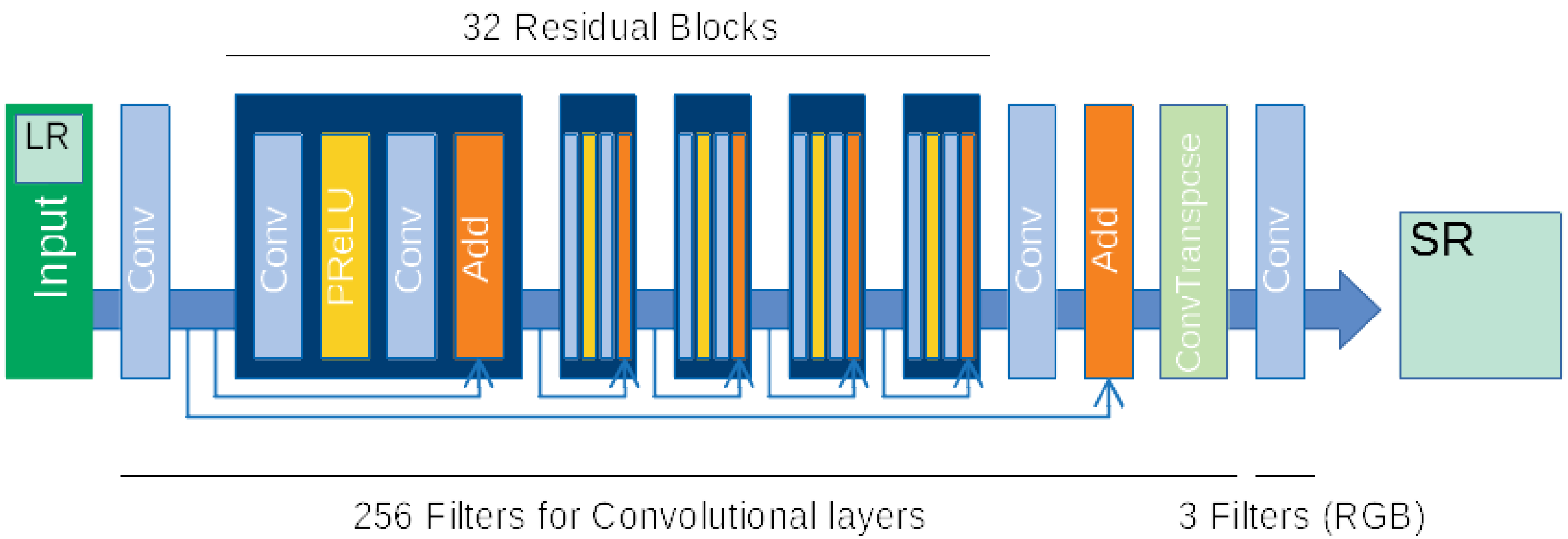
[K. Zhang, 2016] Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising

CNTK

- CNTK is an open source deep learning framework developed by Microsoft.
- It has Brainscript, Python and C# APIs
- Plenty of tutorials and examples available .

Implementation

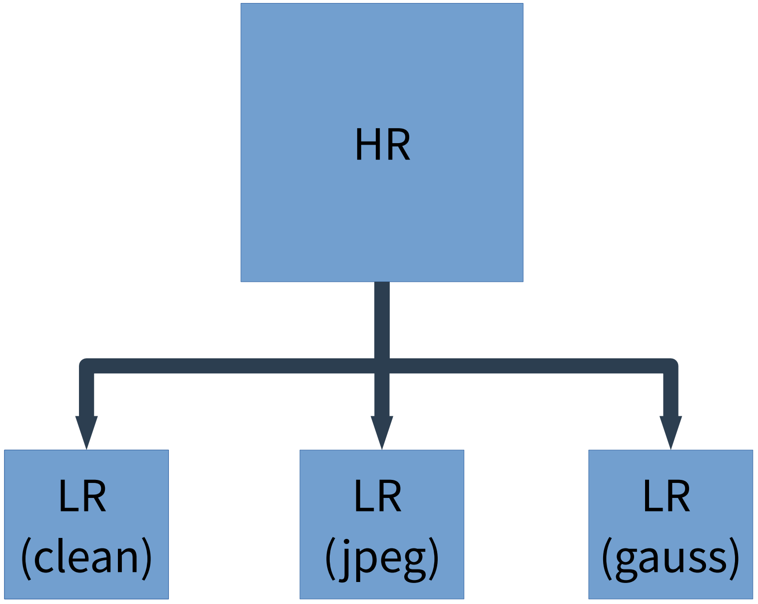
The implementation started by modifying a code example for SRResNet to include the modifications of EDSR, including the removal of Batch Normalization layers and the increase in filters of convolutional layers and number of residual blocks.



Implementation

Training is done with pairs of HR and LR images, for every pair created the LR patch will be either:

- Saved as a PNG (clean)
- Saved as a JPEG with random quality
- Saved as a PNG with random gaussian noise



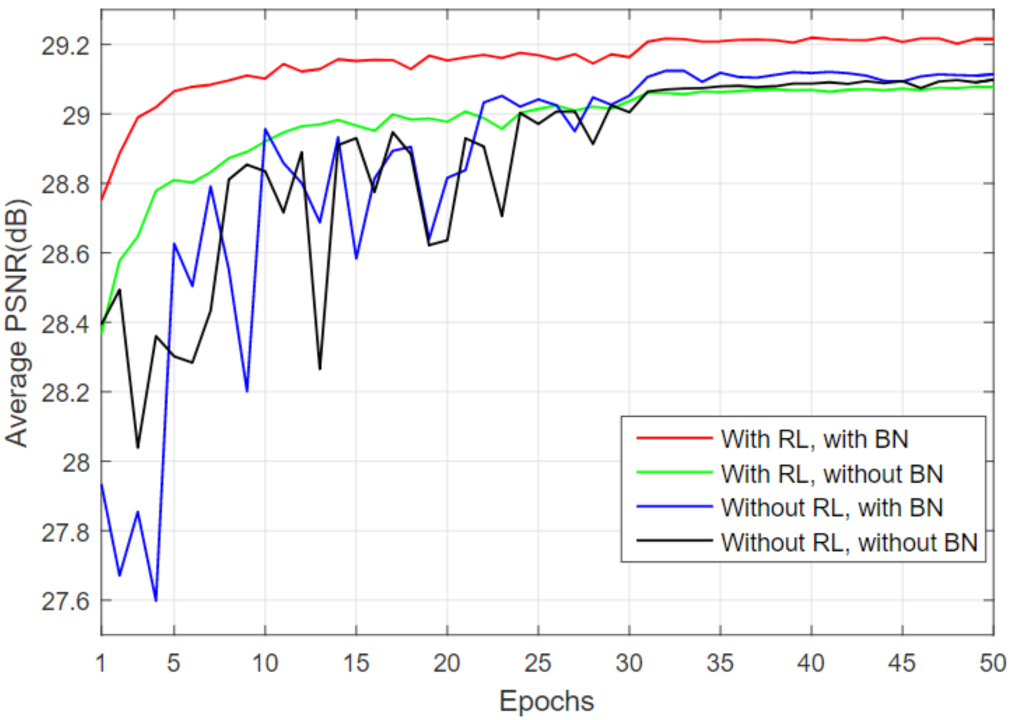
Considerations

- **EDSR: State of the art super resolution, no denoising.**
  - EDSR + BM3D denoising as “state of the art” (CinCGAN paper)
- **DnCNN: State of the art denoising, SRCNN style super resolution, outdated.**
- **Solution: EDSR super resolution with DnCNN style denoising to create a model beyond the current state of the art.**

Considerations

Differences between EDSR and DnCNN architecture-wise:

- EDSR doesn't use Batch Normalization, DnCNN does.
- EDSR uses L1 metric for better performance, DnCNN uses L2.
- DnCNN provides a different kind of output.

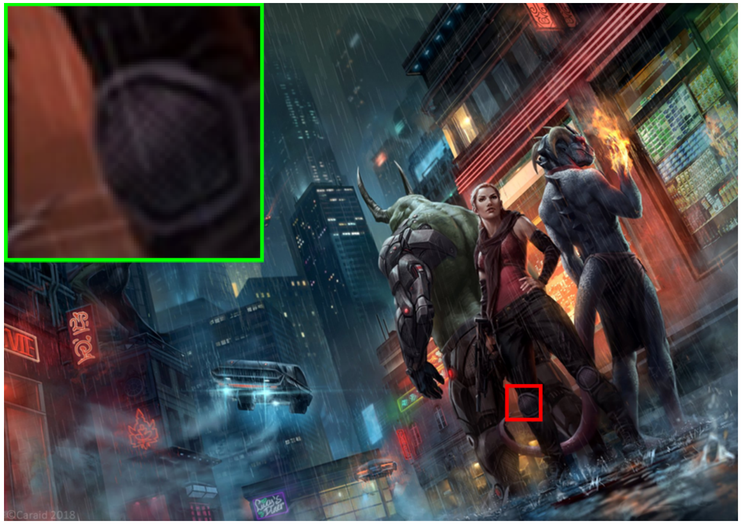


(b) Adam

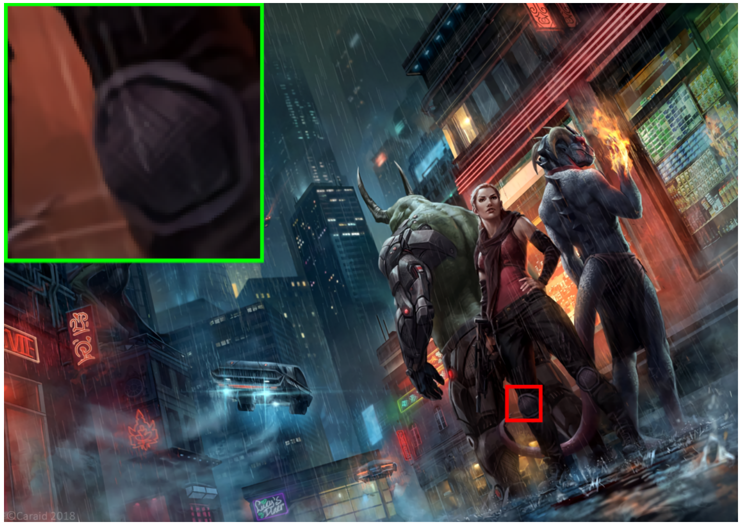
Results

HR

Bicubic



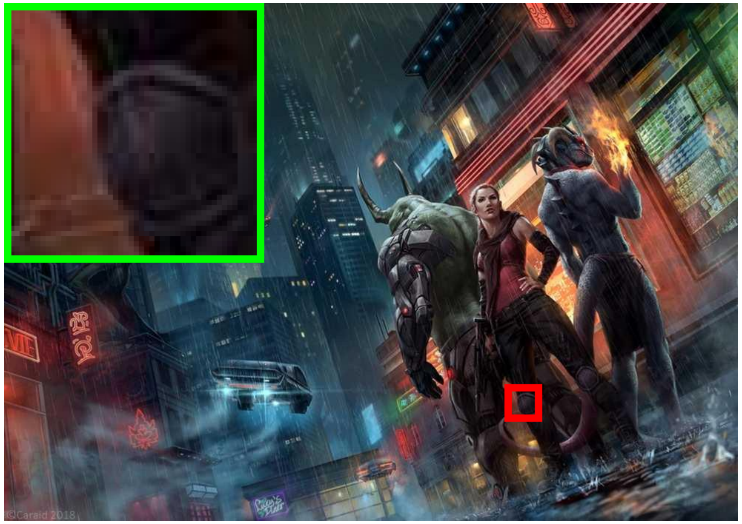
Mine



Waifu2x
(best result)

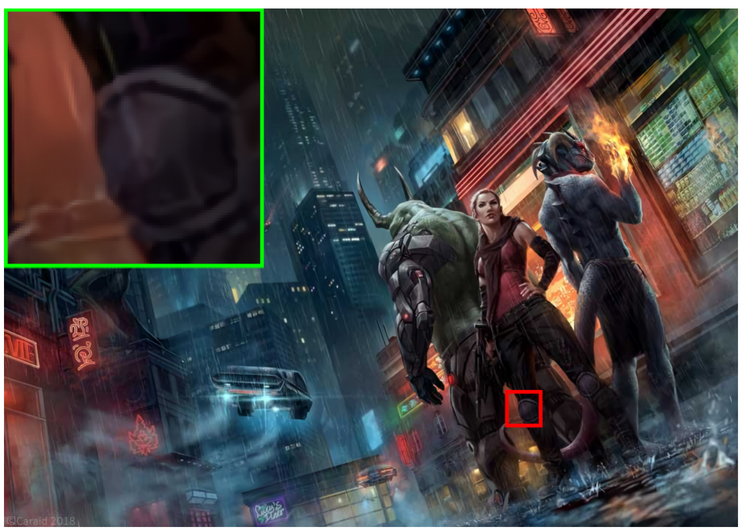
Results

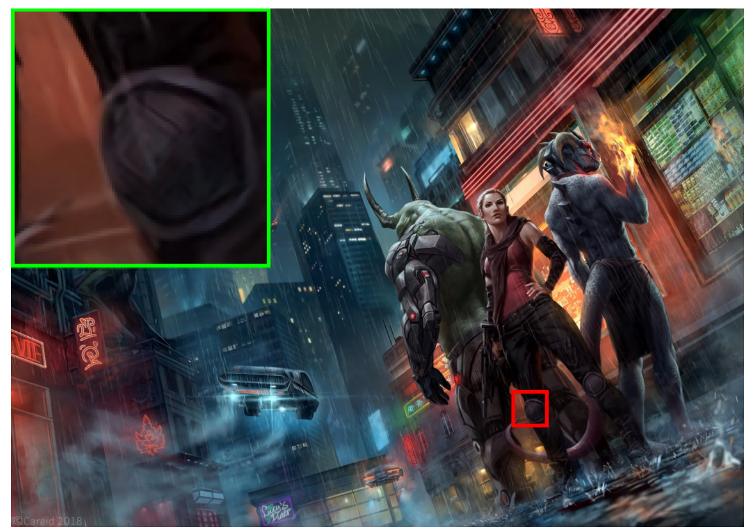
JPEG LR
Quality: 50



Gaussian LR
Sigma: 25

SR Result





SR Result

Results

	GAUSS_0	GAUSS_15	GAUSS_25	GAUSS_50	Average
EDSR	35.654	34.420	33.751	32.423	34.062
BICUBIC	33.468	26.475	23.640	18.774	25.589
Waifu2X (level 0)	34.854	25.066	21.715	16.359	24.499
Waifu2X (level 1)	34.222	25.146	21.966	16.744	24.520
Waifu2X (level 2)	33.739	25.185	22.152	17.493	24.642
Waifu2X (level 3)	32.656	27.014	24.529	19.467	25.916

	JPEG_100	JPEG_75	JPEG_50	JPEG_25	Average
EDSR	35.183	32.176	30.895	29.394	31.912
BICUBIC	30.696	31.013	29.886	28.657	30.063
Waifu2X (level 0)	34.726	31.775	30.182	28.543	31.306
Waifu2X (level 1)	34.090	32.093	30.672	28.950	31.451
Waifu2X (level 2)	33.563	31.898	30.880	29.443	31.446
Waifu2X (level 3)	32.613	31.832	30.923	29.507	31.218

	GAUSS_0	GAUSS_15	GAUSS_25	GAUSS_50	Average
EDSR	0.960	0.940	0.929	0.898	0.932
BICUBIC	0.938	0.750	0.582	0.316	0.647
Waifu2X (level 0)	0.948	0.647	0.435	0.193	0.556
Waifu2X (level 1)	0.938	0.647	0.448	0.207	0.560
Waifu2X (level 2)	0.924	0.629	0.448	0.238	0.560
Waifu2X (level 3)	0.899	0.856	0.742	0.410	0.727

	JPEG_100	JPEG_75	JPEG_50	JPEG_25	Average
EDSR	0.956	0.892	0.857	0.807	0.878
BICUBIC	0.861	0.869	0.832	0.785	0.837
Waifu2X (level 0)	0.948	0.887	0.843	0.785	0.866
Waifu2X (level 1)	0.939	0.892	0.854	0.798	0.871
Waifu2X (level 2)	0.927	0.882	0.855	0.811	0.869
Waifu2X (level 3)	0.899	0.880	0.856	0.813	0.862

Conclusions

- Great restoration of detail in complex artwork.
- Good quality denoising, especially with Gaussian noise.
- Performance comparable or superior to waifu2x, possible overfitting issue in low complexity art, room for more complexity (additional noise patterns)

Demo & Questions