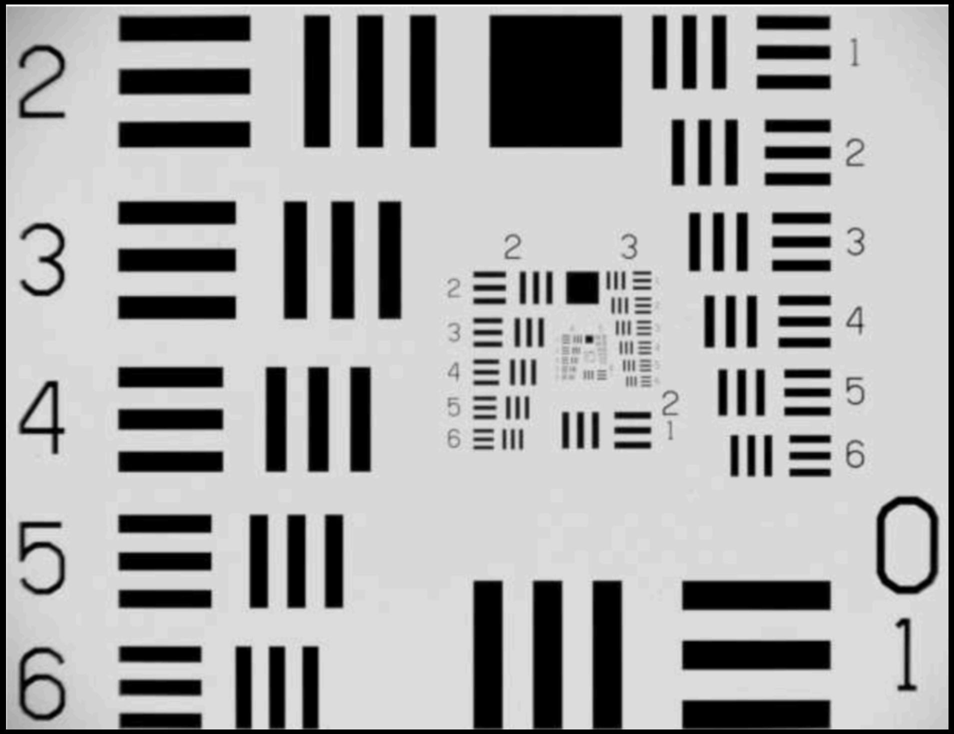
SuperResolution (SR)

Earl Wong

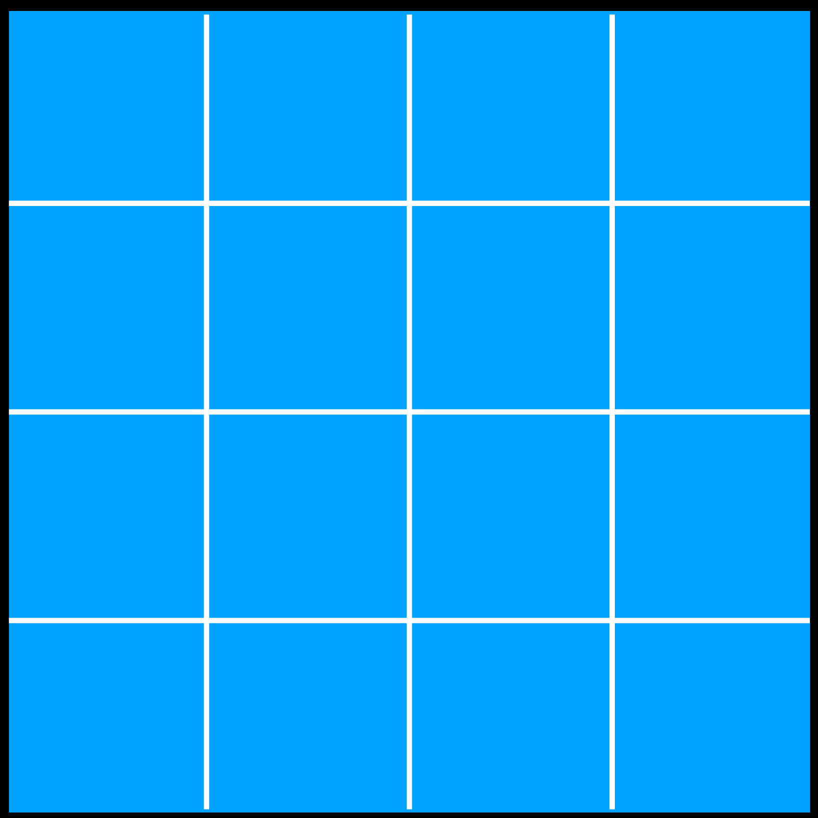
Problem Statement

- Given a low resolution (LR) image, create a new high resolution (HR) image. [Resolution = spatial resolution]



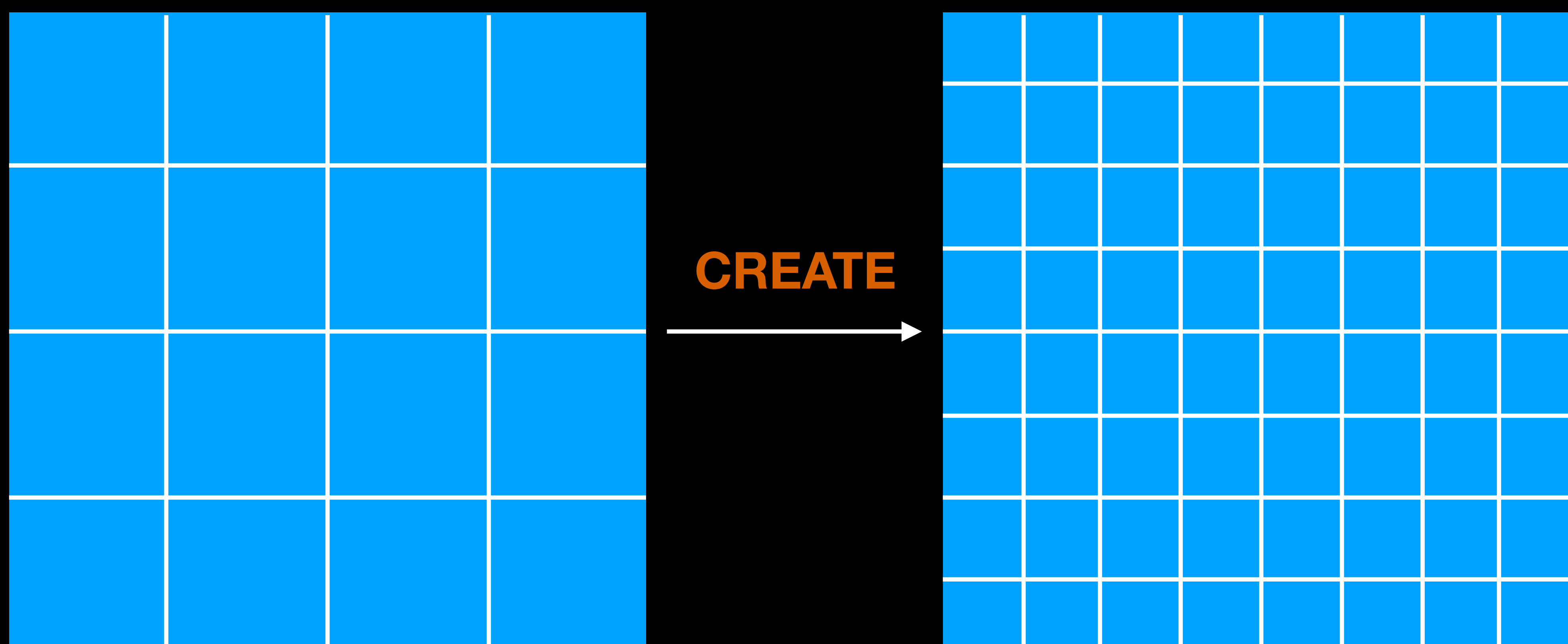
Spatial Resolution Test Chart

Problem Statement



Captured 4x4 LR Image

Problem Statement



Captured 4x4 LR Image

8x8 HR Image

So What’s the Problem?

Eastman Kodak 1975
0.01 Mpixel Camera

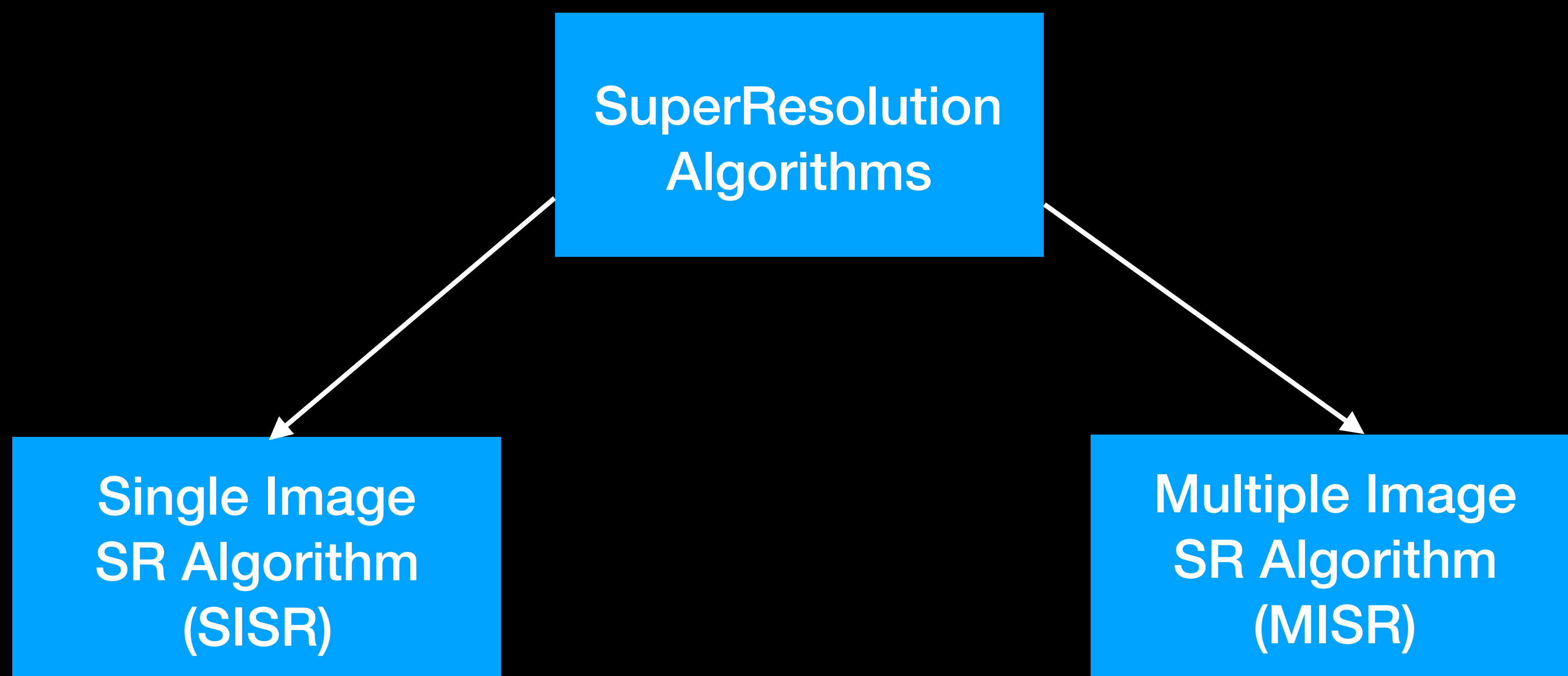
Hasselbad 2021
100 Mpixel Camera

CREATE = hardware based solution

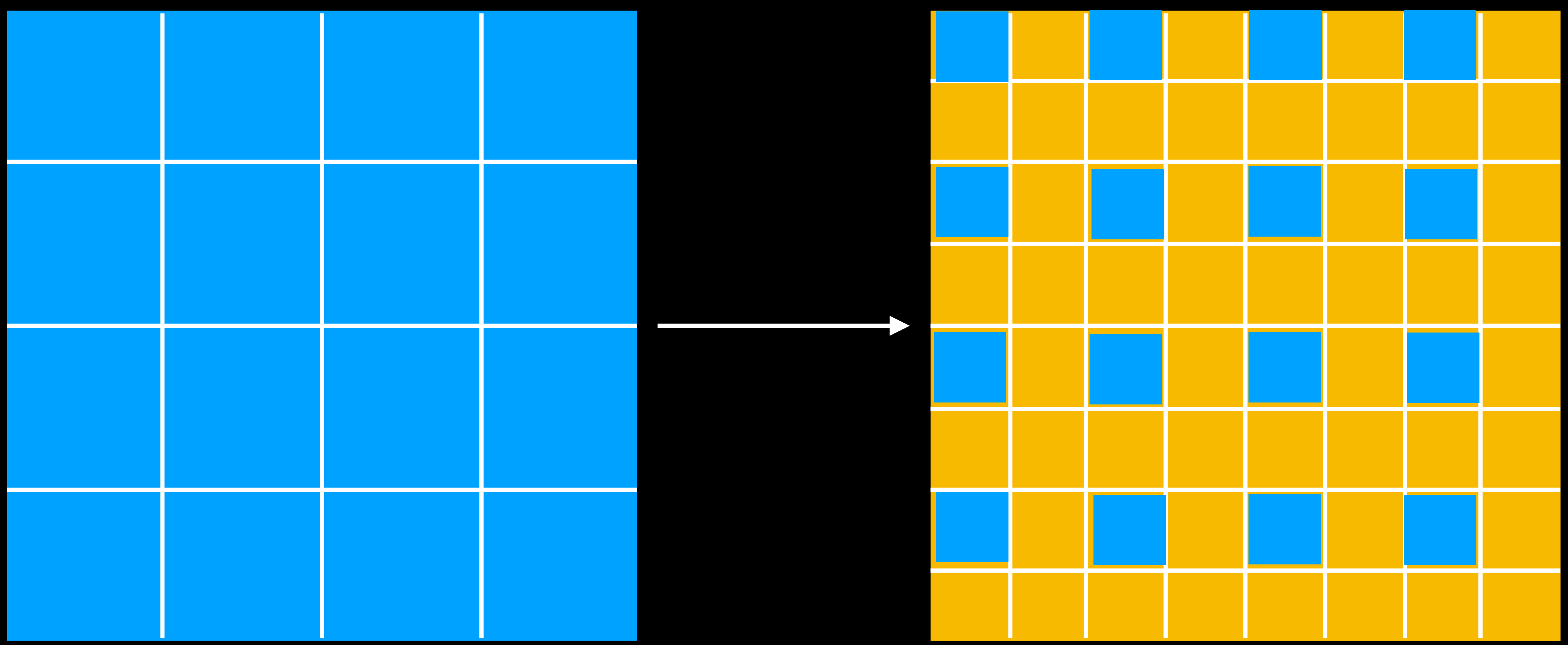
Problem Statement

- We want CREATE = a software based algorithm.
- By construction, this results in an ill posed problem.
- i.e. For a specific LR image, there are infinite number of possible HR images.

History

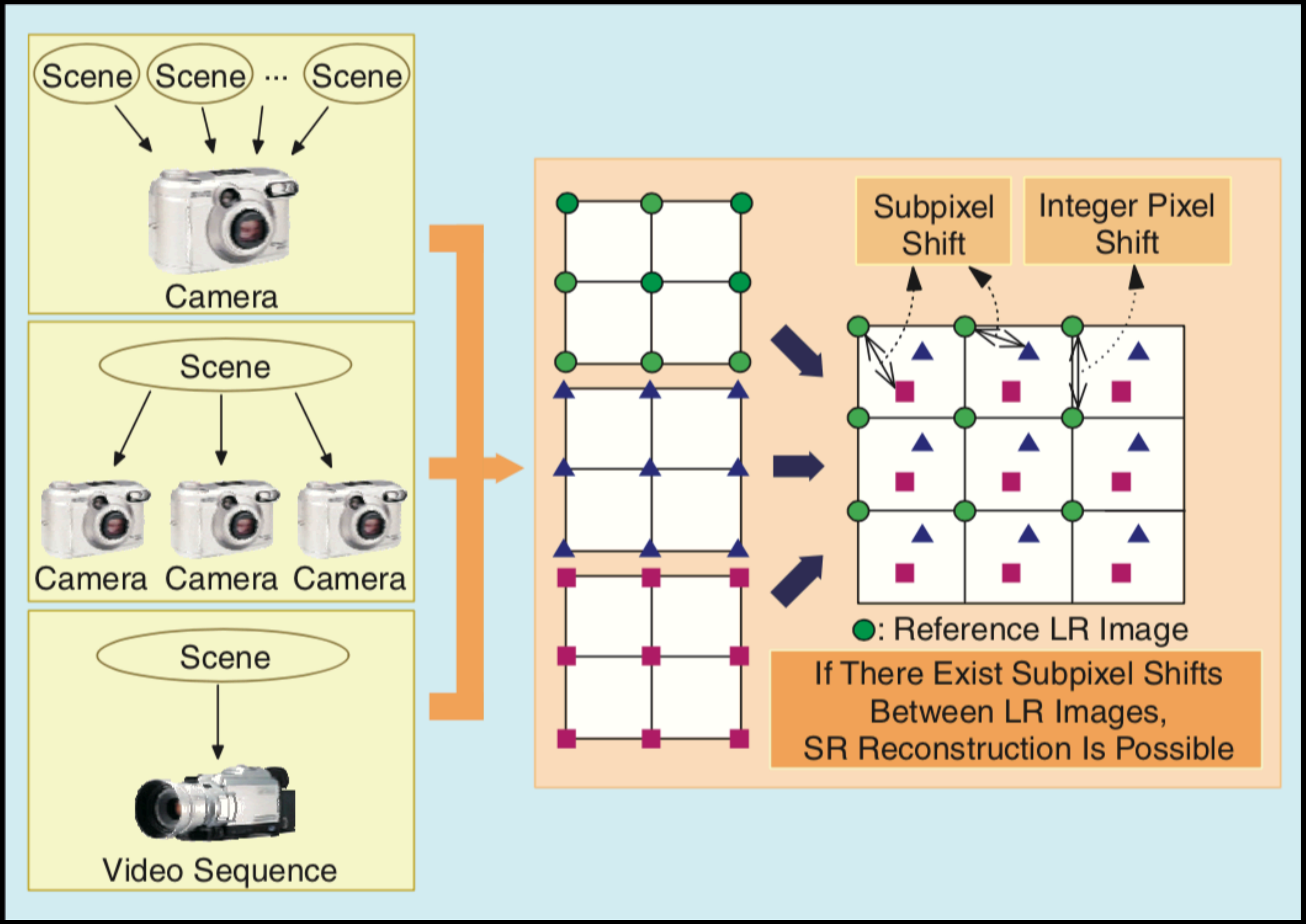

SuperResolution Algorithms
Single Image SR Algorithm (SISR)
Multiple Image SR Algorithm (MISR)

Example: Single Image



Perform Bilinear Interpolation to Approximate the Unknown Locations

Example: Multiple Image

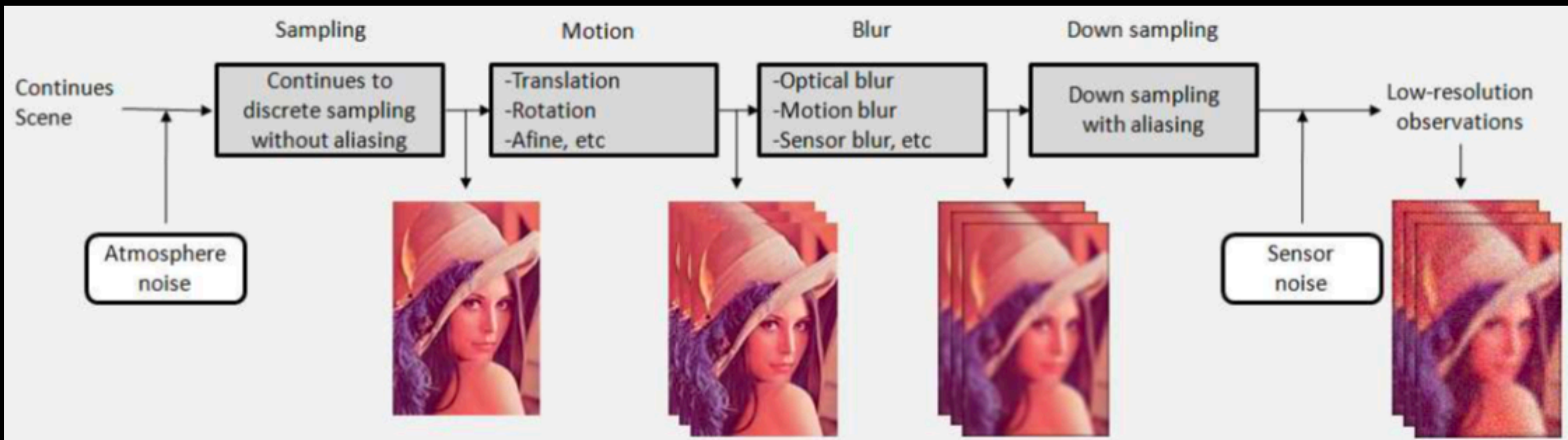


Revised Problem Statement: Given a set of LR images, create a new high resolution image.

Obtain a multiple image capture of the scene (=3 shown here) using one of the methods shown on the left.

Incoporate the new information into a finer spatial grid.

Observation Model



Associated with every LR image, is an image formation model / observation model.

The observation model consists of several major components:
blur, noise, sampling/downsampling and motion/registration.

The obsevation model “maps” a HR image into it’s LR counterpart.

Major Imaging Components

- Blur/Point spread function (PSF): By construction, the LR image is a low pass filtered version of the HR image.
- Sampling: Sampling occurs when the image sensor collect photons in discrete “buckets” associated with different pixels.
- Motion/Image registration: As shown in <u>Example: Mutiple Image</u>, the LR information needs to be incorporated into a denser grid.
- Noise: The noise in the LR image needs to be accurately modeled.
- Reconstruction: This model information needs to be combined in a novel way, resulting in a desirable HR image, sans artifacts.

History

- Super Resolution in the Frequency Domain (Reconstruction Based) (1984)
- Statistical Approaches (Reconstruction Based): ML & MAP (1990)
- Set Theoretic Approach (Reconstruction Based) (2001)
- Example Based Approach (Learning Based) (2002)
- Sparse Representation (Learning Based) (2010)

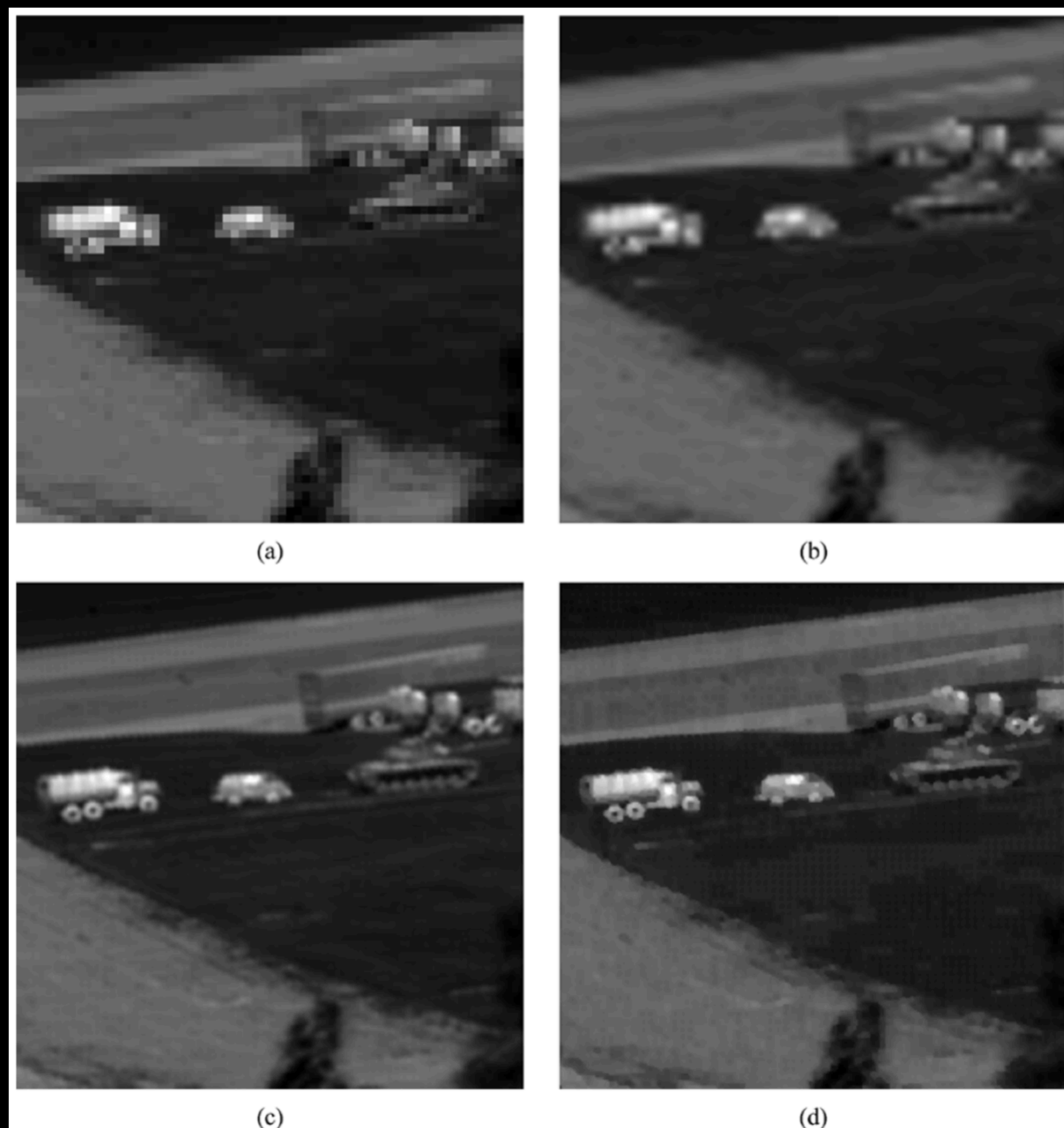
Historical References

- Super Resolution Image Reconstruction, A Technical Overview, Park, et. al., IEEE Spectrum 2003
- High Resolution Images from Low Resolution Compressed Video, Segall, et. al., IEEE Signal Processing Magazine 2003
- Computer Vision Applied to Super Resolution, Capel and Zisserman, IEEE Signal Processing Magazine 2003
- Multiple Frame Image Restoration and Registration, Tsai and Huang, CVIP 1984
- Super Resolution from Image Sequences, Irani and Peleg, Proceedings 10th International Conference on Pattern Recognition 1990
- Extracting of High Resolution Frames from Video Sequences, Schulz and Stephenson, IEEE Trans. Image Processing 1996
- Restoration of Single Super Resolution Image from Several Blurred, Noisy and Downsampled Measured Images, Elad and Feuer, IEEE Trans. Image Processing 1997
- Fast and Robust Multiframe Super Resolution, Farsiu, et. al., IEEE Trans. Image Processing 2004
- Artifact Reduction for Set Theoretic Super Resolution Image Reconstruction with Edge Adaptive Constraints and Higher Order Interpolants, Patti and Altunbasak, IEEE Trans. Image Processing 2001
- Example Based Super Resolution, Freeman, et. al., IEEE Computer Graphics and Applications 2002
- Image Super Resolution via Sparse Representation, Yang, et. al., IEEE Trans. Image Processing 2010

Comments: A/B Comparisons Were Performed to Evaluate Goodness (Multi Frame Approach)

LR Frame

Spline

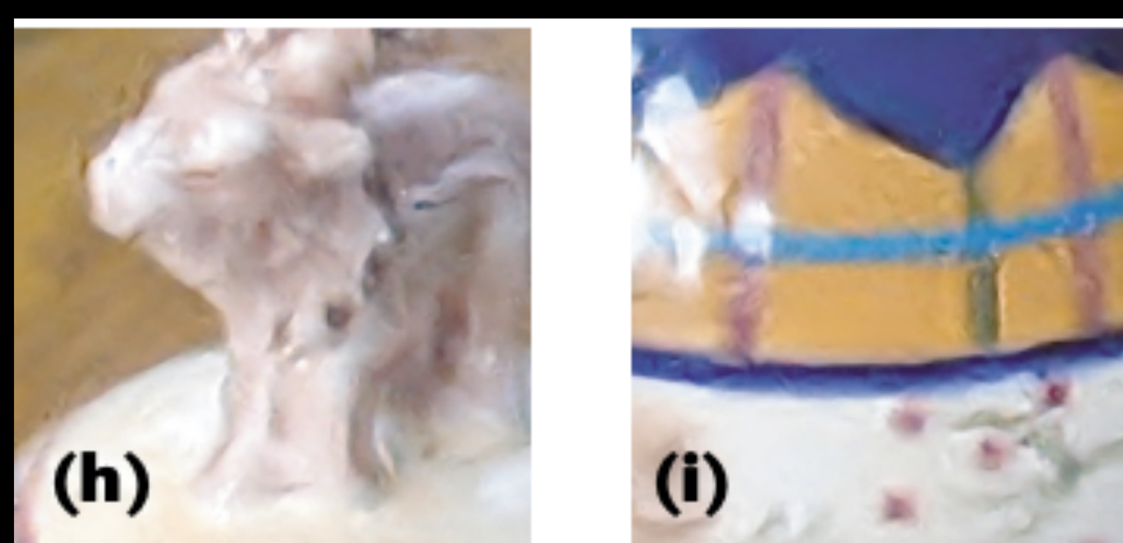


Other

New

Comments: A/B Comparisons Were Performed to Evaluate Goodness

Bi cubic Interpolation



Example Based SR

Comments: A/B Comparisons Were Performed to Evaluate Goodness

Spare Representation Based SR

Original Image

Comments

- One of the first papers published in SR was in 1984.
- In 2021, this problem is still being addressed, using deep learning (DL) tools.
- Current (DL) based SR focuses on the single image super resolution problem (SISR).
- Note: DL training sets consist of many image captures associated with many different scenes.
- As a result, multiple images are (still) used, but in an “indirect” way, since they are not associated with image registration.

Comments

- We discussed the image formation model / observation model.

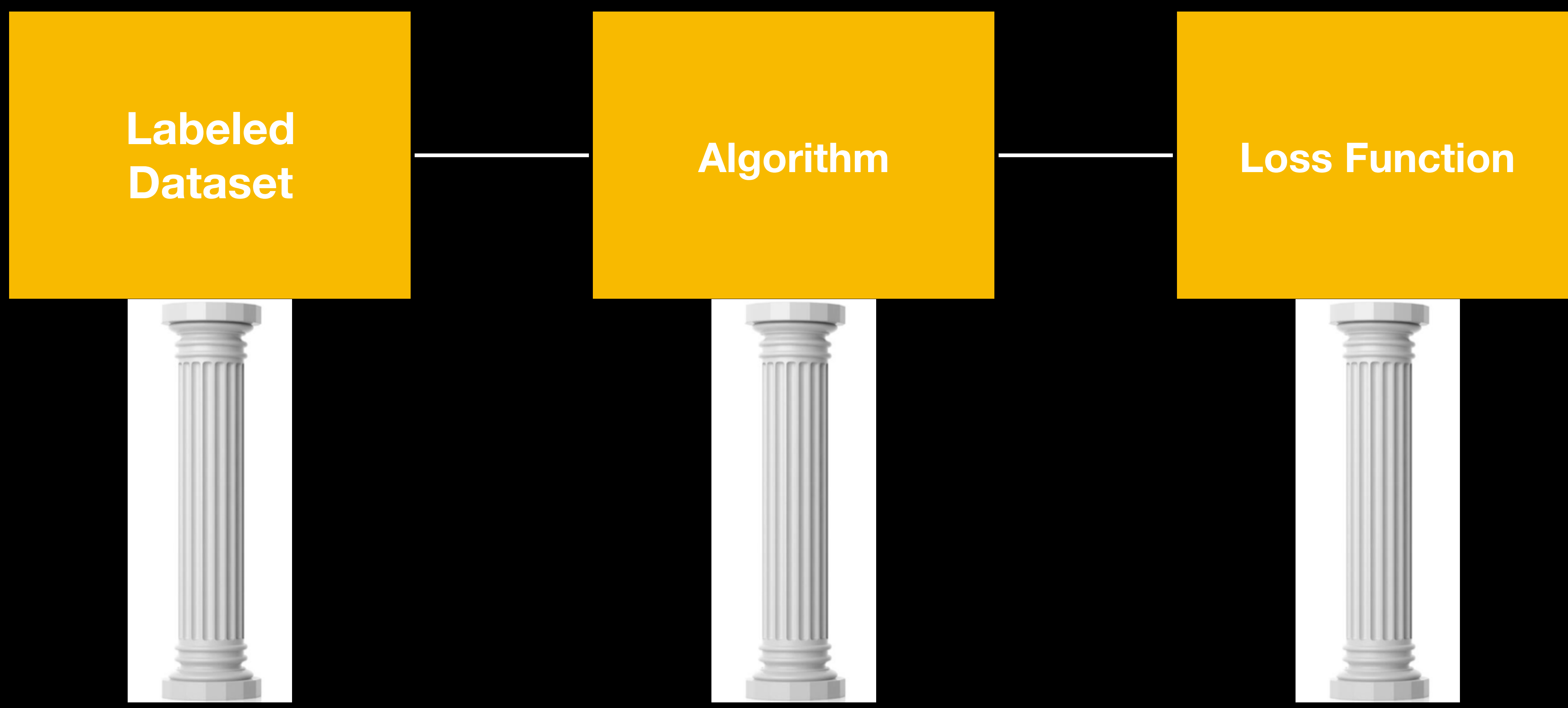
- This model has finally “re-appeared” in the deep learning (DL) realm of SISR, as of ~2020.

- Although prior DL SISR results were impressive, this model illustrated the huge “gap” that existed between understanding and “generating new results”.

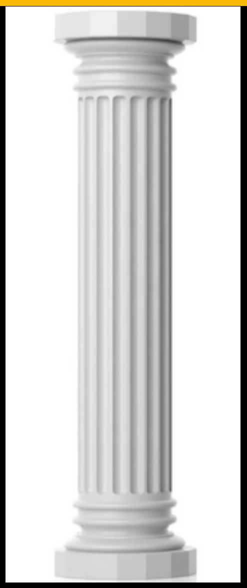
- Finally, in 2021, A/B comparisons are still being used. Only now, standard test sets (Set5, Set14, etc.) exist for comparison.

Comments

- Google productized MISR in 2018.
- An excellent writeup can be found at: https://ai.googleblog.com/2018/10/see-better-and-further-with-super-res.html
- In order to faithfully reproduce 2x zoom, >= four “offset” images need to be captured and merged.
- For 4x zoom, >= sixteen “offset” images need to be captured and merged.
- Previously, Google also introduced a SISR solution called RAISR.


Labeled Dataset
Algorithm
Loss Function

Labeled Dataset



Labeled Dataset

Algorithm	Training Dataset	Evaluation Dataset
SRCNN	91 & 395K Images from ImageNet	Set5
VDSR	91 to 291 Images from ImageNet	Set5, Set14, Urban100, B100
SRGAN	350K images from ImageNet	Set5, Set14, B100
Impressionism	Nitre Challenge Dataset (DF2K and DPED*)	DF2K

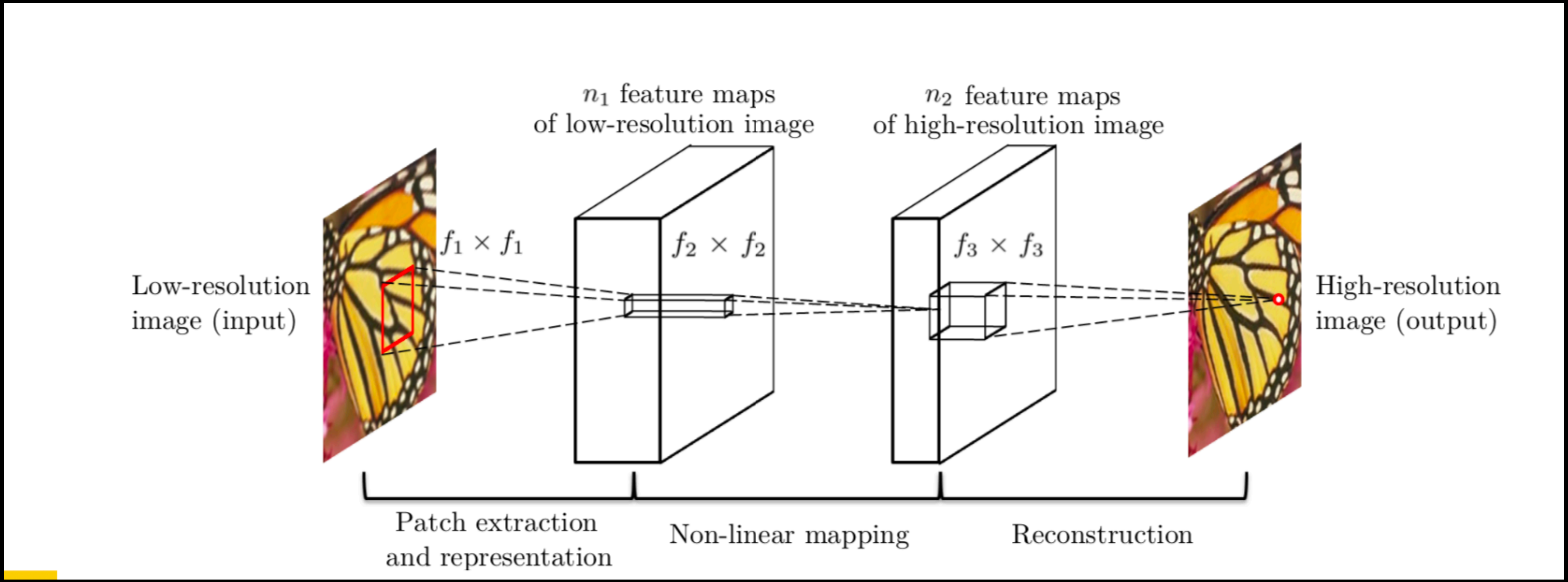
*In Nitre Challenge 2, DPED consists of LR cell phone images.
By correctly estimating the psf and the noise, {LR, HR} pairs can be created.


Algorithm

DL Based Single Image Super Resolution (SISR)

- SRCNN - Super Resolution using a CNN
- VDSR - Using Deeper CNN Networks
- SRGAN - Using a GAN
- Deep Learning for SISR - A Review of Architectures
- Impressionism

SRCNN



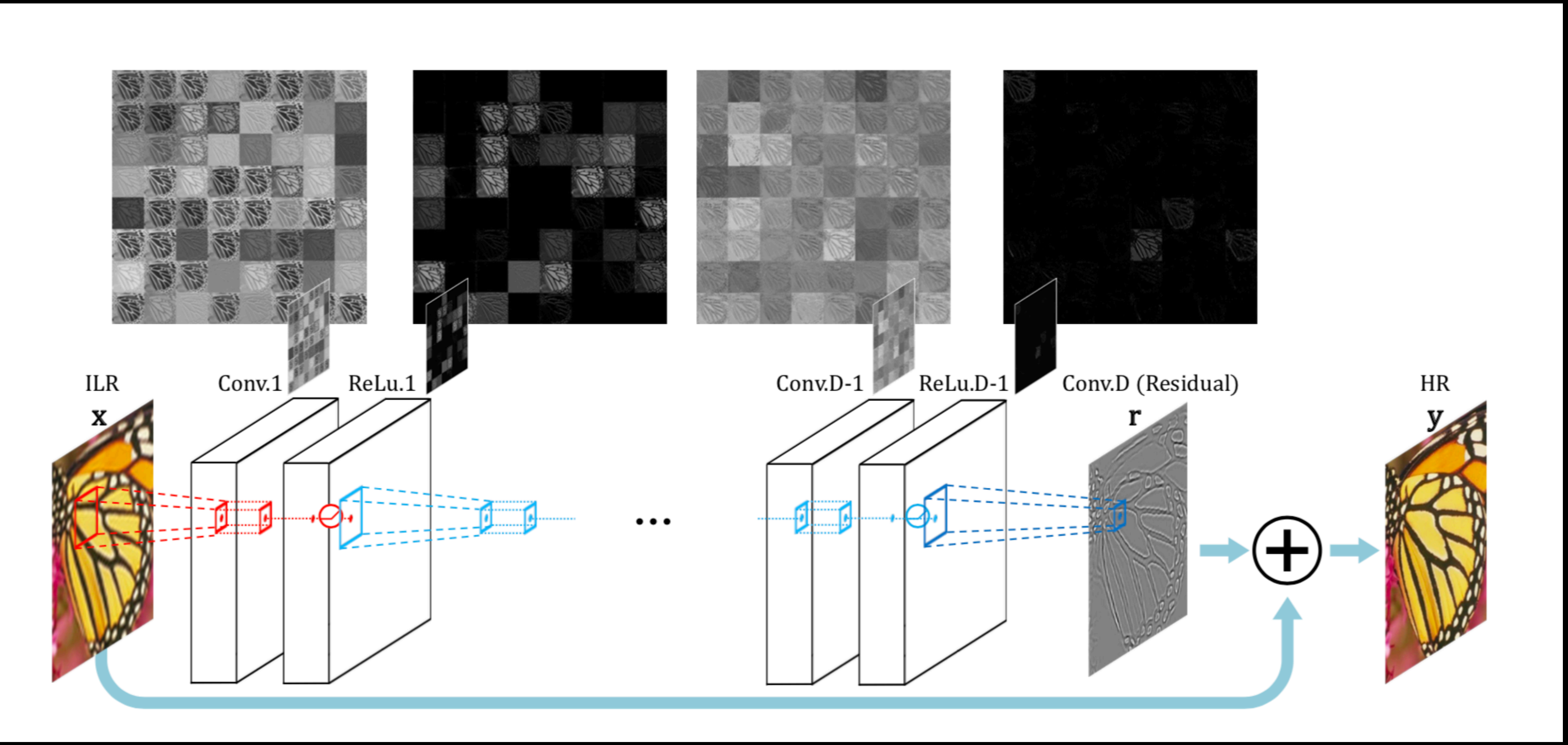
An end to end CNN was trained to perform super resolution.

First, HR images were subsampled, and then re-interpolated to their HR dimensions using bi-cubic interpolation.

Then, SL (using MSE loss) was applied to different bi-cubic-HR image pairs, to learn the weights of a **3 layer CNN**.

This was a proof of concept paper. Only 5 images (Set5) were used for evaluation.

VDSR



Inspired by the results of deeper networks (VGGNet), the authors trained a **twenty layer CNN**.

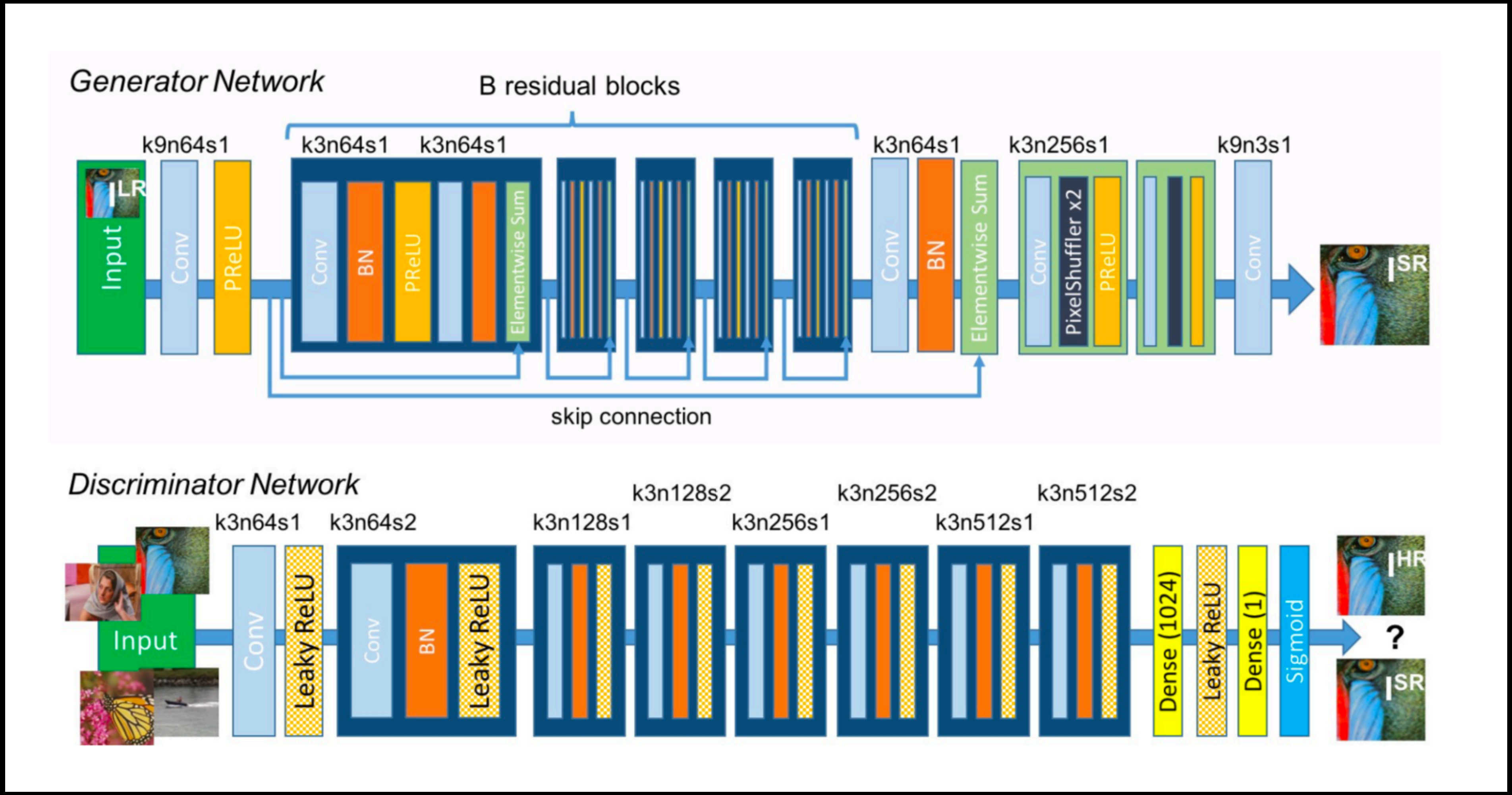
Each layer had 64 channels, and employed Relu non linearities.

In addition, a residual connection was added between the bi-cubic interpolated input image and the output.

To increase the training speed, a larger learning rate was used.

To avoid the exploding gradients problem (associated with the larger learning rate and A deeper network), weight clipping was employed.

SRGAN



Here, the authors applied a GAN to the SR image generation problem.

The generator consisted of a cascade of resnet blocks with skip connections.

The discriminator consisted of a cascade of VGG like CNN layers.

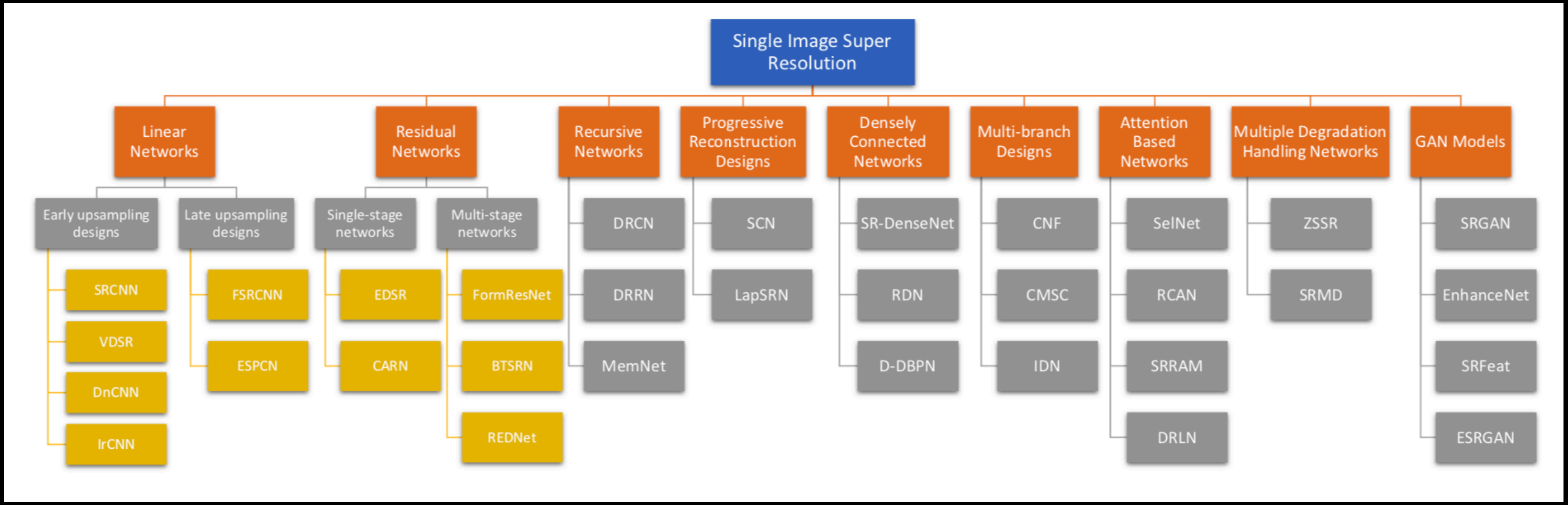
In addition to the standard GAN adversarial loss, the authors added a perceptual loss term.

The perceptual loss was defined to be the MSE loss between the computed feature maps (Ground Truth HR versus generator HR) in the semantic layers of the discriminator.

SISR Architectures

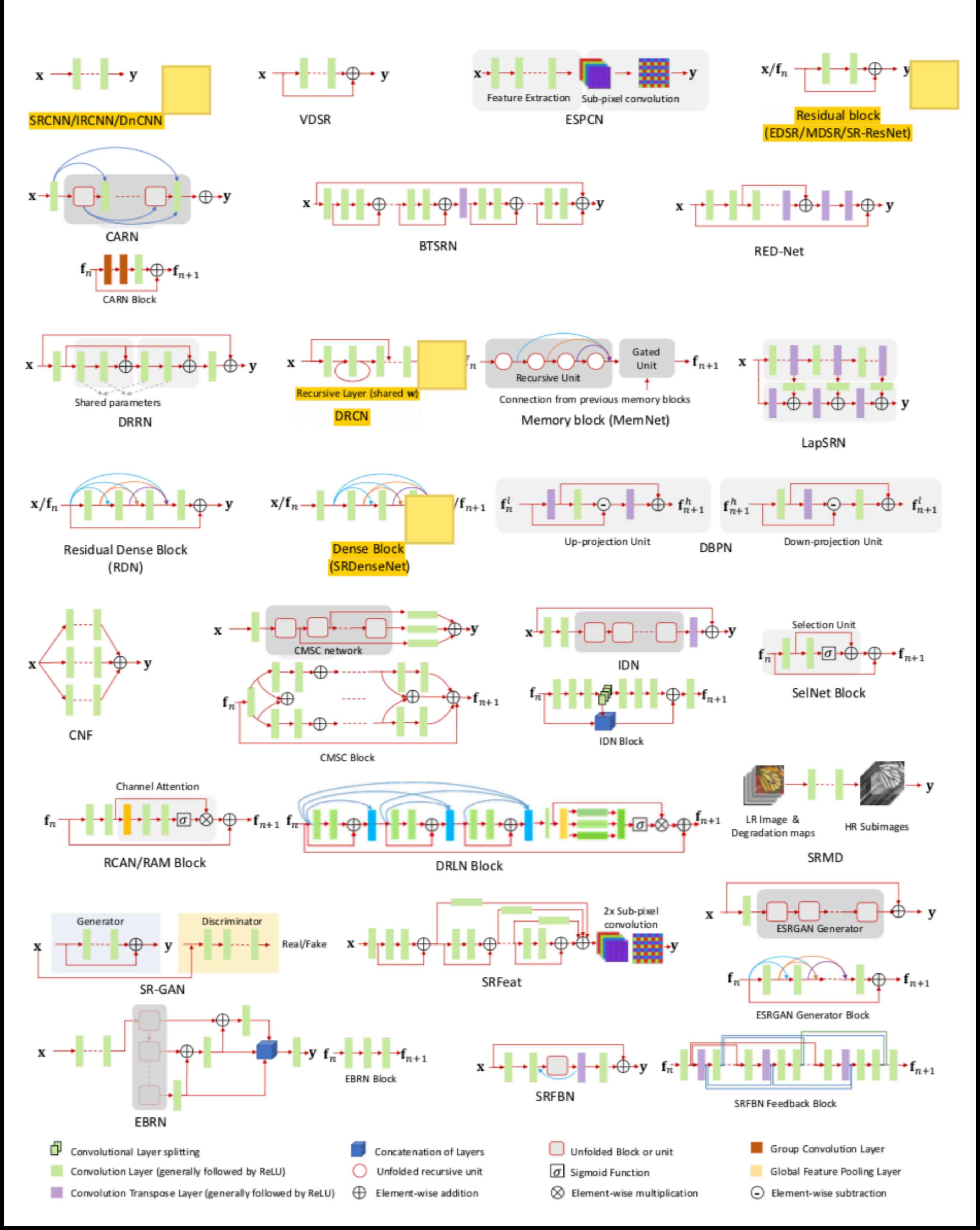
- Architectures for the SR problem were “exhaustively” investigated.
- The next few slides summarize this work.
- In general, deeper networks produced better results (EDSR, SRDenseNet).

Taxonomy



Single Image Super Resolution
Linear Networks
Residual Networks
Recursive Networks
Progressive Reconstruction Designs
Densely Connected Networks
Multi-branch Designs
Attention Based Networks
Multiple Degradation Handling Networks
GAN Models
Early upsampling designs
Late upsampling designs
Single-stage networks
Multi-stage networks
SRCNN
VDSR
DnCNN
IrCNN
FSRCNN
ESPCN
EDSR
CARN
FormResNet
BTSRN
REDNet
DRCN
DRRN
MemNet
SCN
LapSRN
SR-DenseNet
RDN
D-DBPN
CNF
CMSC
IDN
SelNet
RCAN
SRRAM
DRLN
ZSSR
SRMD
SRGAN
EnhanceNet
SRFeat
ESRGAN

Architectures



Details

Method	Input	Output	Blocks	Depth	Filters	Parameters	GRL	LRL	MST	Framework	Loss
SRCNN	bicubic	Direct		3	64	57k				Caffe	ℓ_2
FSRCNN	LR	Direct		8	56	12k				Caffe	ℓ_2
ESPCN	LR	Direct		3	64	20k				Theano	ℓ_2
SCN	bicubic	Prog.	✓	10	128	42k				Cuda-CovNet	ℓ_2
REDNet	bicubic	Direct		30	128	4,131k	✓	✓		Caffe	ℓ_2
VDSR	bicubic	Direct		20	64	665k	✓	✓		Caffe	ℓ_2
DRCN	bicubic	Direct		20	256	1,775k	✓			Caffe	ℓ_2
LapSRN	LR	Prog.	✓	24	64	812k	✓			MatConvNet	ℓ_1
DRRN	bicubic	Direct	✓	52	128	297k	✓	✓	✓	Caffe	ℓ_2
SRGAN	LR	Direct	✓	33	64	1500k				Theano/Lasagne	ℓ_2
DnCNN	bicubic	Direct		17	64	566k			✓	MatConvNet	ℓ_2
IRCNN	bicubic	Direct		7	64	188k			✓	MatConvNet	ℓ_2
FormResNet	bicubic	Direct	✓	20	64	671k	✓		✓	MatConvNet	ℓ_2, ℓ_{TV}
EDSR	LR	Direct	✓	65	256	43000k	✓	✓		Torch	ℓ_1
MDSR	LR	Direct	✓	162	64	8,000k	✓	✓	✓	Torch	ℓ_1
ZSSR	LR	Direct		8	64	225k	✓			Tensorflow	ℓ_1
MemNet	bicubic	Direct	✓	80	64	677k	✓	✓	✓	Caffe	ℓ_2
MS-LapSRN	LR	Prog.	✓	84	64	222k	✓	✓	✓	MatConvNet	ℓ_1
CMSC	bicubic	Direct	✓	35	64	1220k	✓	✓	✓	PyTorch	ℓ_2
CNF	bicubic	Direct		15	64	337K				Caffe	ℓ_2
IDN	LR	Direct	✓	31	64	796k	✓	✓		Caffe	ℓ_2,ℓ_1
BTSRN	LR	Direct	✓	22	64	410K	✓	✓		Tensorflow	ℓ_2
SelNet	LR	Direct		22	64	974K	✓	✓		MatConvNet	ℓ_2
CARN	LR	Direct	✓	32	64	1,592K	✓	✓	✓	PyTorch	ℓ_1
SRMD	LR	Direct		12	128	1482k				MatConvNet	ℓ_2
SRDenseNet	LR	Direct	✓	64	16-128	5,452k	✓	✓		TensorFlow	ℓ_2
EnhanceNet	LR	Direct	✓	24	64	889k		✓		TensorFlow	ℓ_2, ℓ_t, GAN
SRFeat	LR	Direct	✓	54	128	6,189k	✓	✓		TensorFlow	ℓ_2, ℓ_p, GAN
SRRAM	LR	Direct	✓	64	64	1,090K	✓	✓	✓	Tensorflow	ℓ_1
D-DBPN	LR	Direct	✓	46	64	10,000K	✓	✓		Caffe	ℓ_2
RDN	LR	Direct	✓	149	64	21,900k	✓	✓		Torch	ℓ_1
ESRGAN	LR	Direct	✓	115	64	38,549k	✓	✓		Pytorch	ℓ_1
SRFBN	LR	Direct	✓	28	64	3,500k	✓	✓	✓	Pytorch	ℓ_1
RCAN	LR	Direct	✓	500	64	16,000k	✓	✓	✓	Pytorch	ℓ_1
DRLN	LR	Direct	✓	160	64	34,000k	✓	✓	✓	Pytorch	ℓ_1
EBRN	LR	Direct	✓	173	64	7,900k		✓		Pytorch	ℓ_1, ℓ_2

Wait! Are We Solving the Right Problem??

General Formulation

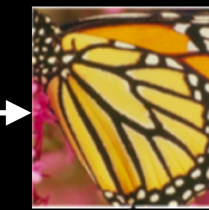
- Training pairs were generated from HR images. Let's denote this by: {INPUT, HR}
- Case1: INPUT = subsampled HR images from the constrained set
- Case2: INPUT = apply bi-cubic interpolation to the Case1 images
- SL was then applied to the {INPUT, HR} pairs, to train a deep transpose convolution and / or CNN architecture == Network
- A new INPUT from a small test set (Set5, Set14) was applied to the network, to demonstrate the efficacy of the network.

Step1
Create {INPUT_train,
HR_train} image pairs

Sub-sample

Case1

Bicubic Interpolation

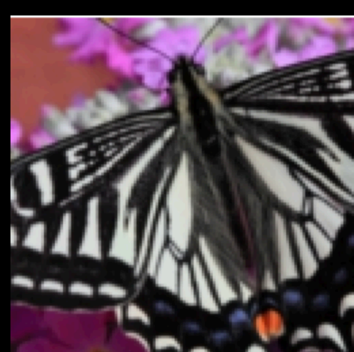


Case2

Step2
Train
via
SL

INPUT_train Image N → Network → HR_train Image N

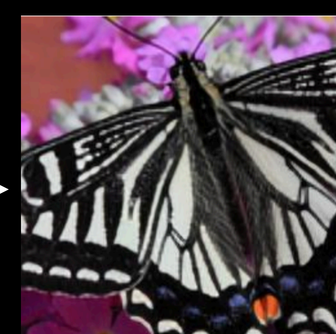
Step3
Test



Case2

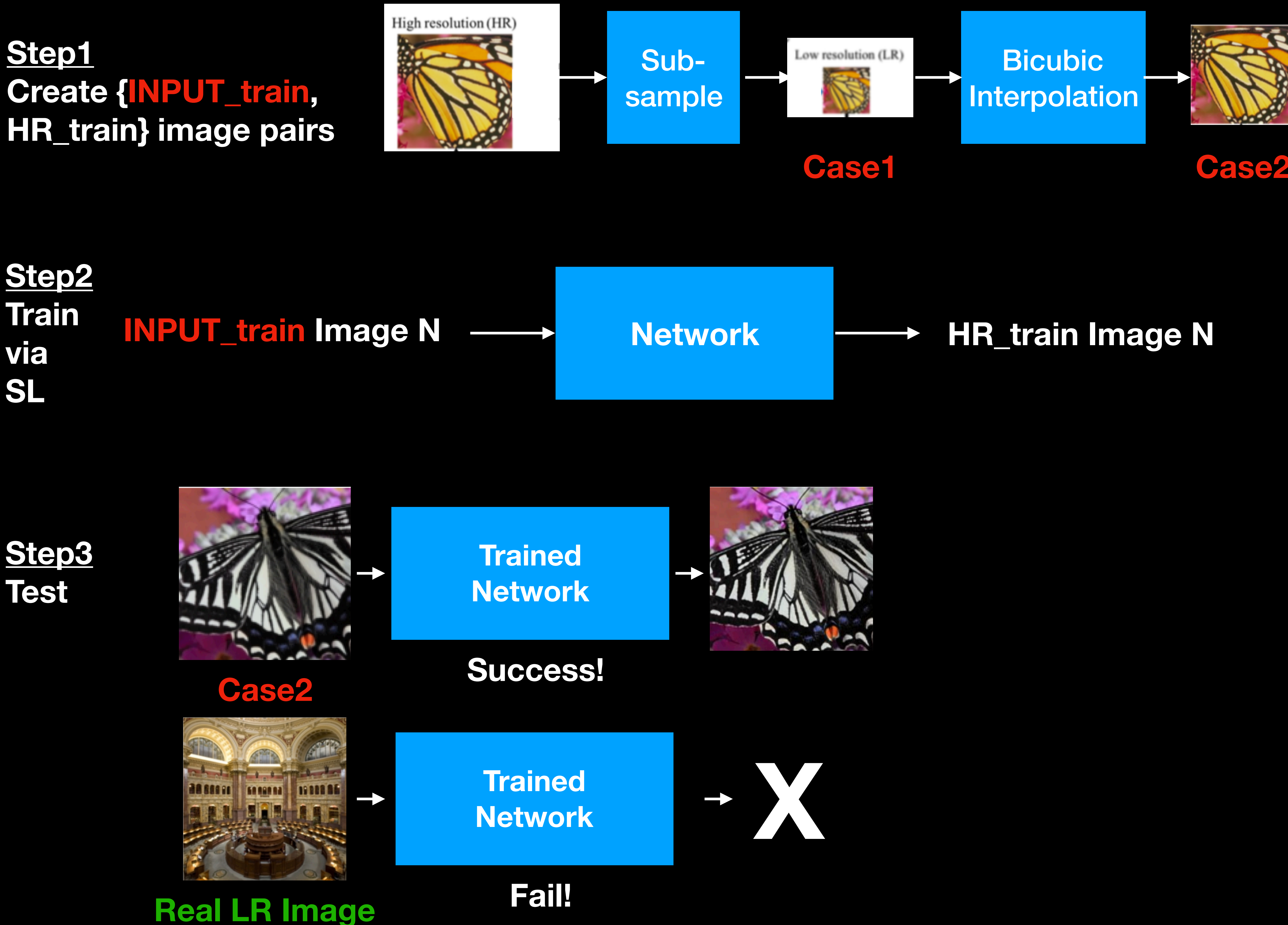
Trained Network

Success!



Wait! Are We Solving the Right Problem?

- However, this result did NOT generalize.
- When the trained network was fed a real world LR image, a SR image was NOT generated.


Step1
Create {INPUT_train,
HR_train} image pairs
High resolution (HR)
Sub-
sample
Low resolution (LR)
Bicubic
Interpolation
Case1
Case2
Step2
Train
via
SL
INPUT_train Image N
Network
HR_train Image N
Step3
Test
Trained
Network
Success!
Case2
Trained
Network
X
Fail!
Real LR Image

Wait! Are We Solving the Right Problem??

In order to understand one of the reasons why, we need to return to the observation model / image formation model discussed earlier.

In it's simplest form, Step 1 also needed to incorporate blur and noise, as shown above.

One of the goals of the NITRE SR Challenges was to move DL based SISR research in this direction.

New Problem Formulation(s)

NITRE Challenge1

We are given a set of LR images and a different set of HR images.

All of the LR images have undergone the same degradation.

NO {LR, HR} IMAGE PAIRS exist.

What is the underlying degradation?

NITRE Challenge2

We are given a set of low quality, LR cell phone images.

We want to generate high quality, HR output images.

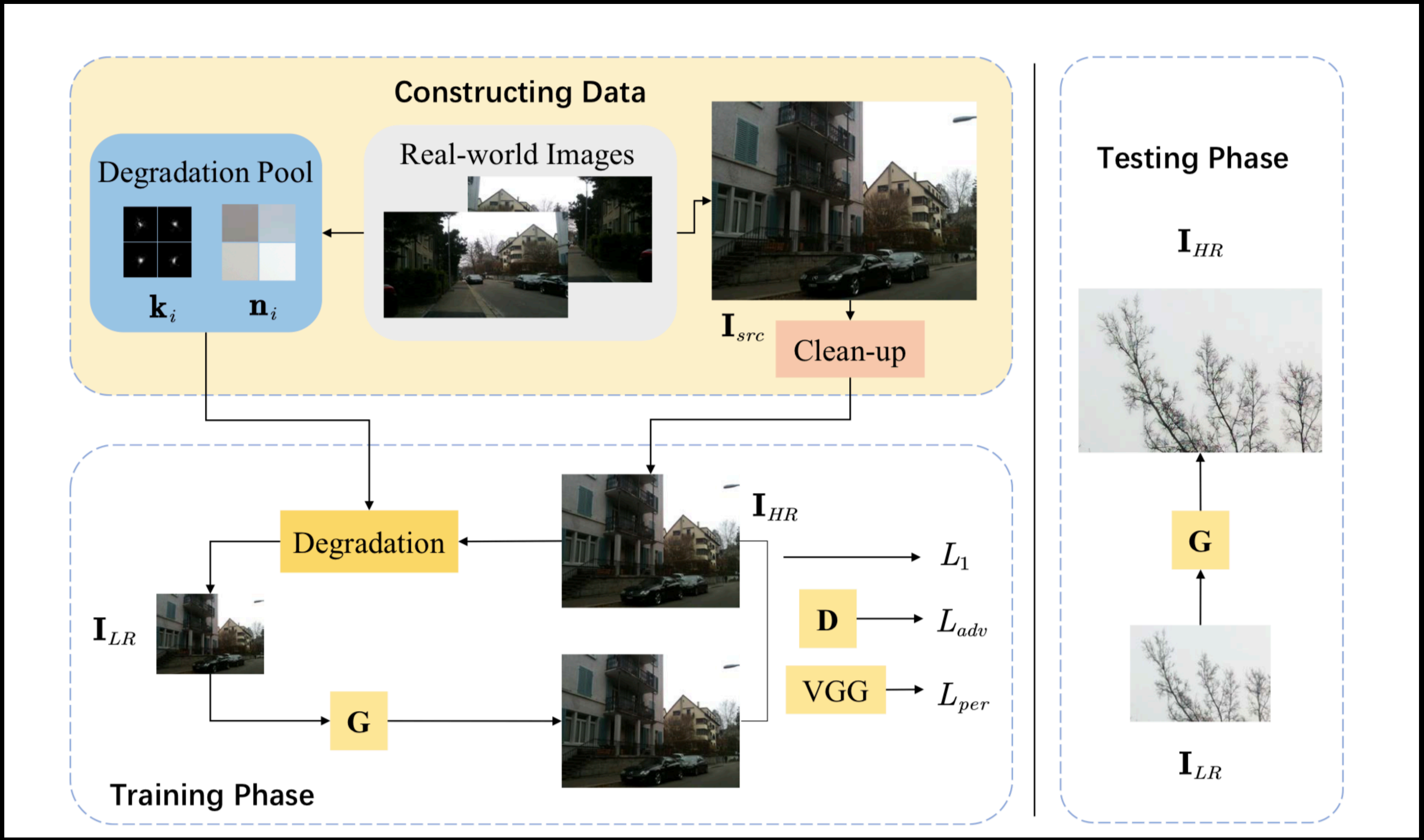
New Problem Formulation(s)

- Under the new problem formulation(s), researchers were tasked with estimating these parameters for the observation model / image formation model.

- Now, with a good observation model / image formaton model, improved LR images (images that “more closely” reflected real world images) could be generated.

- When the new {LR, HR} training set was then used to train a network, improved results were obtained for real world images created using a similar model.

Impressionism

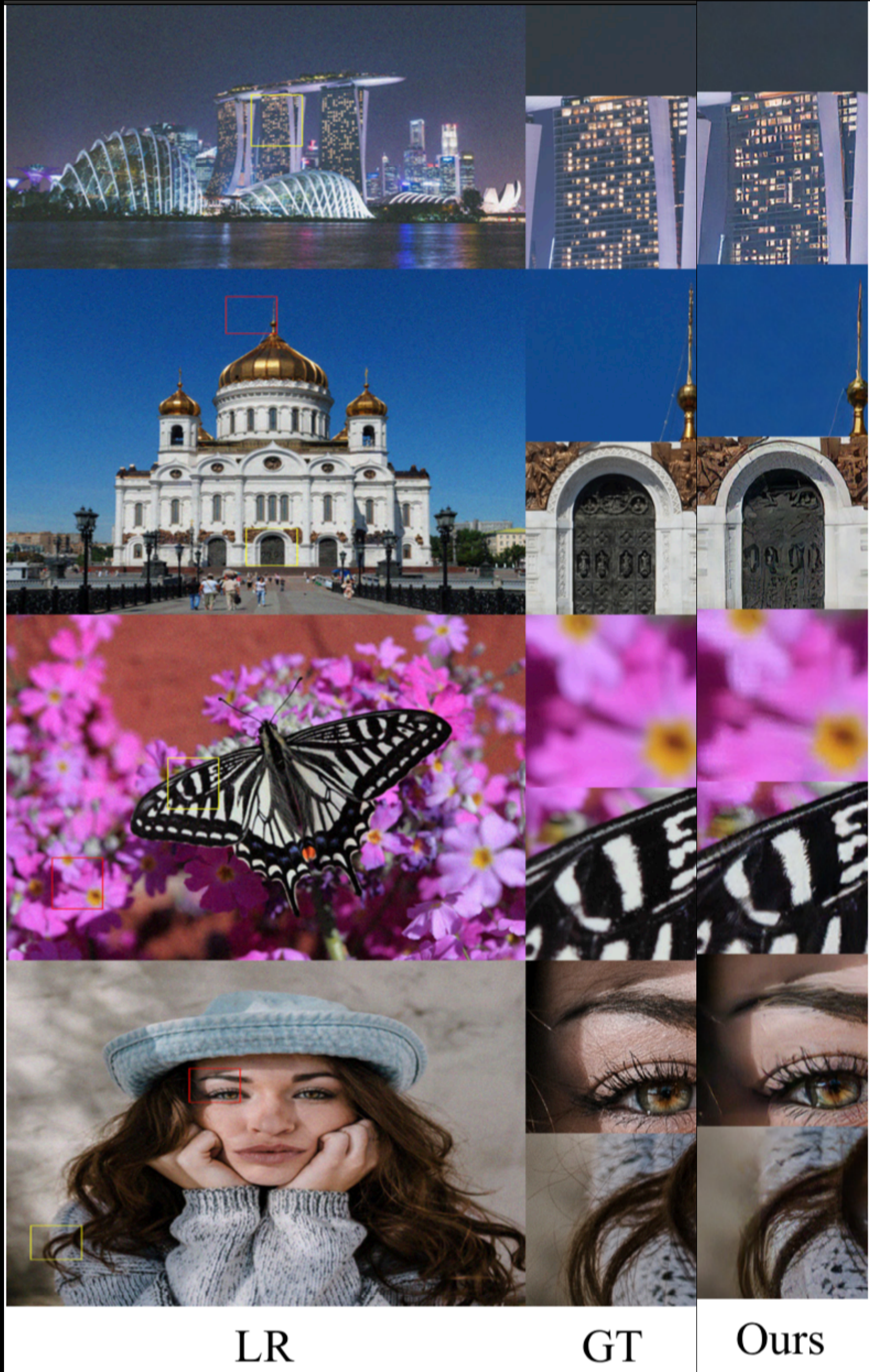


The authors used ESRGAN to generate their SR images.

The authors utilized a {LR, HR} training set, created by using an obsevation model / image formation model that incorporated accurate blur and noise information.

Impressionism

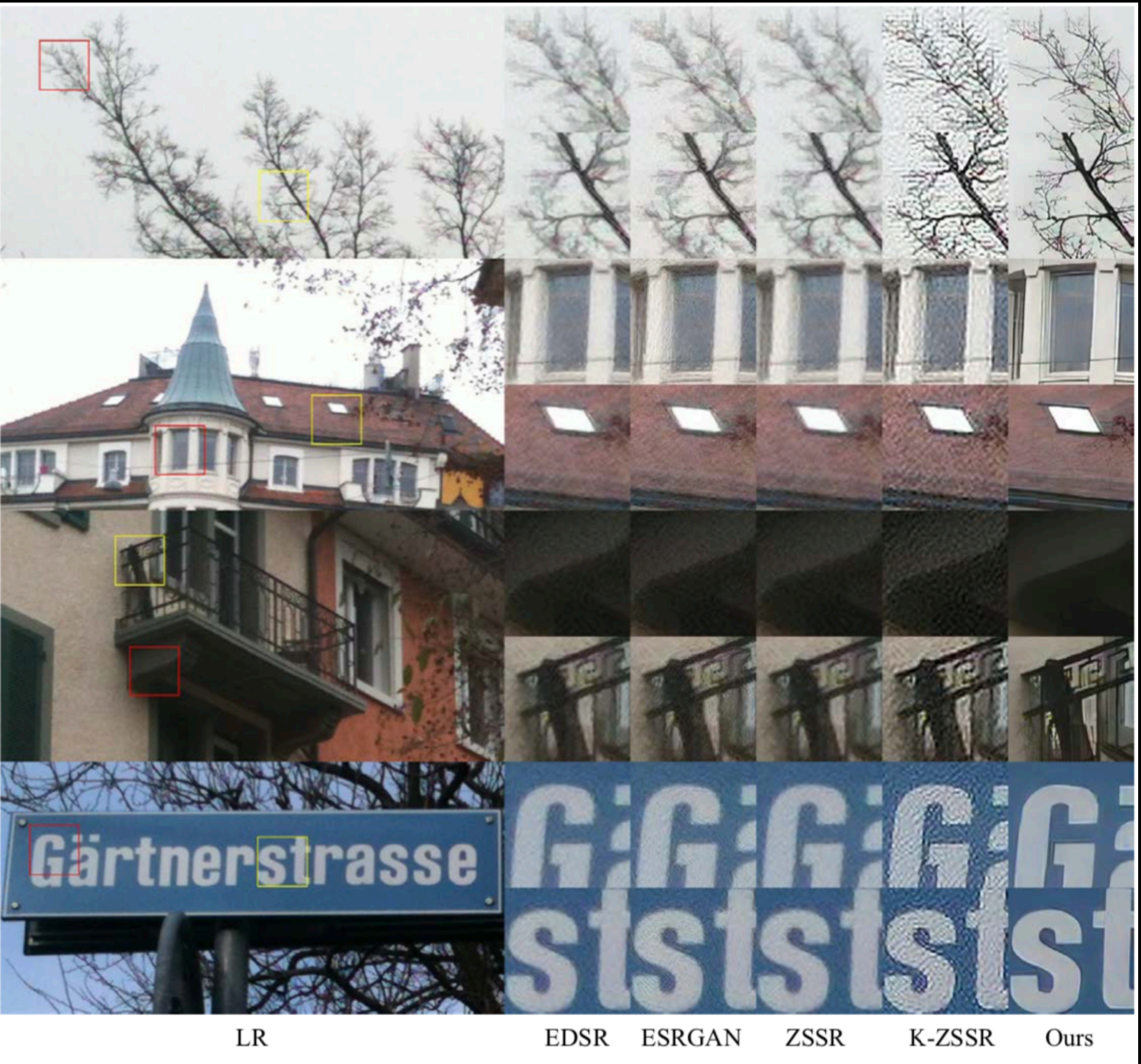
- Step1: From the given LR images = LR_given, an accurate estimate of the image blur and noise was obtained.
- Step2: Next, LR_given was subsampled, and then bi-cubic interpolated to a resolution higher than the original LR_given, producing HR_clean
- Step3: Using the LR image formation model / observation model, LR_new images were created using the HR_clean images as input.
- ESRGAN was then trained, using the newly constructed training set: {LR_new, HR_clean}



When compared with other approaches, Impressionism "impressed", since no false or annoying artifacts were created.

Upon closer inspection, some of the side effects resulting from HR_clean can be seen though.

Specifically, fine details are missing in the temple door and the skin of the girl. In addition, in the last example, fine strands of hair are missing in the SR generated image.



With respect to the LR cell phone images:
The authors should have included the HR_clean images in the comparison.

Under the current presentation, there is no true reference point.

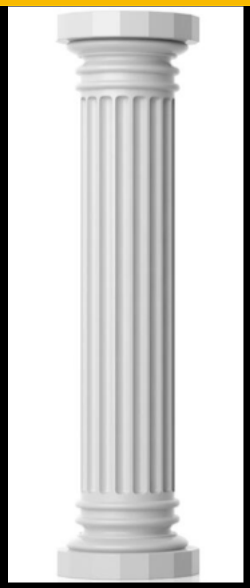
Comments

- Although DL SISR has produced excellent results, there is still a long way to go.
- Currently, evaluation has progressed from Set5 and Set14 to Urban100 (buildings) and B100 (textures).
- Ideally, we would like a SR network to be successful for any arbitrary input image - and not just input images restricted to constrained test sets like Urban100 or B100.

Algorithm

- SRCNN - Three Layer CNN
- VDSR - Twenty Layer CNN with a residual connection from the input to the output
- SRGAN - GAN {VGG like CNN Discriminator network and a ResNet with skip connections Generator network}
- Impressionism - ESRGAN and {LR, HR} pairs that utilize an observation model describing the LR image formation process

Loss Function



Loss Function

- Pixel Loss = MSE of the GT image and the SR image
- Perceptual Loss = MSE of the GT high level feature maps and the SR image feature maps
- SRCNN - Pixel Loss
- VDSR - Pixel Loss
- SRGAN - Perceptual Loss and GAN Adversarial Loss
- Impressionism - Pixel Loss, Perceptual Loss and GAN Adversarial Loss

References

- Image Super Resolution using Deep Convolutional Networks, Cong, et. al., arXiv December 2014
- Accurate Image Super Resolution using Very Deep Convolutional Networks, Kim, et. al., arXiv November 2015
- Photo Realistic Single Image Super Resolution using a Generative Adversarial Network, Ledig, et. al., arXiv September 2016
- Pixel Recursive Super Resolution, Dahl, et. al., arXiv February 2017
- Enhanced Deep Residual Networks for Single Image Super Resolution, Lim, et. al., CVPR 2017
- Image Super Resolution Using Dense Skip Connections, Tong, et. al., CVPR 2017
- Deep Learning for Single Image Super Resolution: A Review, Yang, et. al., arXiv August 2018
- Real World Super Resolution via Kernel Estimation and Noise Injection, Ji, et. al., CVPR 2020
- NITRE 2020 Challenge on Real World Image Super Resolution: Methods and Results, Lugmayer, et. al., CVPR Workshop 2020