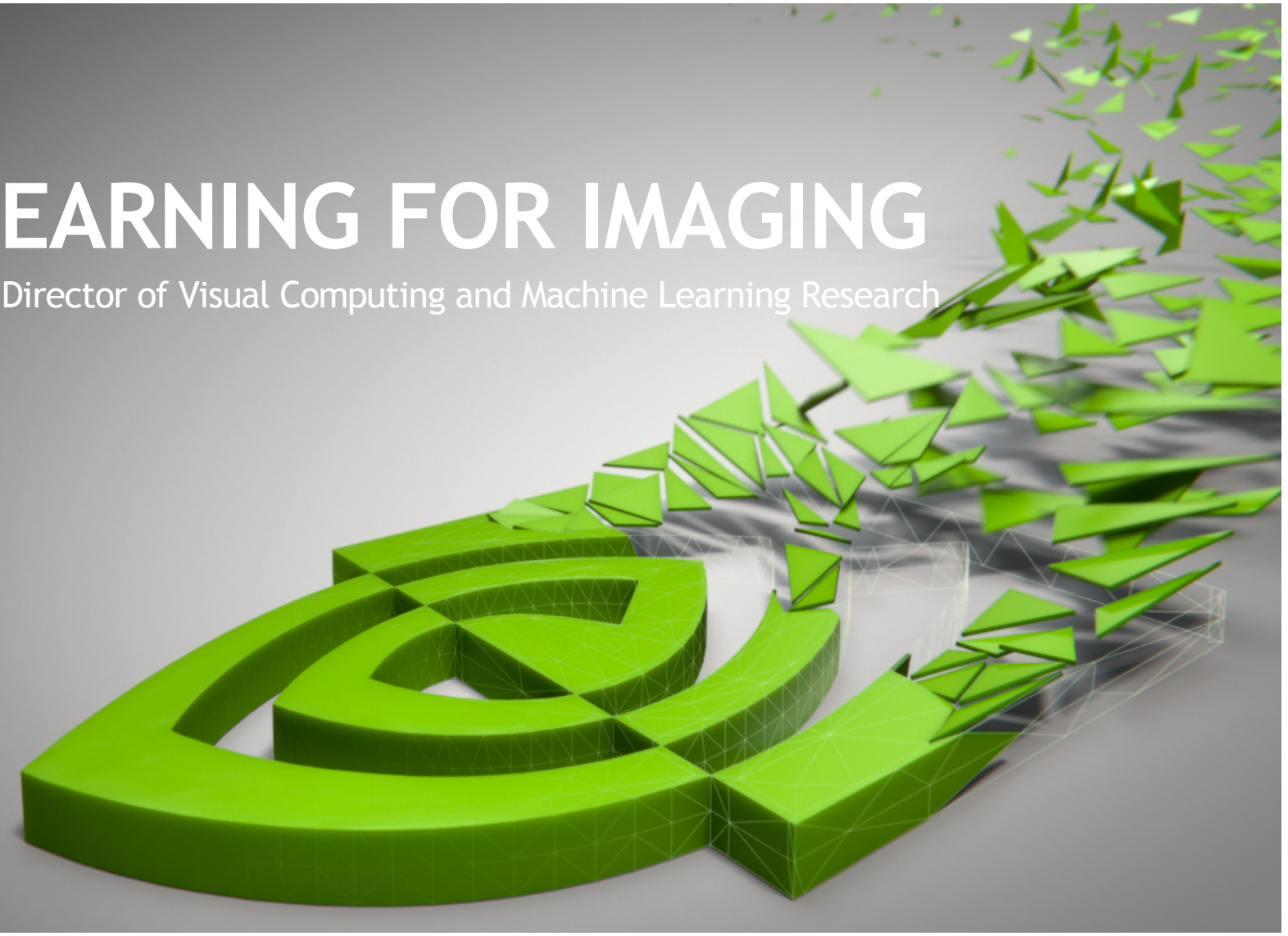
DEEP LEARNING FOR IMAGING

Jan Kautz, Senior Director of Visual Computing and Machine Learning Research

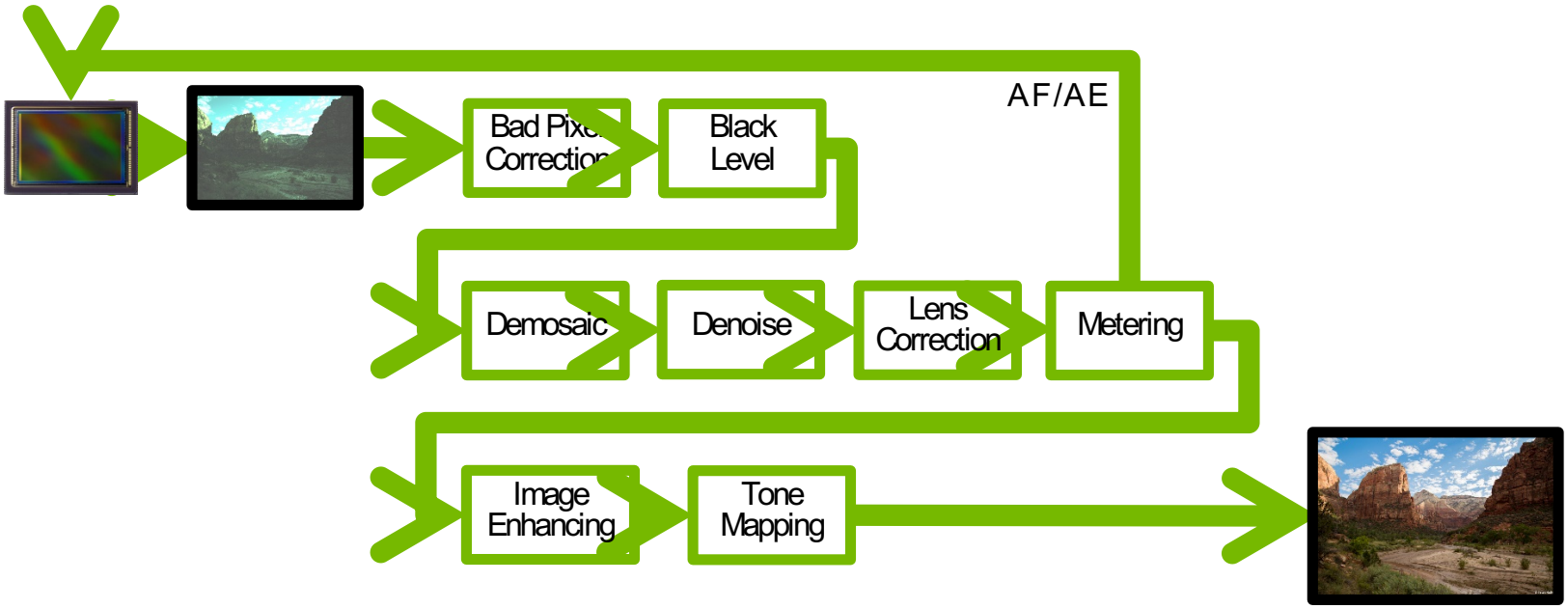
NVIDIA.




© Orazio Gallo

MOTIVATION

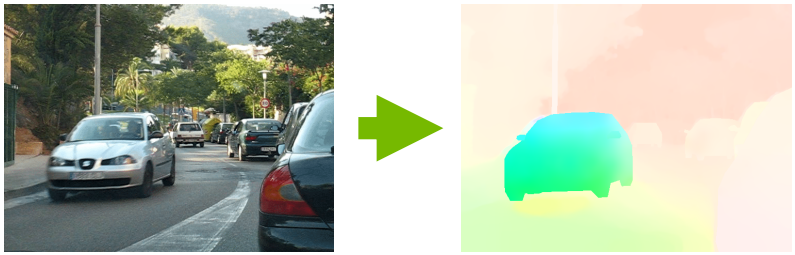
(the long path from photons to a digital image)



MOTIVATION

Or you might have several images

Stereo



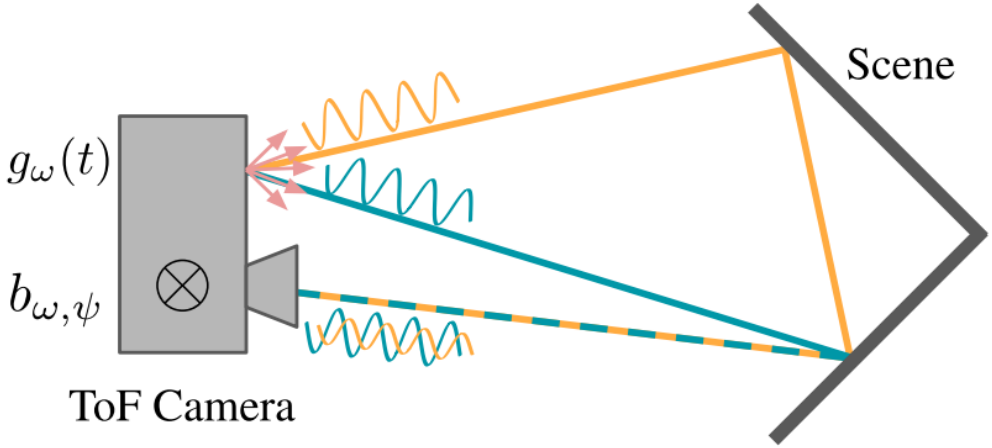
Optical Flow

Deblurring

MOTIVATION

Non-standard imaging sensors

Event-based Cameras



Time-of-flight Cameras

OUTLINE

Image Processing with Deep Learning

- Denoising
- Demosaicking
- Loss functions for Image Processing

Multiple Images

- DL for Stereo
- DL for Optical Flow
- DL for Deblurring

DL for Other Sensors

- Event-based
- Time-of-Flight

IMAGE PROCESSING WITH DEEP LEARNING

DENOISING

Several types of noise involved in the image formation:

- Photon shot noise
- Dark current (AKA thermal noise)
- Photo-response non-uniformity
- Readout noise:
  - Reset noise (charge-to-voltage transfer)
  - White noise (during voltage amplification)
  - Quantization noise (ADC)

DENOISING

Before the ML era

1) Signal processing (aka Fourier, Wavelet) techniques

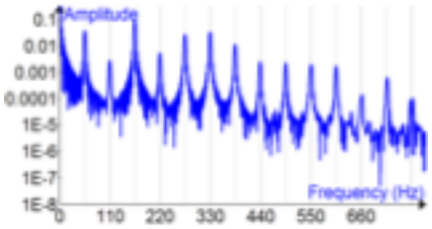


Image from https://en.wikipedia.org/wiki/Fourier_analysis

2) Non-Local approaches (NLM, BM3D)

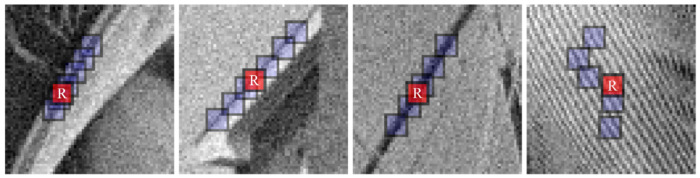


Image from http://www.cs.tut.fi/~foi/3D-DFT/BM3DDEN_article.pdf

3) Fixed (DCT, Fourier, Wavelet) vs. Learned dictionaries

External dictionaries

Internal dictionaries

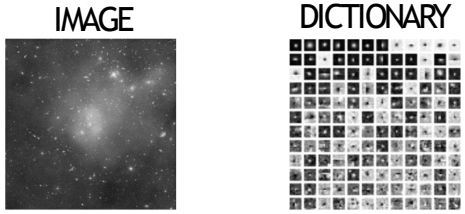


Image from https://arxiv.org/pdf/1304.3573.pdf

DENOISING

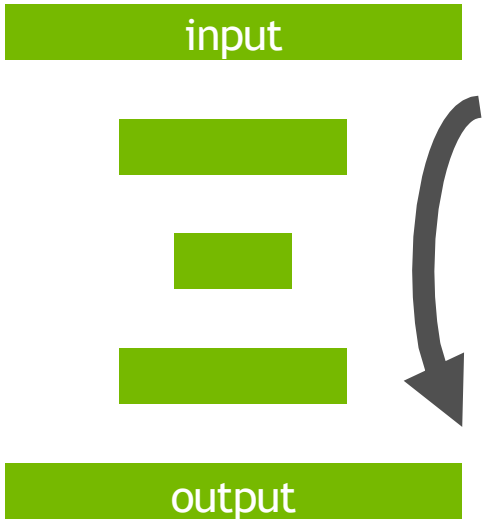
At the beginning of the ML era

Auto-Encoder (AE): learn **x = f(x)** function

Need to avoid the **trivial solution f = I**:

- **shrink space** (compressed representation)
- force **sparsity**

Denoising AE: input is corrupted by noise, x = f(x + n)



Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, **"Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion"**, 2010.

DENOISING

At the beginning of the ML era (another stacked denoising AE)

A stacked denoised AE for **denoising and inpainting**

Impose **sparse** representation (an additional term in the cost function)

Image from http://staff.ustc.edu.cn/~linlixu/papers/nips12.pdf

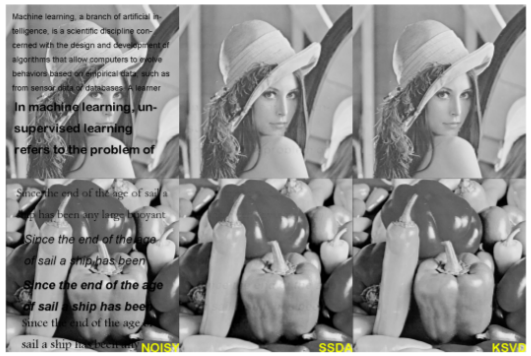


Image from http://staff.ustc.edu.cn/~linlixu/papers/nips12.pdf

Junyuan Xie, Linli Xu, Enhong Chen, **Image Denoising and Inpainting with Deep Neural Networks**, NIPS 2012

DENOISING

At the beginning of the ML era

"A plain multilayer perceptron (MLP) applied to image patches" (the simplest architecture ever ☺)

Can be trained on a **single or multiple noise levels** (assuming white Gaussian noise)

Image quality **comparable with SOA** BM3D (@ 0.1% of the engineering effort ☺)

A **large training dataset** is required

Can be applied to other kinds of noises (the DNN **learns both processing filters and basis** to represent the image)

H. C. Burger, C. J. Schuler and S. Harmeling, "**Image denoising: Can plain neural networks compete with BM3D?**," IEEE Conference on Computer Vision and Pattern Recognition, 2012

DENOISING

Kernel Predicting Network (KPN) (1)

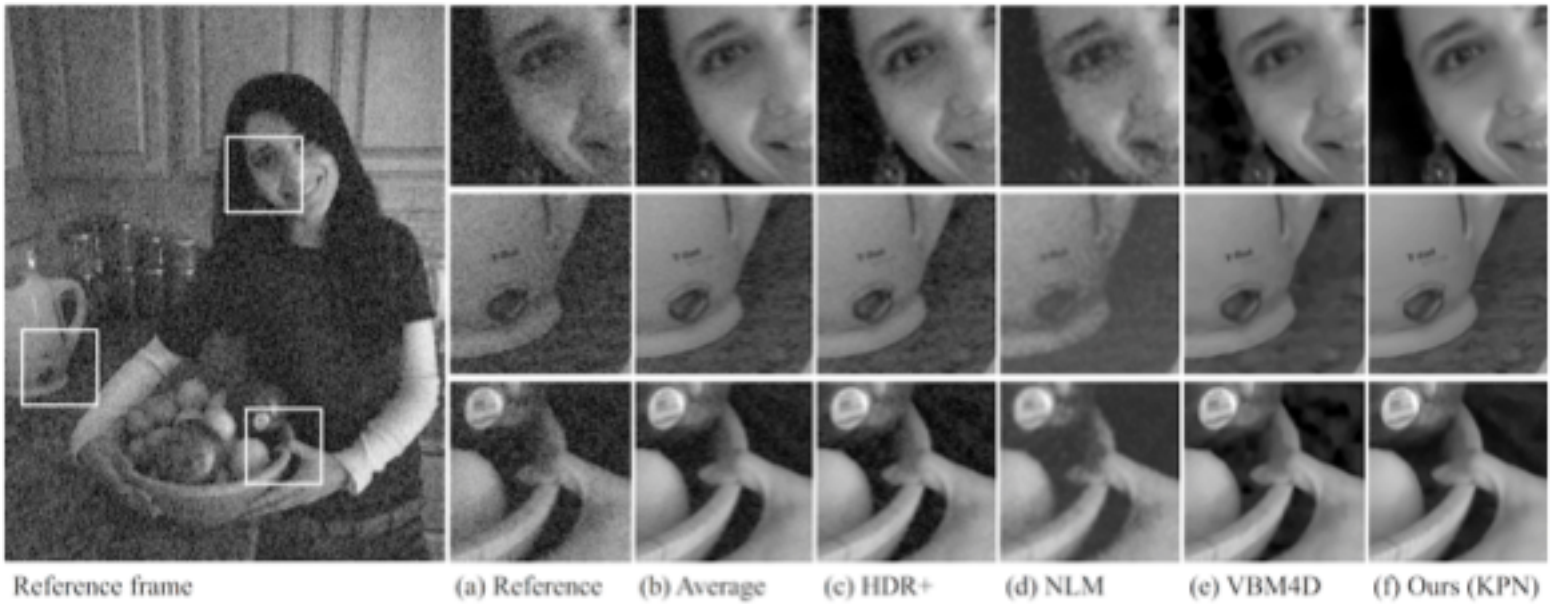


Image from http://people.eecs.berkeley.edu/~bmild/kpn/

Mildenhall, Ben et al. **"Burst Denoising with Kernel Prediction Networks."** *CoRR*abs/1712.02327 (2017)

DENOISING

Kernel Predicting Network (KPN) (2)

Aim: “produce **a single clean image from a noisy burst of N images** captured by a **handheld camera**”

Synthetize the dataset with global / local shift (from real images)

KPN: our architecture **“generates a stack of per-pixel filter kernels** that jointly **aligns, averages, and denoises** a burst to produce a clean version of the reference frame”.

Cost function including an **alignment terms**, vanishing during training.

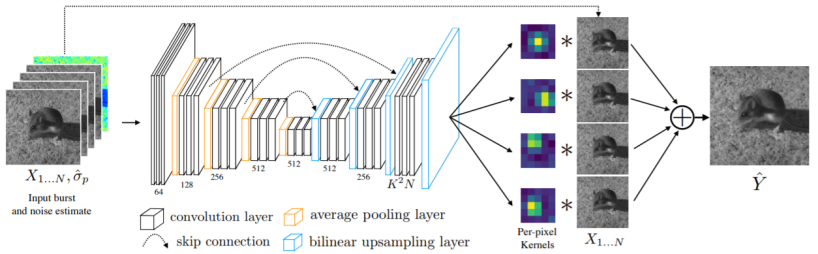


Image from https://arxiv.org/pdf/1712.02327.pdf

Mildenhall, Ben et al. “**Burst Denoising with Kernel Prediction Networks.**” *CoRR*abs/1712.02327 (2017)

DENOISING

Noise to noise (no need for clean data?) (1)

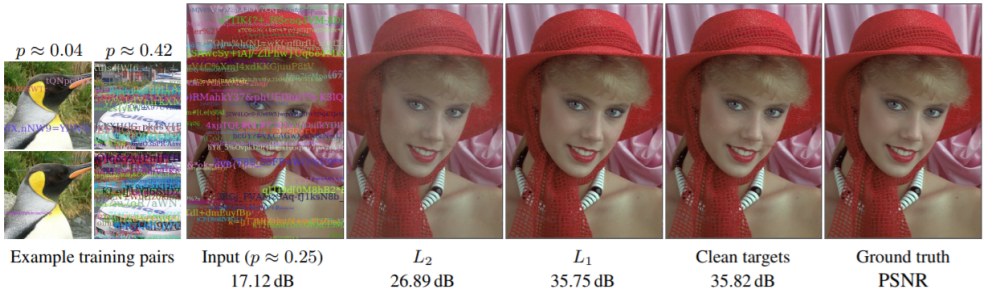


Figure 3. In case of text removal the mean (L_2 loss) is not the correct answer but median (L_1) is.

Figure 4. In case of random-valued impulse noise the mode (L_0) produces unbiased results, unlike mean (L_2) or median (L_1).

Image from https://arxiv.org/pdf/1803.04189.pdf

Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, Timo Aila, **"Noise2Noise: Learning Image Restoration without Clean Data"**, ICML 2018.

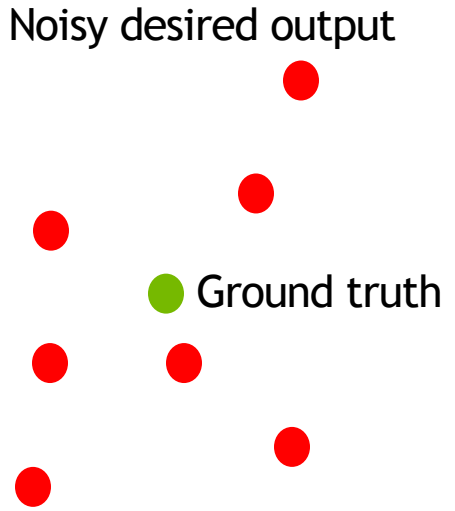
DENOISING

Noise to noise (no need for clean data?) (2)

Desired output = noisy image

Use the **proper cost function** (e.g. L2 for Gaussian noise), s.t. the expected value of the desired output is the ground truth

Et voilà! **Converge time** and image **quality comparable to** those achieved with **clean targets**.



Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, Timo Aila, **"Noise2Noise: Learning Image Restoration without Clean Data"**, ICML 2018.

DEMOSAICKING

(from RGGB Bayer to RBG images)

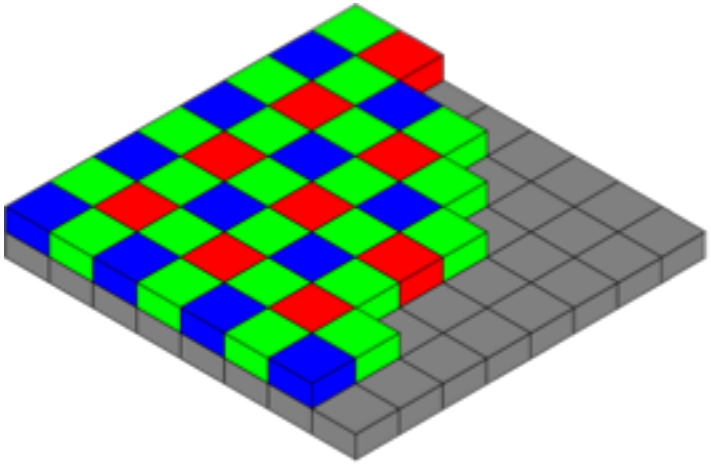
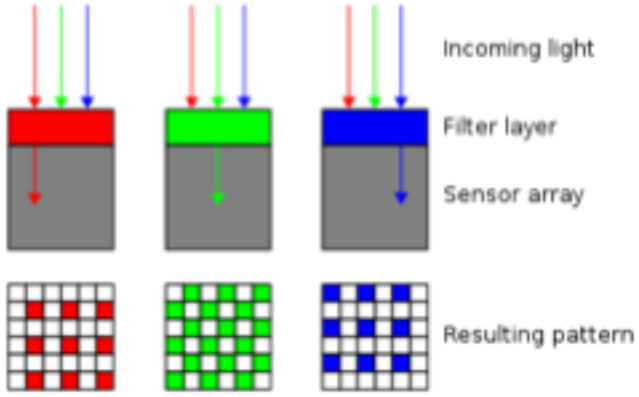


Image credit: Wikipedia



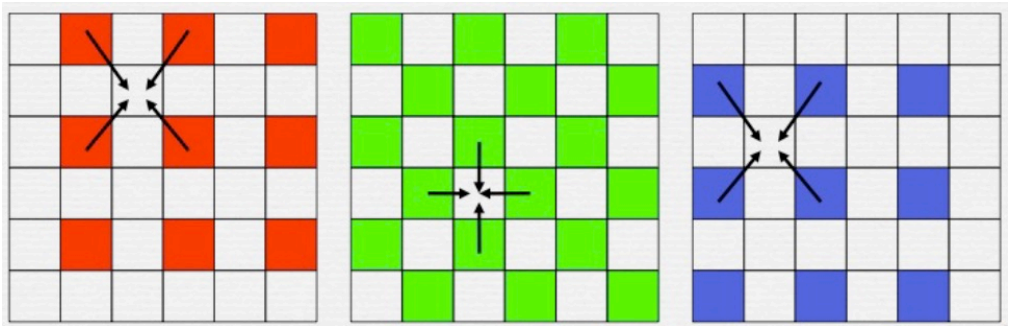
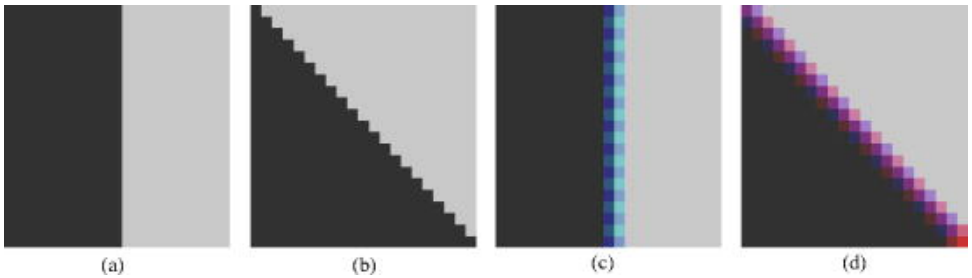


Image credit: Marc Levoy



Malvar et al., High-Quality Linear Interpolation for Demosaicing of Bayer-Patterned Color Images, 2004

DEMOSAICKING

ML for demosaicing (1)

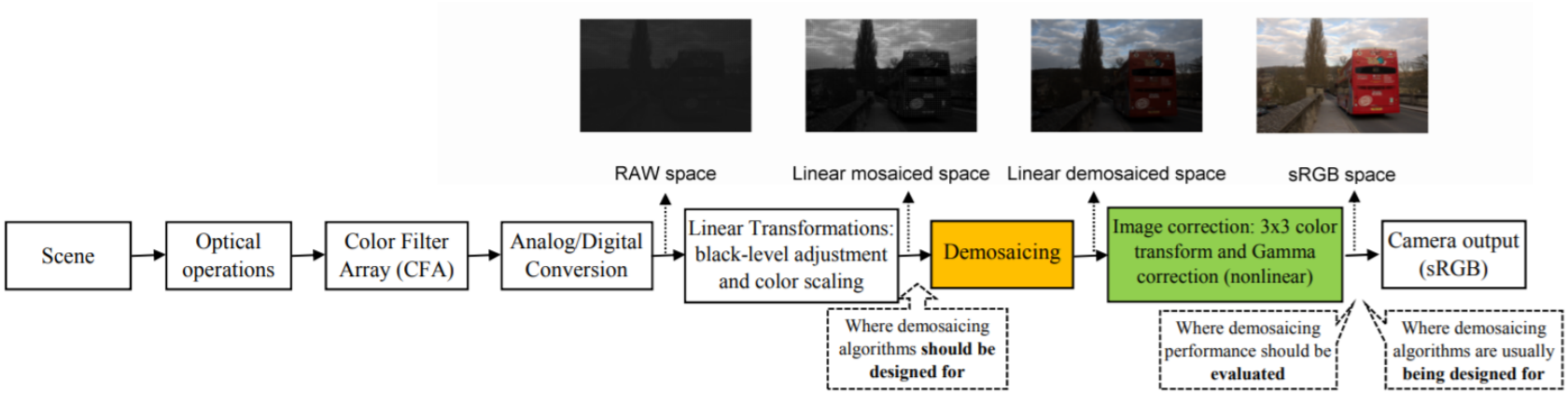


Fig. 1: A simplified camera pipeline. Many academic demosaicing algorithms work on fully developed sRGB images, masked by a CFA pattern. Instead, a practical method must use raw linear-space images as its input (orange block), since the 3x3 color transform to follow (green block) requires all missing measurements to have been filled in. Nonetheless, one should aim at *optimizing* the quality of the output in sRGB space, where images are fully developed and ready to be viewed by a human.

Image https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/khashabi2014demosaicing.pdf

Daniel Khashabi, Sebastian Nowozin, Jeremy Jancsary,Andrew W. Fitzgibbon, **"Joint Demosaicing and Denoising via Learned Non-parametric Random Fields"**, 2013.

DEMOSAICKING

ML for demosaicing (2)

"… two challenges to overcome by a demosaicing method: first, it needs to model and respect the statistics of natural images **in order to reconstruct natural looking images**; second, it needs to be able to **perform well in the presence of noise.**"

Introduce **a large dataset** for learning demosaicking and denoising

PSRN or SSIM can be optimized (more on this later!)

Overcome SOA, with less engineering effort

End-to-end camera pipeline training

Train Regression Tree Fields (ML, not DL)

Daniel Khashabi, Sebastian Nowozin, Jeremy Jancsary,Andrew W. Fitzgibbon, "**Joint Demosaicing and Denoising via Learned Non-parametric Random Fields**", 2013.

DEMOSAICKING

Patch selection improves results (1)

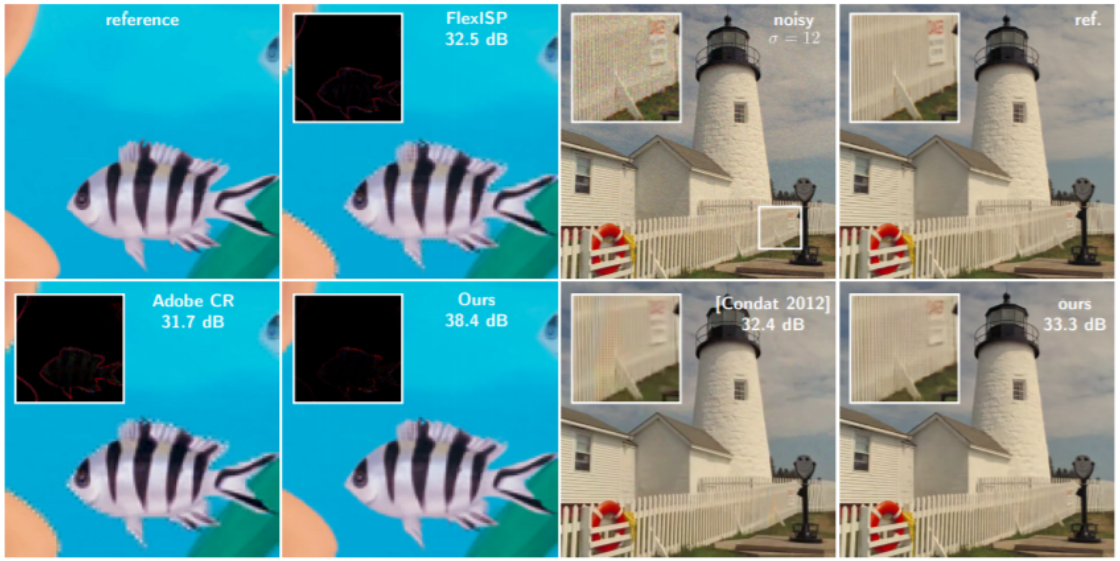


Figure 1: *We propose a data-driven approach for jointly solving denoising and demosaicking. By carefully designing a dataset made of rare but challenging image features, we train a neural network that outperforms both the state-of-the-art and commercial solutions on demosaicking alone (group of images on the left, insets show error maps), and on joint denoising–demosaicking (on the right, insets show close-ups). The benefit of our method is most noticeable on difficult image structures that lead to* moiré *or* zippering *of the edges.*

Image from https://groups.csail.mit.edu/graphics/demosaicnet/data/demosaic.pdf

Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand, "**Deep joint demosaicking and denoising**", ACM Trans. Graph., 2016

DEMOSAICKING

Patch selection improves results (2)

Joint demosaicking and denoising

"To create a better training set, we present **metrics to identify difficult patches** and techniques for mining community photographs for such patches."

Train (all data), find difficult patches, re-train (weighted loss function).

Convolutional architecture with **additional input for the noise level** and **skip connections.**

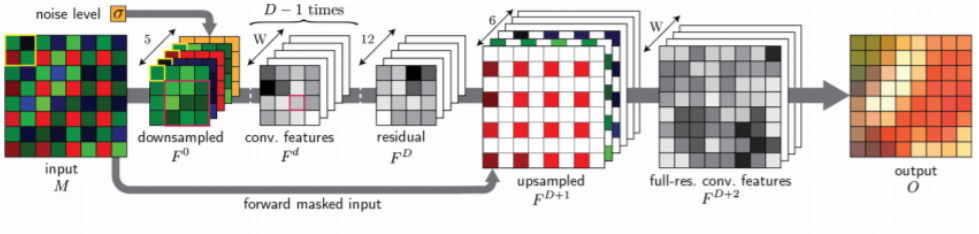


Figure 2: *Our proposed architecture. The first layer of the network packs 2 × 2 blocks in the Bayer image into a 4D vector to restore translation invariance and speed up the processing. We augment each vector with the noise parameter σ to form 5D vectors. Then, a series of convolutional layers filter the image to interpolate the missing color values. We finally unpack the 12 color samples back to the original pixel grid and concatenate a masked copy of the input mosaick. We perform a last group of convolutions at full resolution this time to produce the final features. We linearly combine them to produce the demosaicked output.*

Image from https://groups.csail.mit.edu/graphics/demosaicnet/data/demosaic.pdf

DEMOSAICKING

Using GANs

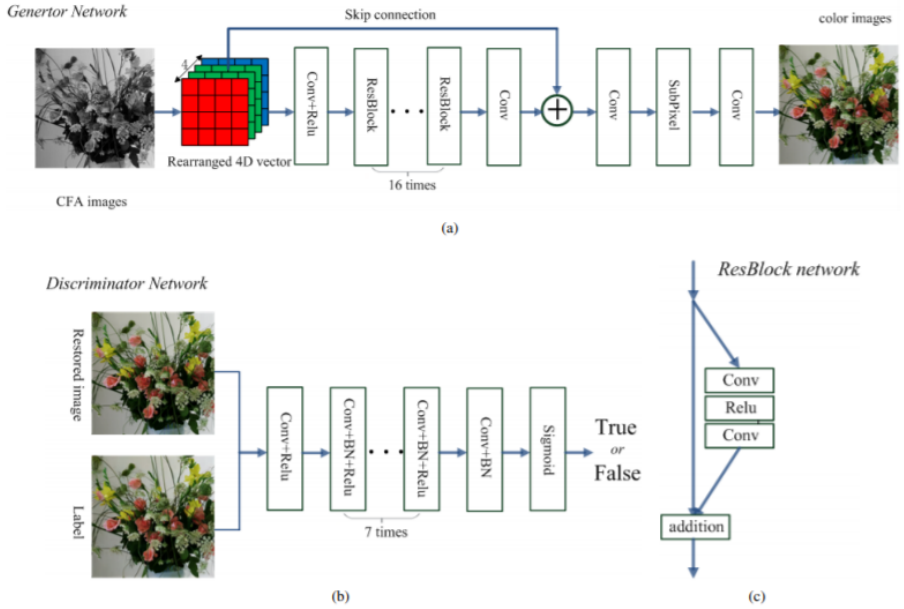


Fig. 2: The architecture of our Generative adversarial networks for joint demosaicing and denoise. The top is the generator network structure. The lower left corner is the discriminator network structure. The bottom right is the structure of the residual block

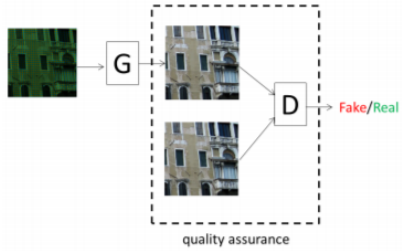


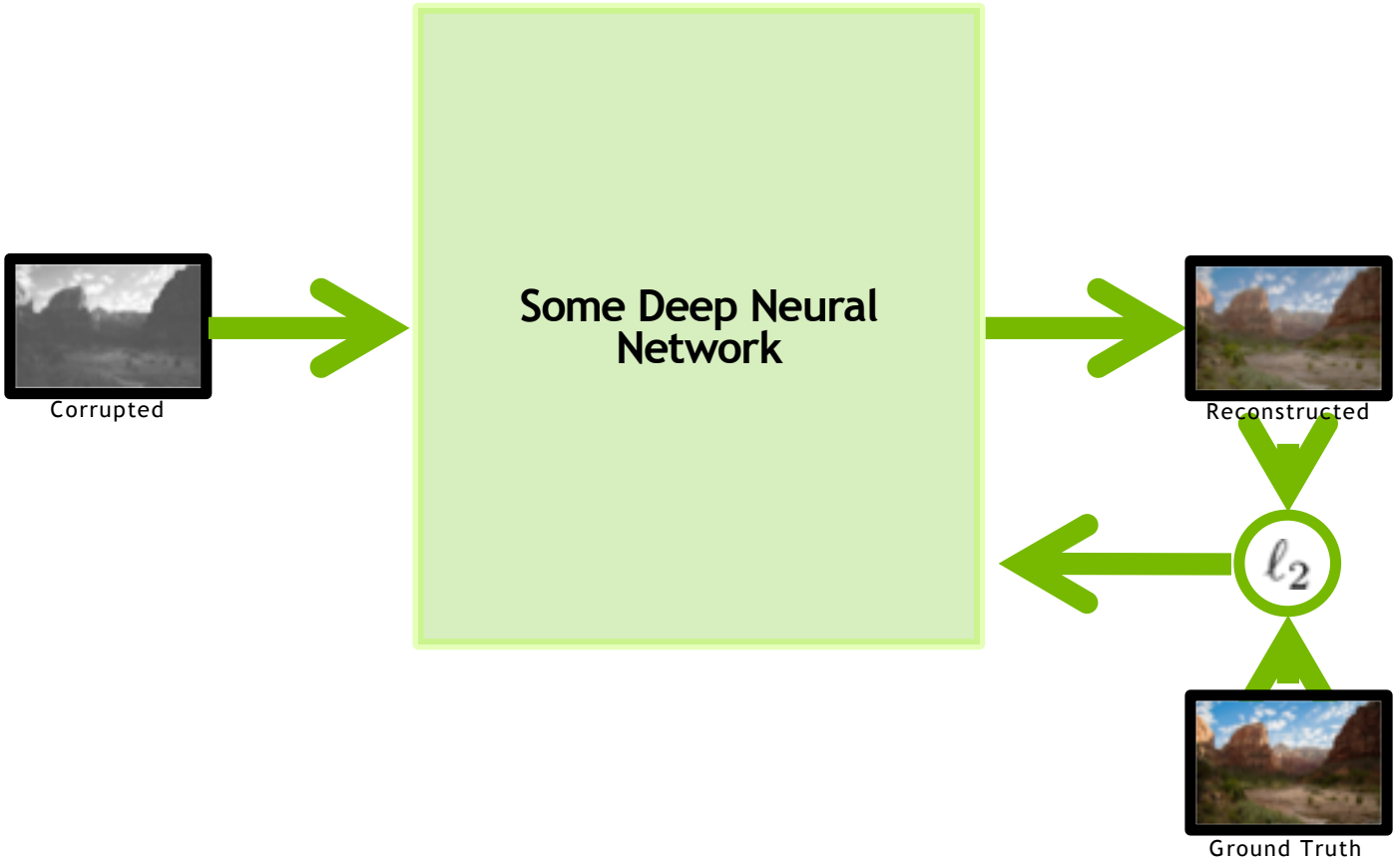
Fig. 1: Introducing GAN as a strategy of quality assurance in JDD.

Images from https://arxiv.org/pdf/1802.04723.pdf

Weisheng Dong, Ming Yuan, Xin Li, Guangming Shi, **"Joint Demosaicing and Denoising with Perceptual Optimization on a Generative Adversarial Network"**, 2018.

LOSS FUNCTIONS

LOSS FUNCTIONS FOR IMAGE PROCESSING



MEASURING IMAGE QUALITY

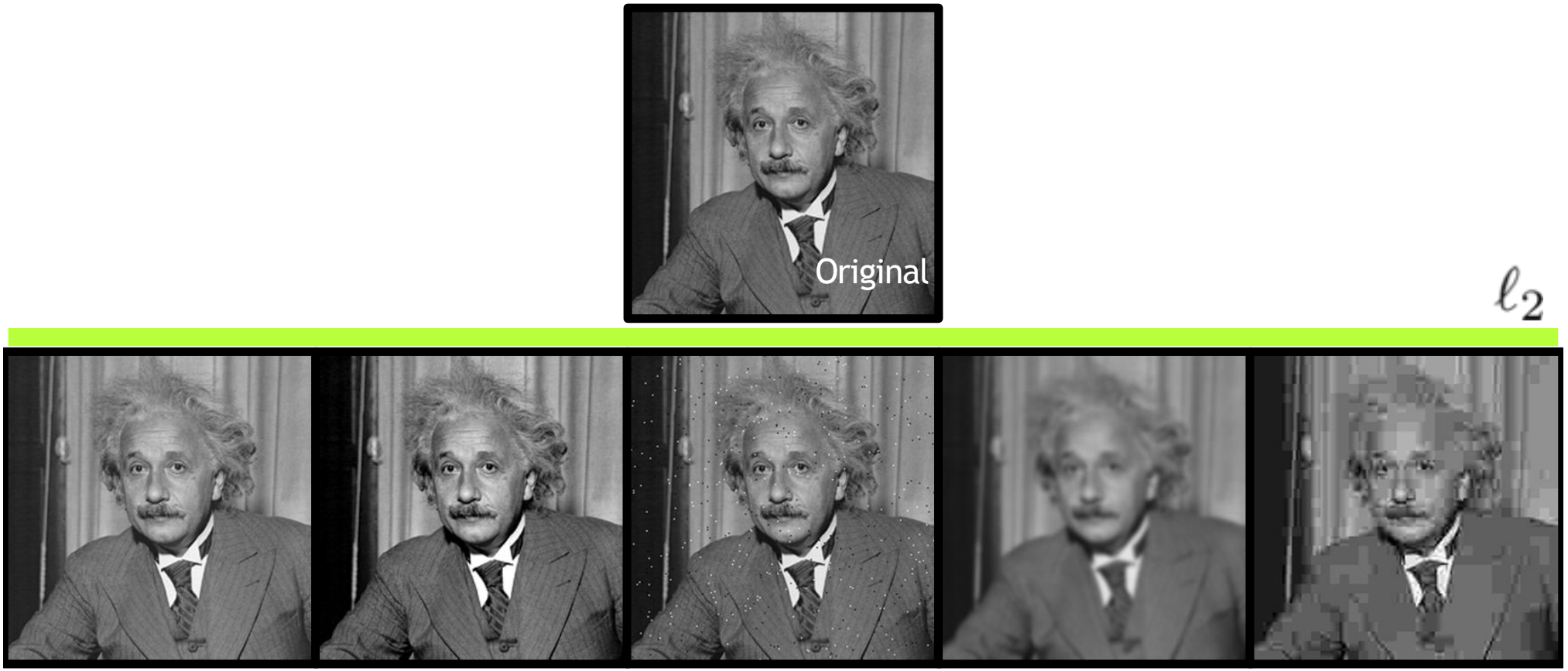


Image adapted from https://ece.uwaterloo.ca/~z70wang/research/ssim/

MEASURING IMAGE QUALITY

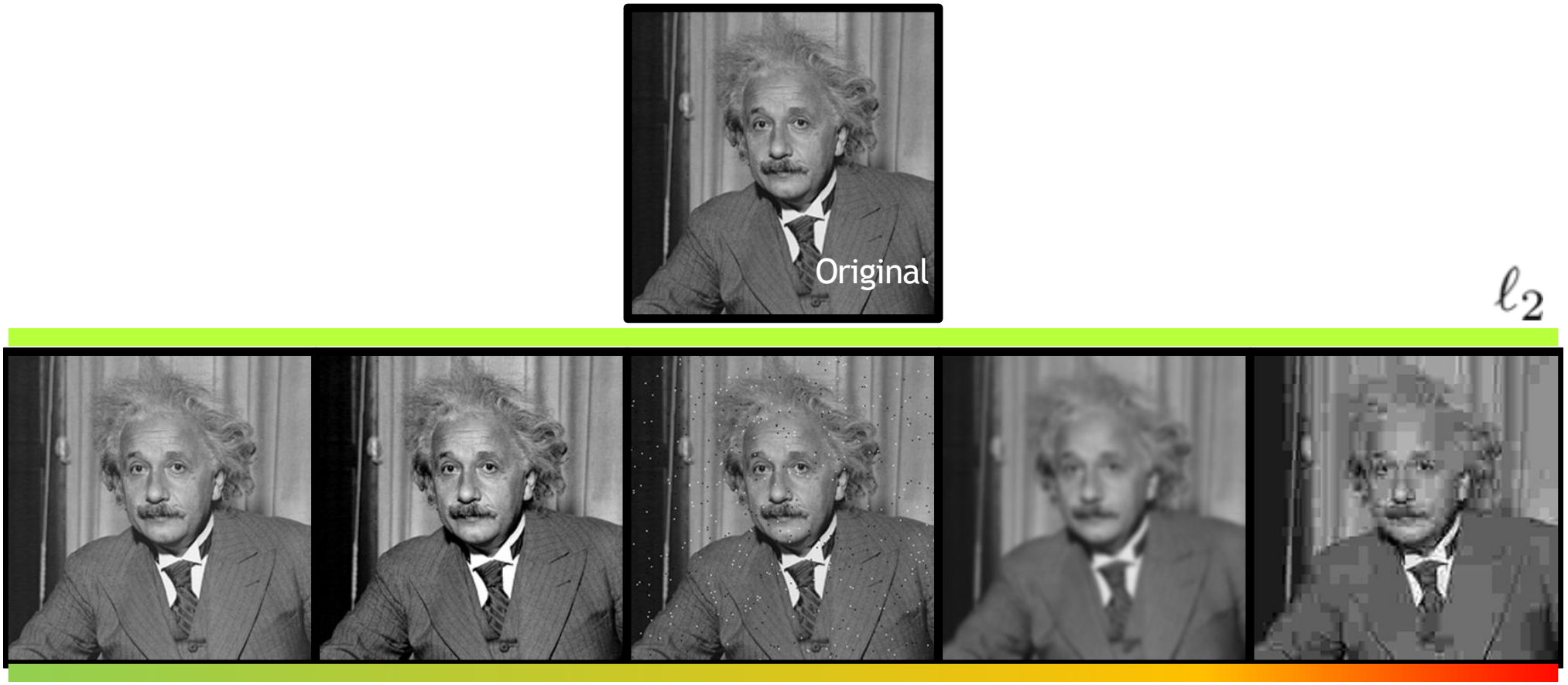


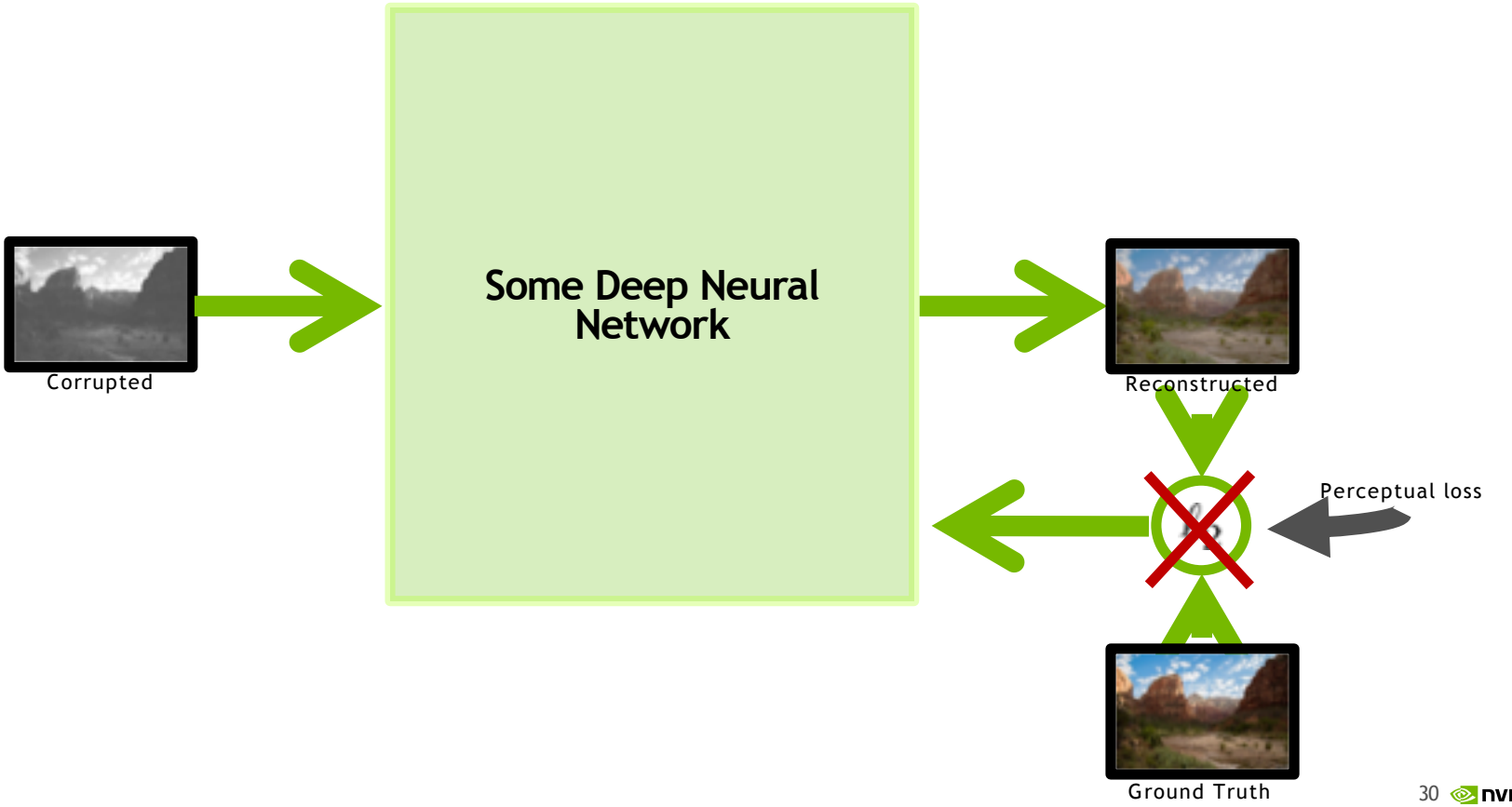
Image adapted from https://ece.uwaterloo.ca/~z70wang/research/ssim/

MEASURING IMAGE QUALITY

$$\ell_2(p) = \sqrt{I_1^2(p) - I_2^2(p)}$$

$$\mathrm{SSIM}(I_1, I_2) = l(I_1, I_2) \cdot c(I_1, I_2) \cdot s(I_1, I_2)$$

LOSS FUNCTIONS FOR IMAGE PROCESSING



MEASURING IMAGE QUALITY

$$\ell_1(p) = |I_1(p) - I_2(p)|$$

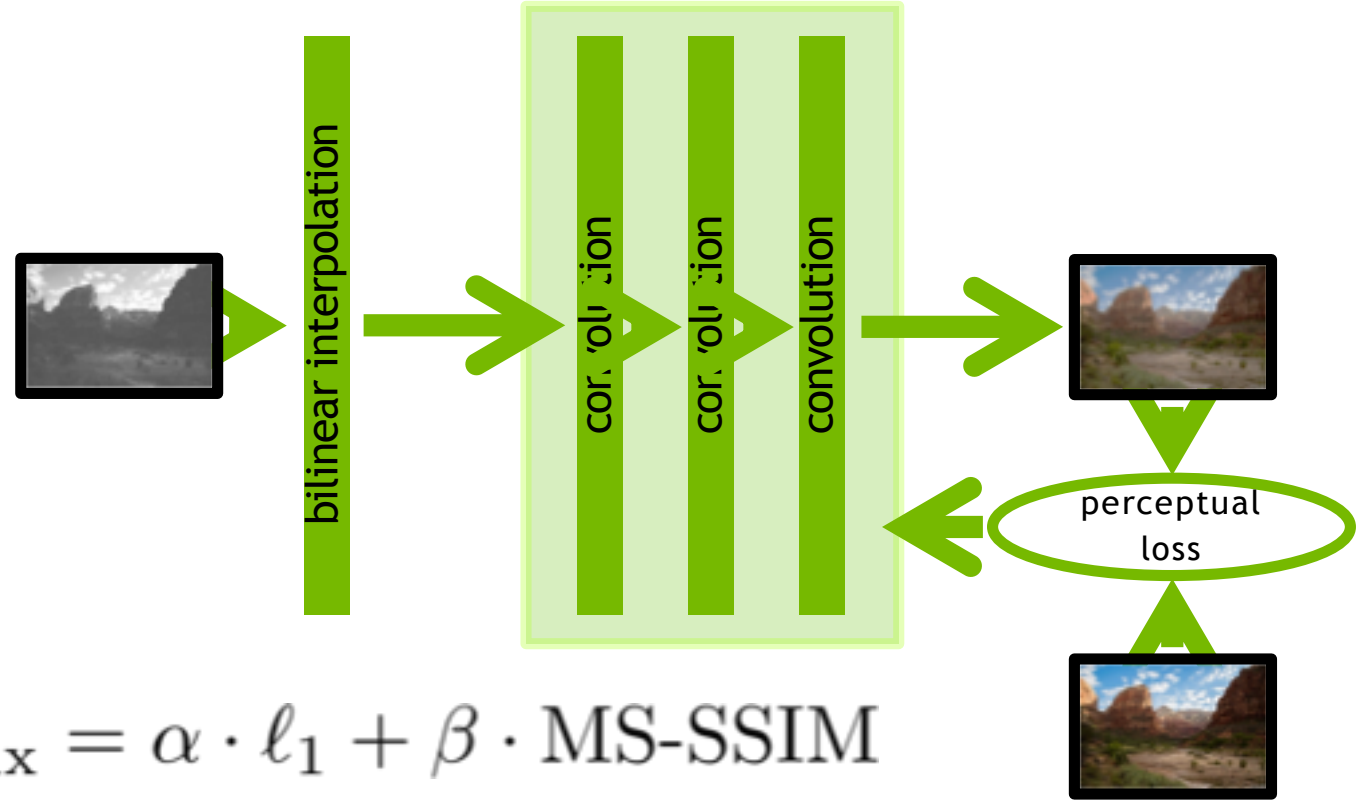
$$\ell_2(p) = \sqrt{I_1^2(p) - I_2^2(p)}$$

$$\mathrm{SSIM}(I_1, I_2) = l(I_1, I_2) \cdot c(I_1, I_2) \cdot s(I_1, I_2)$$

$$\text{MS-SSIM}(I_1, I_2) = \mathrm{Multiscale}(\mathrm{SSIM}(I_1, I_2))$$

JOINT DEMOSAICKING AND DENOISING

Network architecture



$$\mathcal{L}_{\text{Mix}} = \alpha \cdot \ell_1 + \beta \cdot \text{MS-SSIM}$$

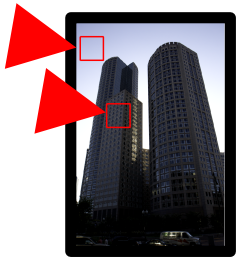
Zhao et al., “Loss Functions for Image Restoration With Neural Networks,” IEEE TIP, 2017

Ground truth

Noisy

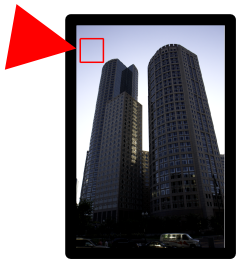
ℓ_2

$\mathcal{L}_{\text{Mix}}$



RESULTS

Visual comparison (+ unsharp masking)

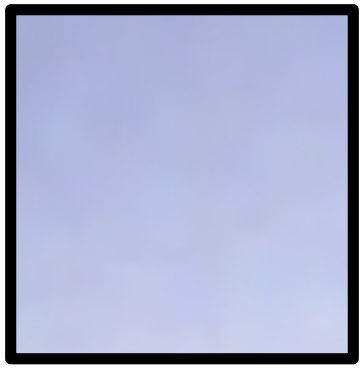


RESULTS

Why mixing MS-SSIM and ℓ_1?

Clean

MS-SSIM



MS-SSIM + ℓ_1

DO WE NEED HANDCRAFTED LOSSES?

Gatys et al., “A Neural Algorithm of Artistic Style,” Arxiv 2015

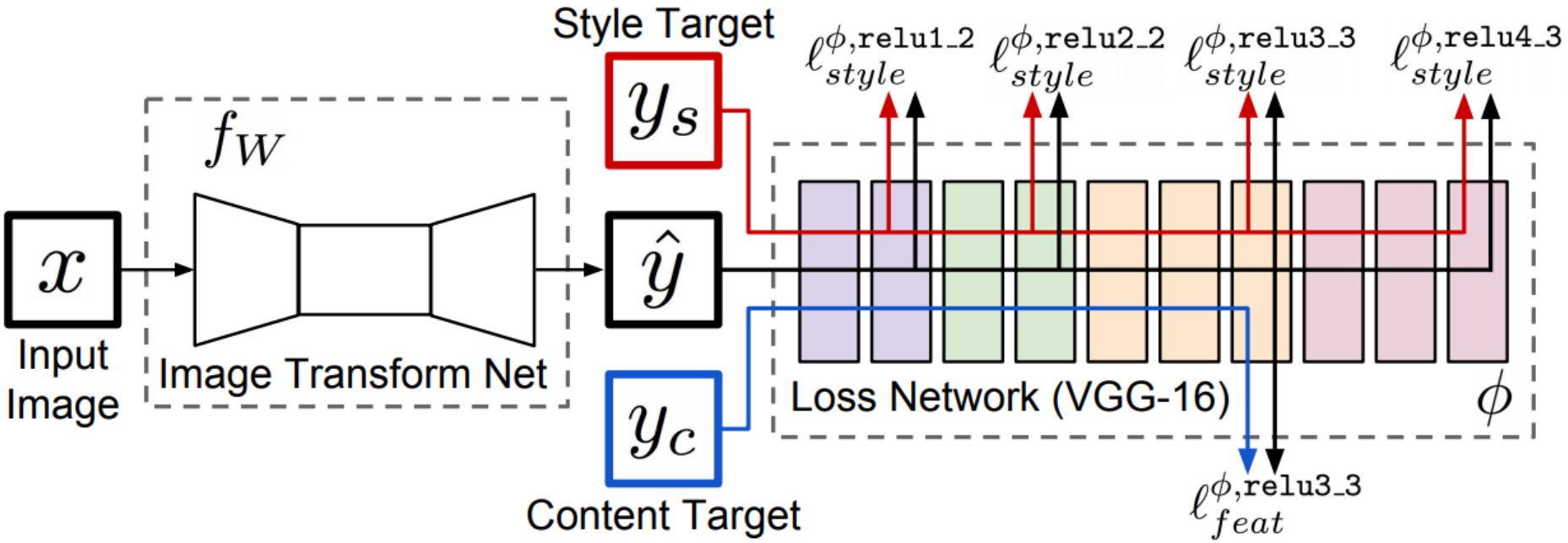
GATYS LOSS

$$\mathcal{L}_{content}(l) \propto \sum_{ij} \left(F_{ij}^{l} - P_{ij}^{l} \right)^2$$

$$\mathcal{L}_{style}(l) \propto \sum_{ij} \left(G_{ij}^{l} - A_{ij}^{l} \right)^2$$

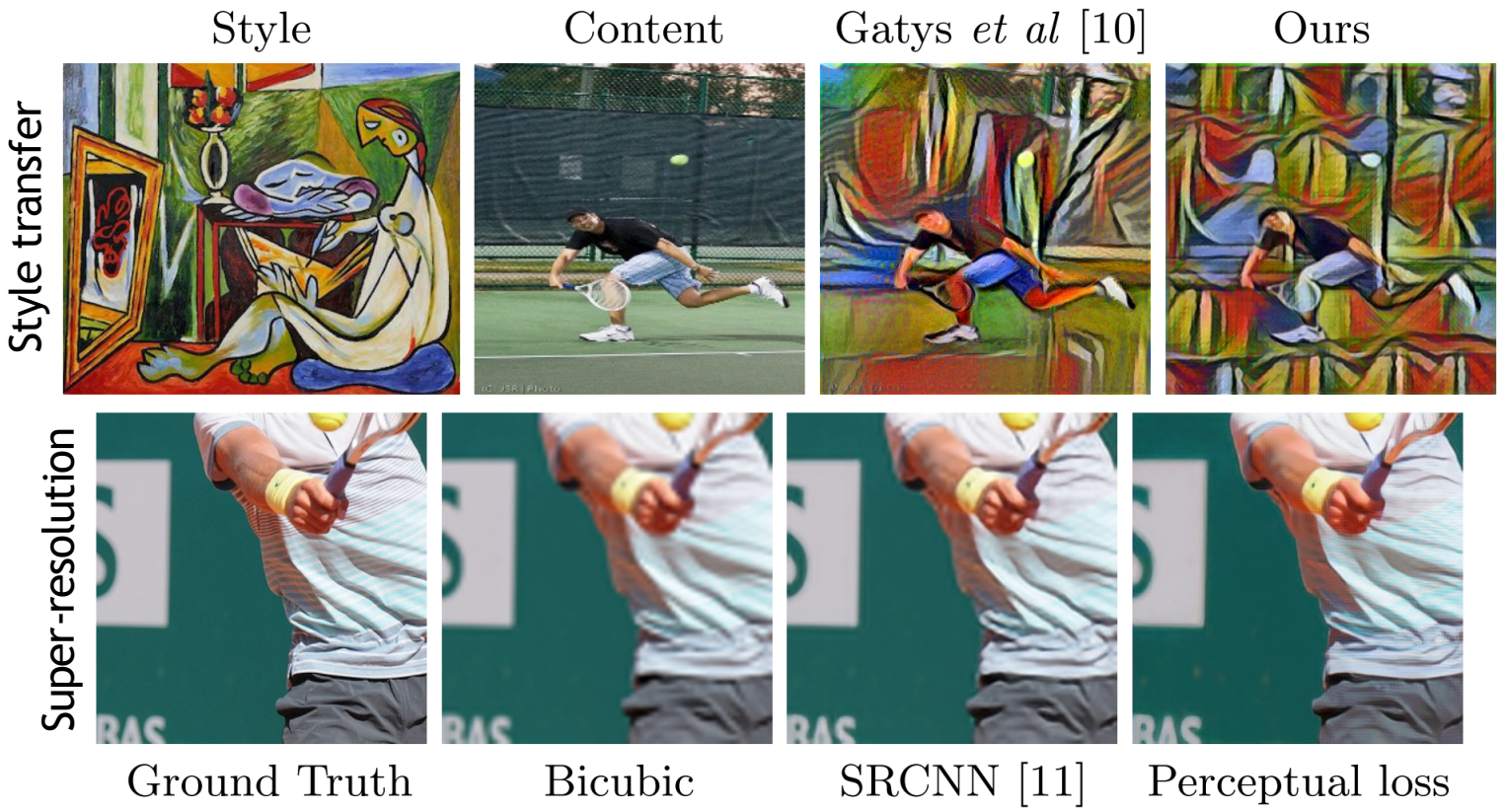
Gatys et al., “A Neural Algorithm of Artistic Style,” Arxiv 2015

PERCEPTUAL LOSS



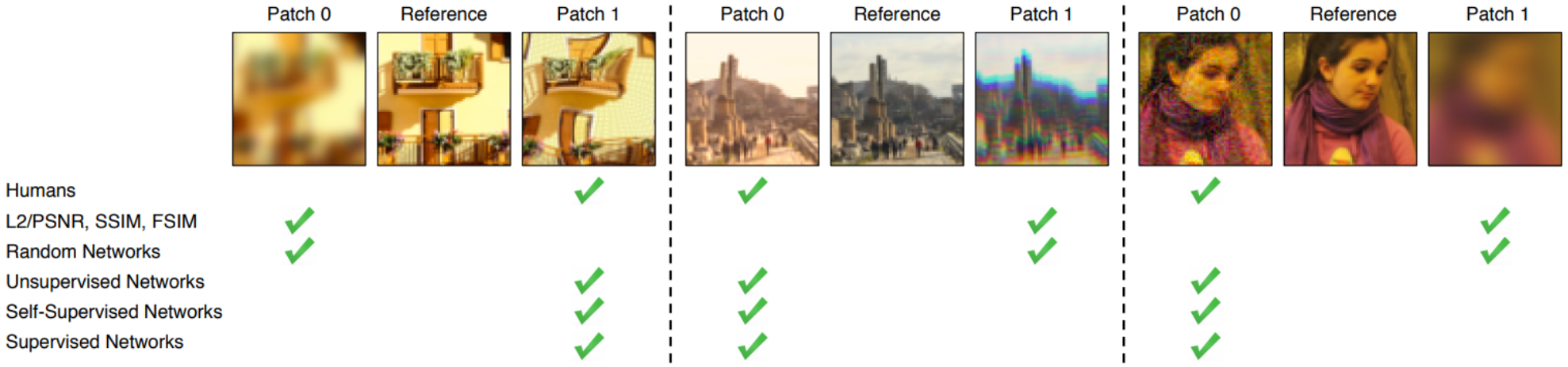
Johnson et al., “Perceptual Losses for Real-Time Style Transfer and Super-Resolution,” ECCV 2016

PERCEPTUAL LOSS



Johnson et al., "Perceptual Losses for Real-Time Style Transfer and Super-Resolution," ECCV 2016

FEATURES VS HANDCRAFTED METRICS



Zhang et al., “The Unreasonable Effectiveness of Deep Features as a Perceptual Metric,” IEEE CVPR 2018

GENERATIVE ADVERSARIAL NETWORKS

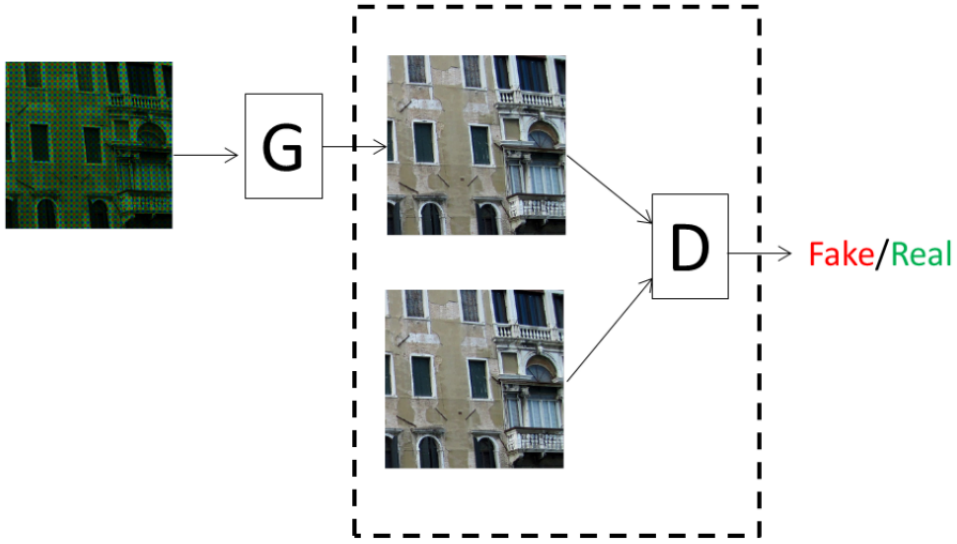


Image from: Dong et al., "Joint Demosaicing and Denoising with Perceptual Optimization on a Generative Adversarial Network," Arxiv 2018

GENERATIVE ADVERSARIAL NETWORKS

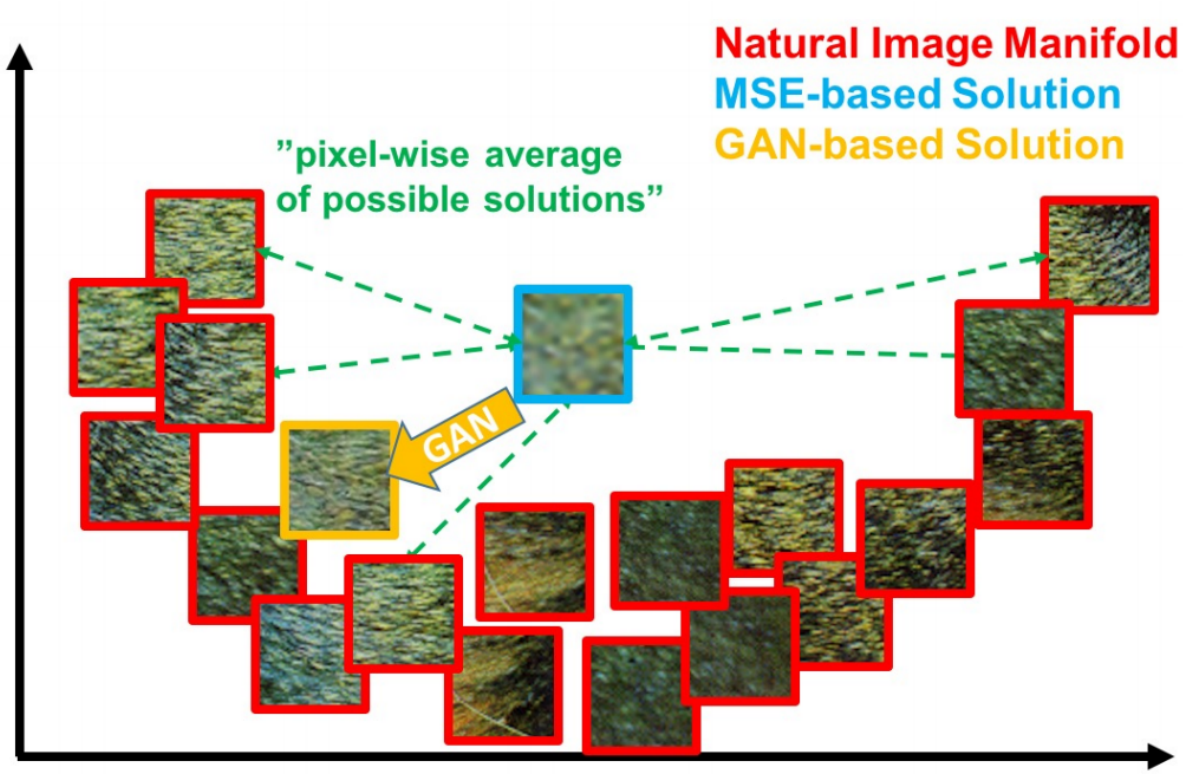


Image from: Ledig et al., "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," CVPR 2017

DEMOSAICKING

Using GANs

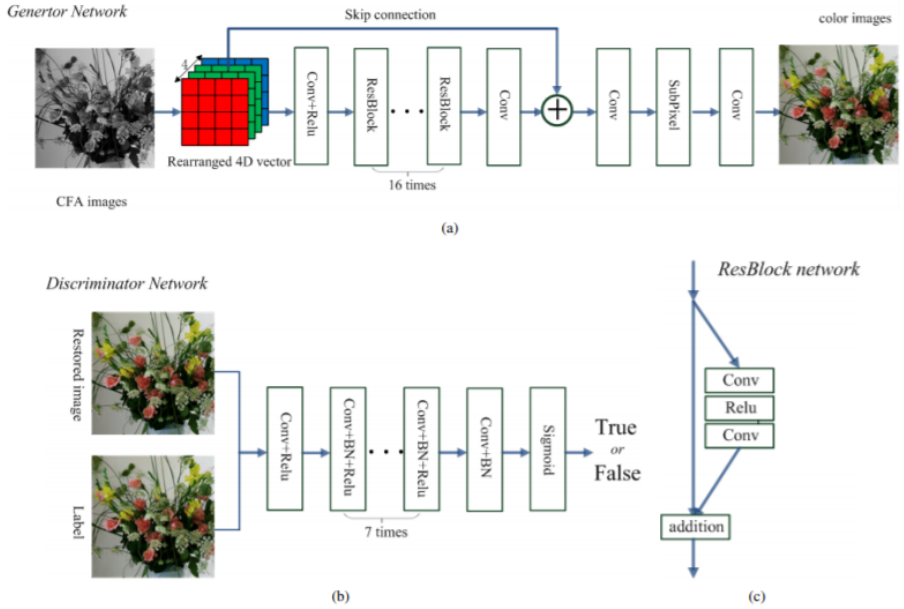


Fig. 2: The architecture of our Generative adversarial networks for joint demosaicing and denoise. The top is the generator network structure. The lower left corner is the discriminator network structure. The bottom right is the structure of the residual block

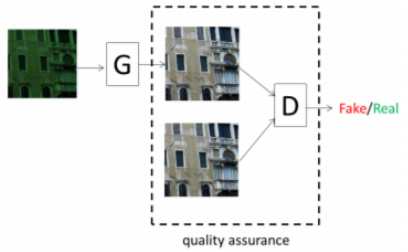


Fig. 1: Introducing GAN as a strategy of quality assurance in JDD.

Images from https://arxiv.org/pdf/1802.04723.pdf

Weisheng Dong, Ming Yuan, Xin Li, Guangming Shi, **"Joint Demosaicing and Denoising with Perceptual Optimization on a Generative Adversarial Network"**, 2018.

SUPER-RESOLUTION WITH GANS

bicubic
(21.59dB/0.6423)

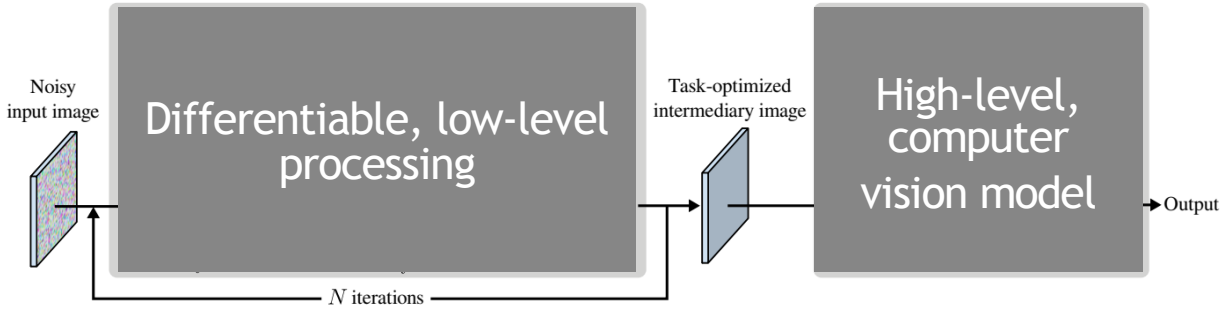
SRResNet
(23.53dB/0.7832)

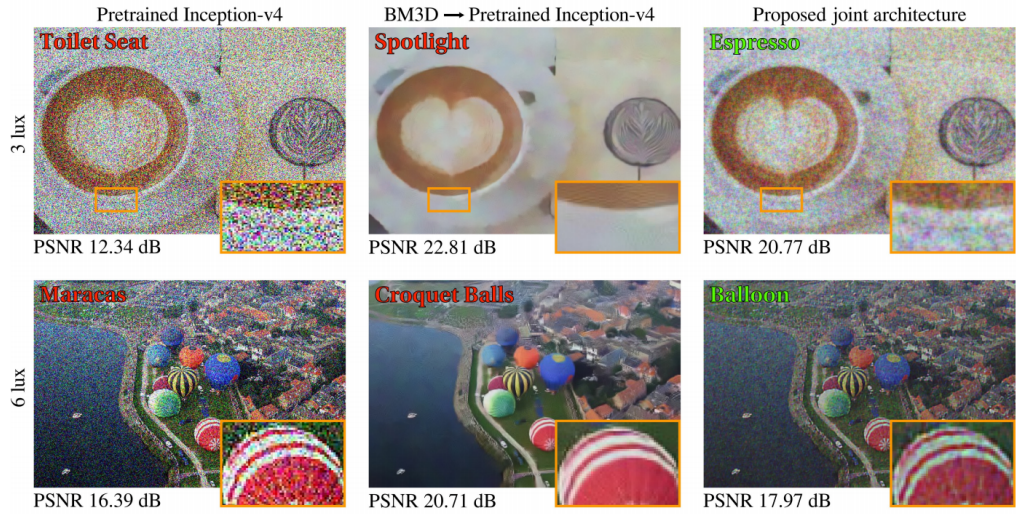
SRGAN
(21.15dB/0.6868)

original

Ledig et al., “Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network,” CVPR 2017

TASK-SPECIFIC PROCESSING





Diamond et al., "Dirty Pixels: Optimizing Image Classification Architectures for Raw Sensor Data," Arxiv 2017

MULTIPLE IMAGES

DL FOR STEREO

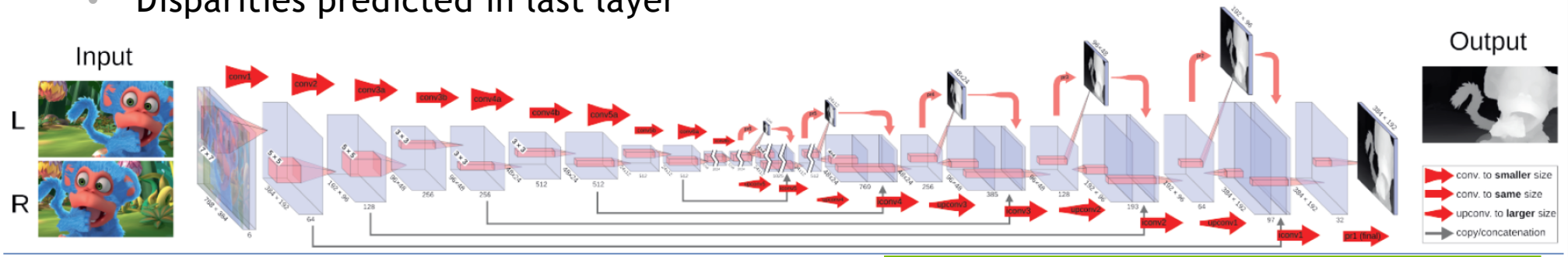
Computing disparity maps using 2D convolutions

Use 2D convolutions:

- Features extracted from images
- Features are cross-correlated
- Hourglass network
  with skip connections
- Disparities predicted in last layer

Example methods:

- DispNet
  [Mayer et al. 2015]
- Cascade Residual Learning (CRL)
  [Pang et al. 2017]



[from https://www.slideshare.net/yuhuang/optic-flow-estimation-with-deep-learning]

DL FOR STEREO

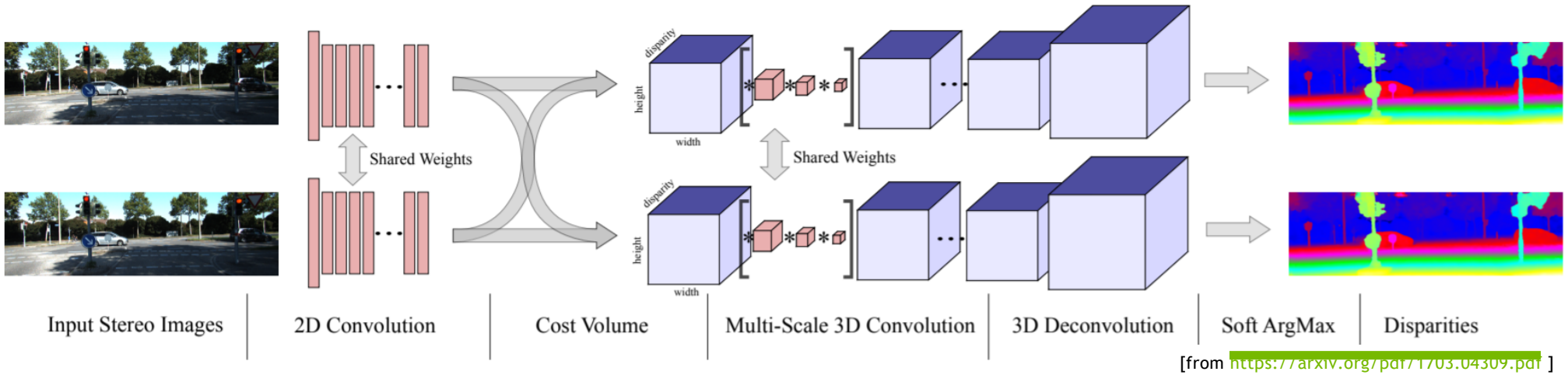
Computing disparity maps using 3D convolutions

Use 3D convolutions:

- Better results
- More computation (slower)

Example methods:

- Geometry and Context (GC-Net)
  [Kendall et al. 2017]
- Pyramid Stereo Matching (PSM-Net)
  [Chang and Chen 2018]



[from https://arxiv.org/pdf/1703.04309.pdf]

DL FOR OPTICAL FLOW

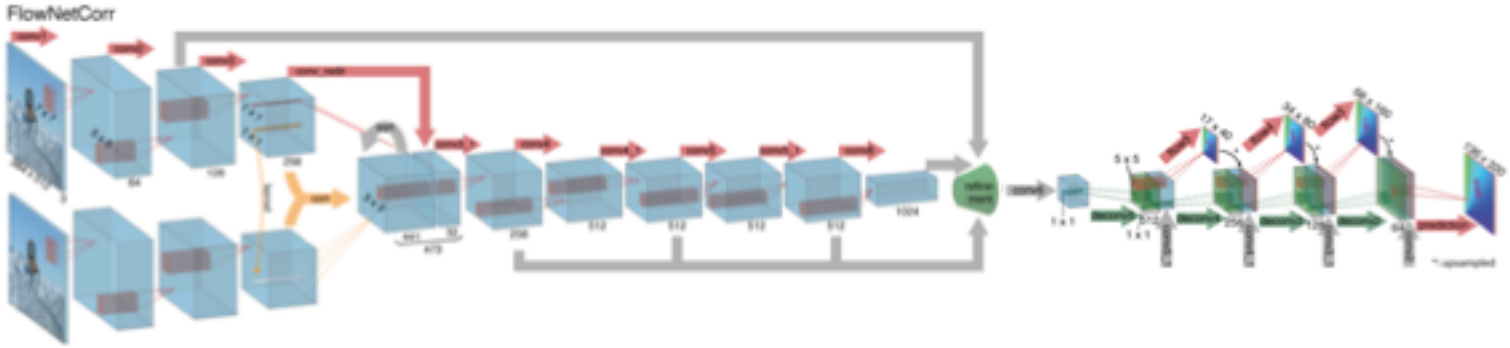
Computing optical flow using 2D convolutions

Use 2D convolutions:

- Features extracted from images
- Features are cross-correlated
- Hourglass network with skip connections
- Disparities predicted in last layer

Example methods:

- FlowNet / FlowNet2
  [Fischer et al. 2015] [Ilg et al. 2017]
- PWC-Net (multiscale/warping)
  [Sun et al. 2018]

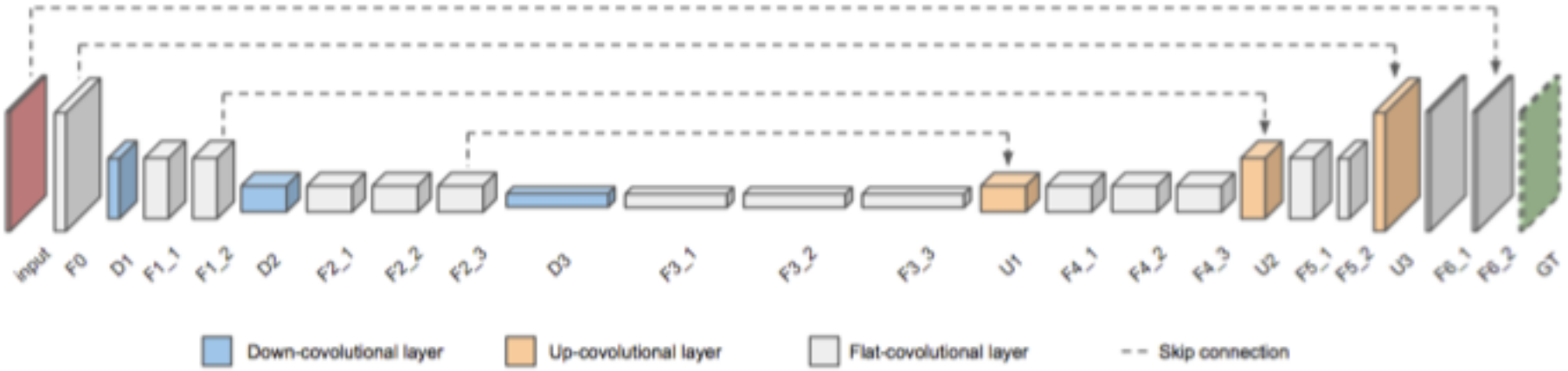


DEEP VIDEO DEBLURRING

[Su, et al., CVPR 2017]

DNNs learn to handle deblurring challenges implicitly

- Unknown spatially-varying blur kernel
- Frame-to-frame mis-alignment
- Simple U-Net with skip connections + L2-loss



DEBLURGAN

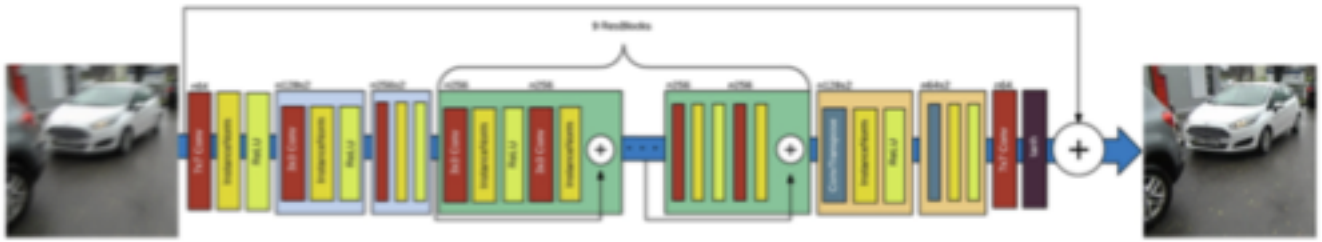
[Kupyn, et al., CVPR 2018]

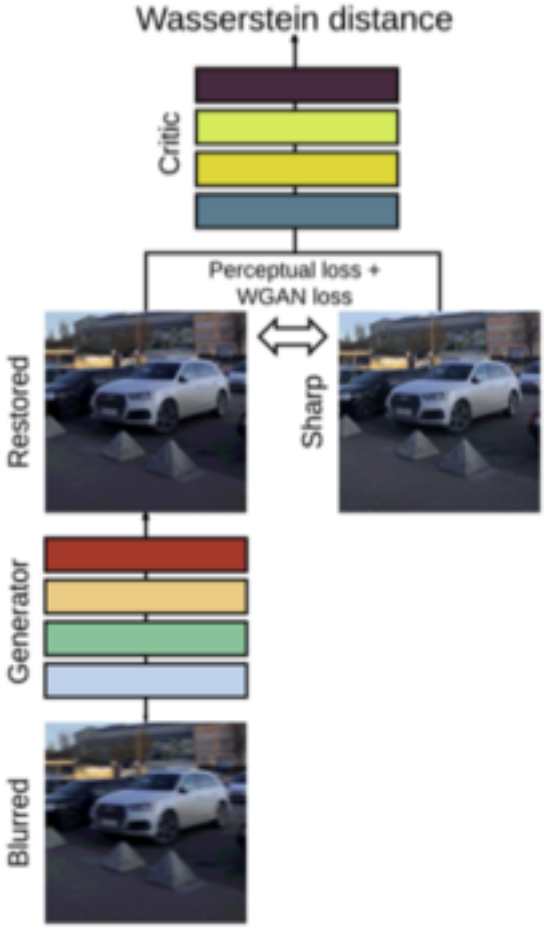
Simple ResNet architecture

More perceptually-motivated loss function

$$\mathcal{L} = \underbrace{\mathcal{L}_{GAN}}_{adv\ loss} + \underbrace{\lambda \cdot \mathcal{L}_X}_{content\ loss}$$

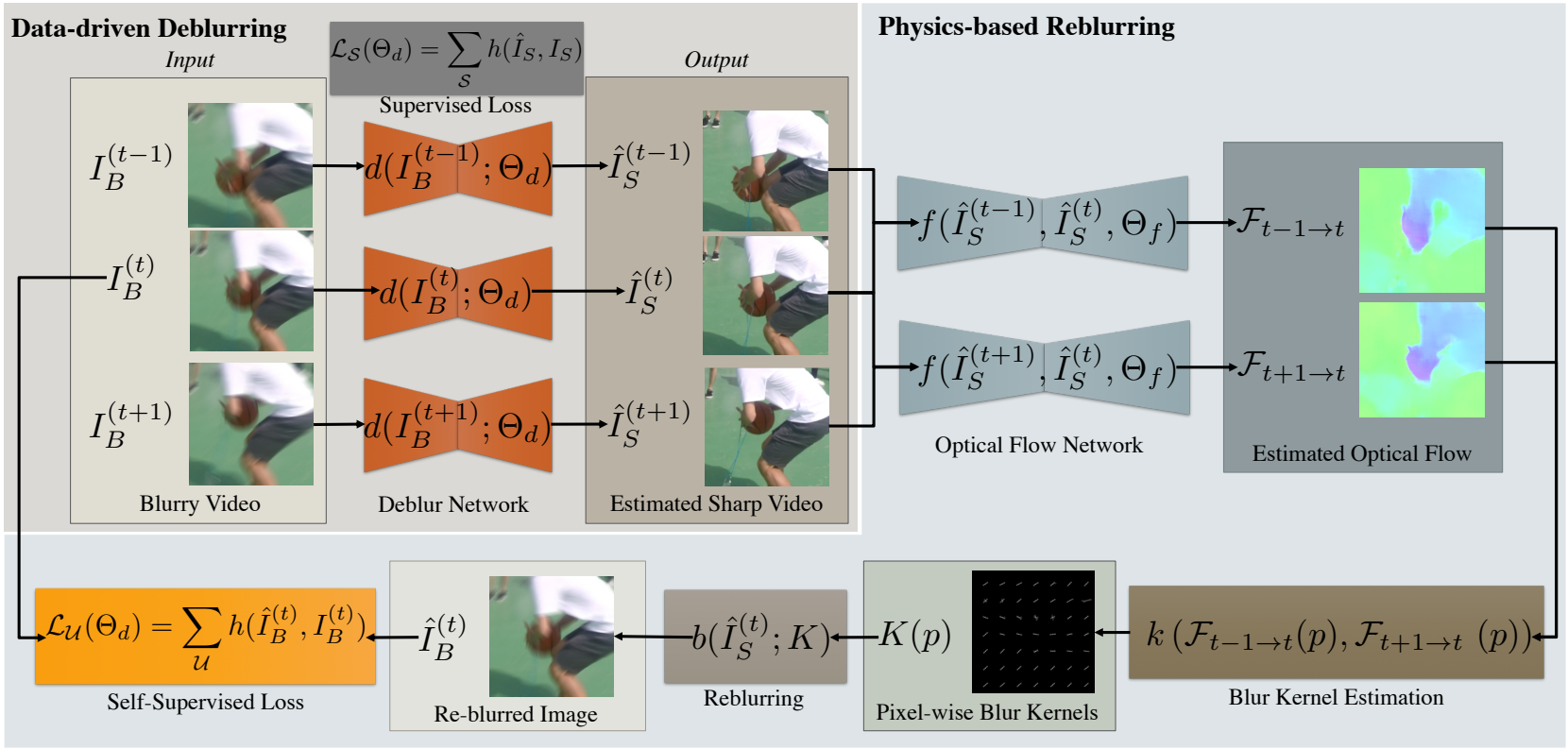
$$total\ loss$$





REBLUR2DEBLUR

[Chen, et al., ICCP 2018]



Input

Ground Truth

Deep Video Deblurring

DeblurGAN

Reblur2Deblur

Input

Ground Truth

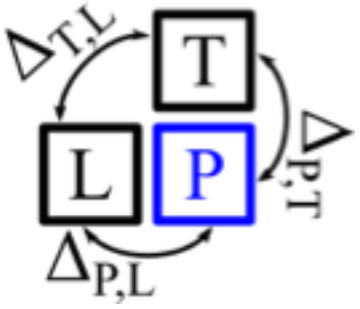
Deep Video Deblurring

DeblurGAN

Reblur2Deblur

OTHER SENSORS

BINARY GRADIENT CAMERAS



Gottardi et al., "A 100 μW 128 × 64 pixels contrast-based asynchronous binary vision sensor for sensor networks applications. IEEE Journal of Solid-State Circuits, 2009

BINARY GRADIENT CAMERAS

Jayasuriya et al., “Reconstructing Intensity Images from Binary Spatial Gradient Cameras,” IEEE CVPRW, ‘17

TIME-OF-FLIGHT CAMERAS

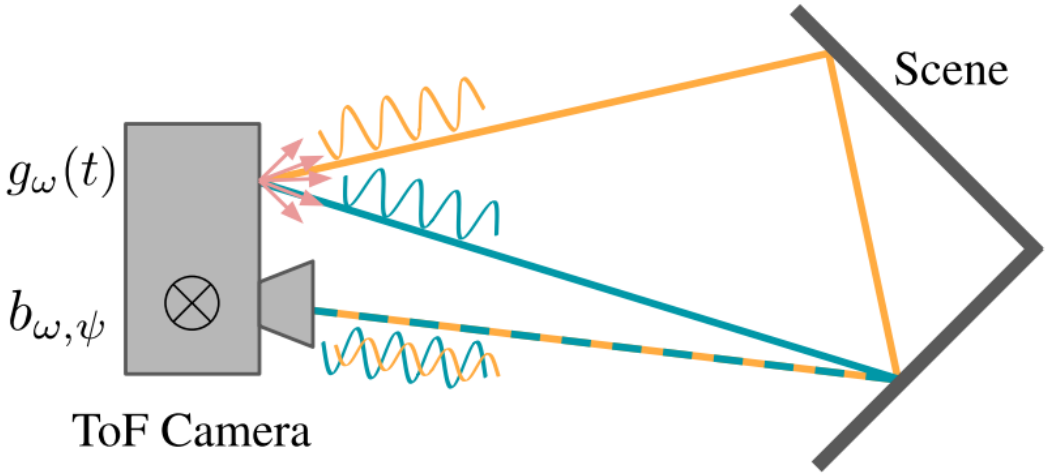
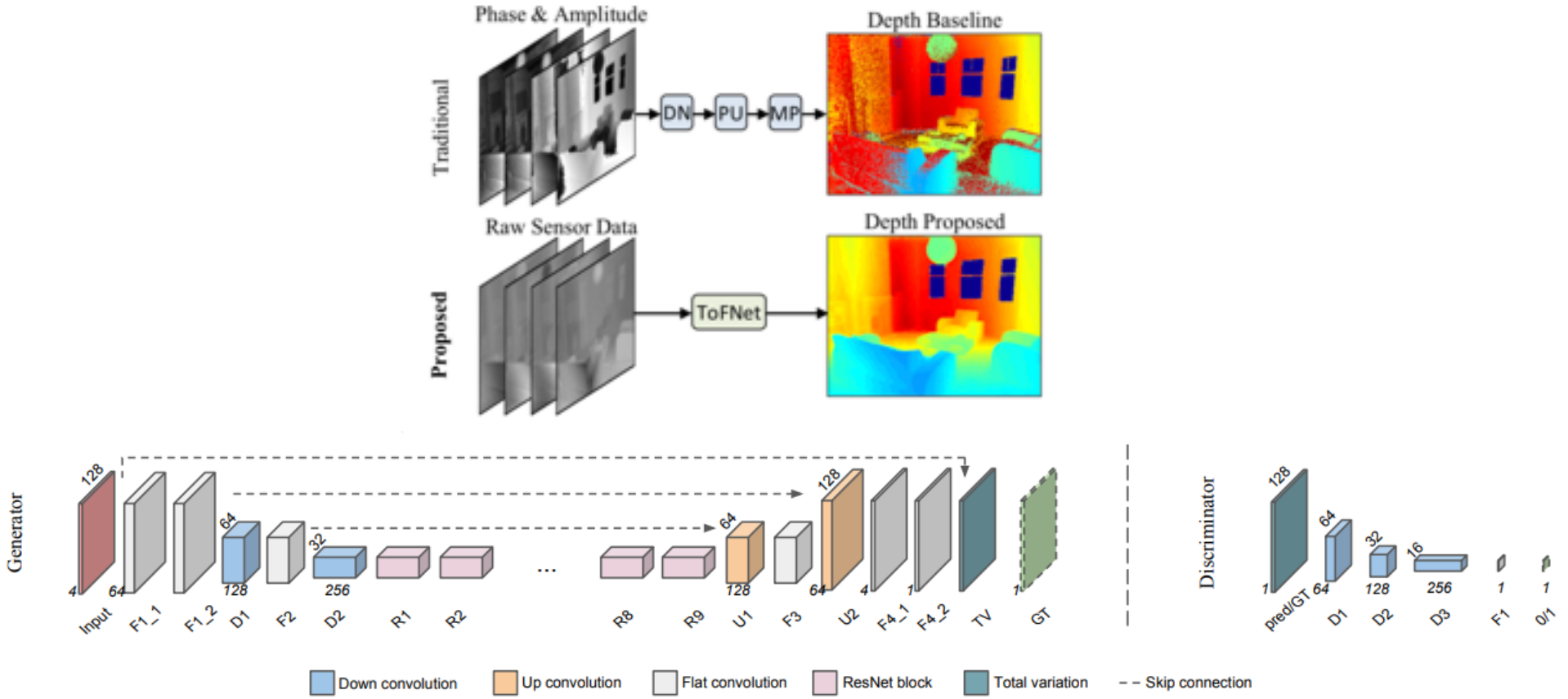


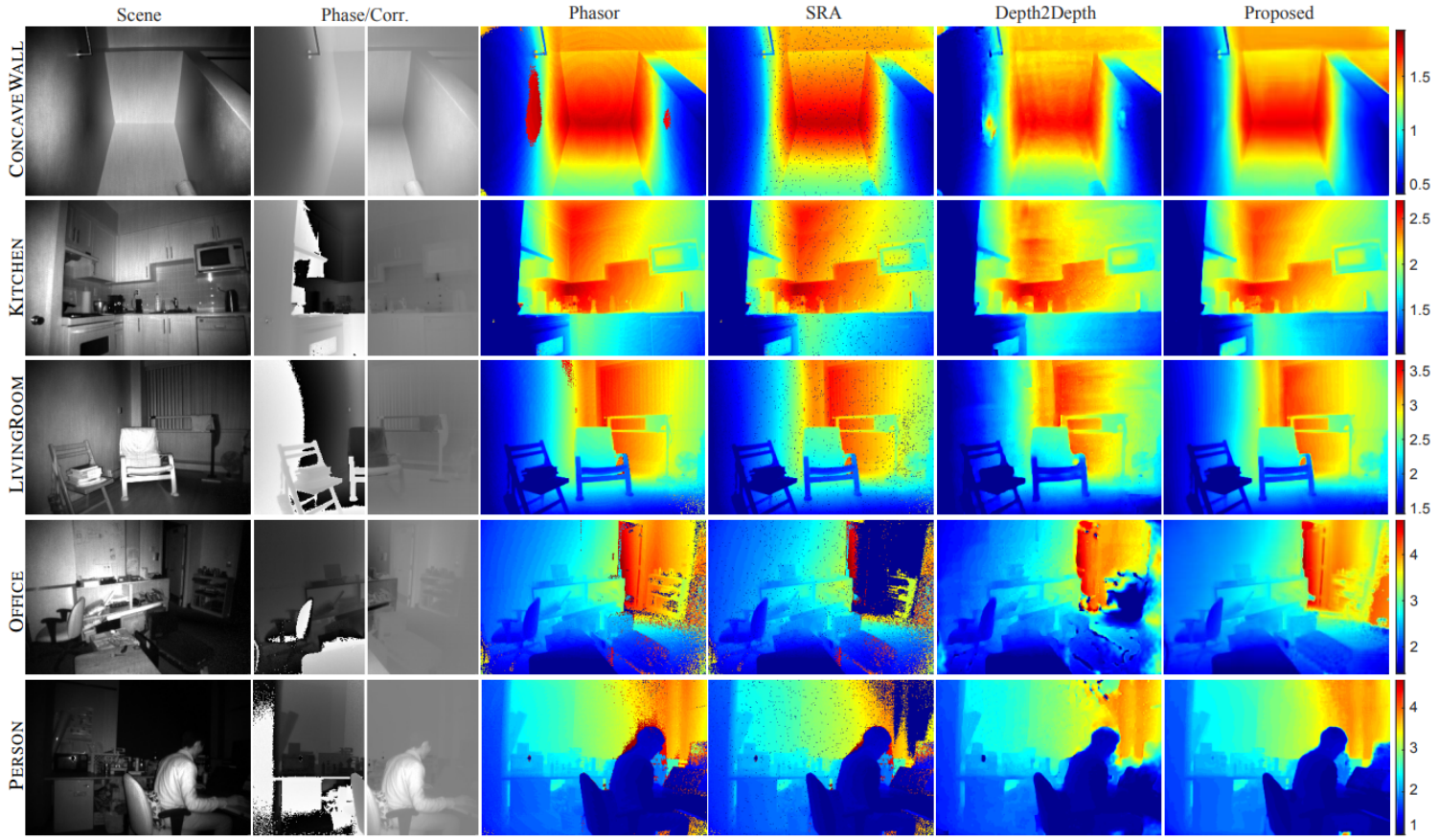
Image from: Su et al., “Deep End-to-End Time-of-Flight Imaging,” IEEE CVPR 2018

TIME-OF-FLIGHT CAMERAS



Su et al., “Deep End-to-End Time-of-Flight Imaging,” IEEE CVPR 2018

TIME-OF-FLIGHT CAMERAS



Su et al., “Deep End-to-End Time-of-Flight Imaging,” IEEE CVPR 2018

CONCLUSION

Image Processing with Deep Learning

- Single Image
- Multiple Images
- Other Sensors

Recurring Themes

- Loss Functions (GANs)
- Encoder/Decoder Networks
- Correlation

YOUR LIFE'S WORK STARTS HERE

JOIN NVIDIA

100 Best Companies to Work For
— Fortune

Most Innovative Companies
— Fast Company

World's Most Admired Companies
— Fortune

Employees' Choice: Highest Rated CEOs
— Glassdoor

50 Smartest Companies
— MIT Tech Review

INTERESTED? Email: aijobs@nvidia.com

NVIDIA