Design of Deep Learning Architectures

Michael Elad

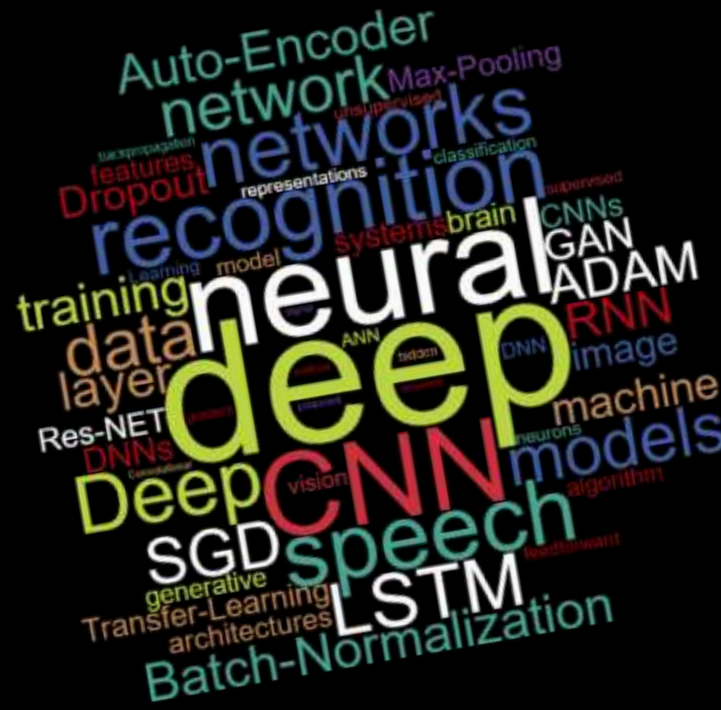
Computer Science Department
The Technion - Israel Institute of Technology
Haifa 32000, Israel

Google Research
Perception - LUMA

Joint work with
Peyman Milanfar

Background

Deep Learning is Everywhere



- ❑ Our focus: computational imaging tasks such as denoising, restoration, segmentation, super-resolution …
- ❑ Deep-learning based solutions have taken a leading role in our field, due to their impressive performance and ease of design
- ❑ The general feeling among younger researchers: No need to understand anything anymore – the learning takes care of that

Deep-Learning Working Path

In building supervised deep learning solutions in computational imaging we operate along the following lines:

Define the problem to be solved → Gather training data to use → **Define an architecture for the solution** → Define a cost function (loss) to optimize → Choose your optimization strategy → Train and hope for good generalization

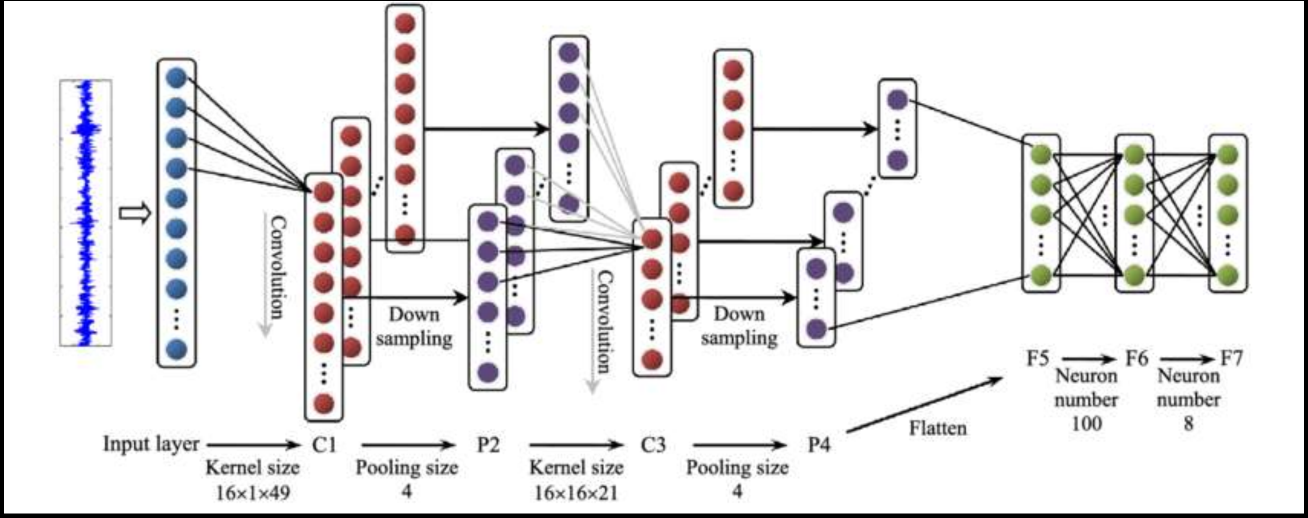
This talk is about item #3: The choice of the architectures

Choosing Architectures

So, how do we choose an architecture for a given task?

Option 1 - Copy an existing network that has shown good results in earlier work (VGG, U-Net, ...), and slightly modify it

Option 2 – Pile and Guess a series of steps that mix known pieces such as convolutions, fully connected layer, batch-norm, ReLU, pooling, stride, skips, upscale/downscale, connections, ... and maybe add new “tricks”



Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures

Variational Denoising Network: Toward Blind Noise Modeling and Removal

Zongsheng Yue[1,2], Hongwei Yong[2], Qian Zhao[1], Lei Zhang[2,3], Deyu Meng[1,*]

[1] *School of Mathematics and Statistics, Xi'an Jiaotong University, Shaanxi, China*
[2] *Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong*
[3] *DAMO Academy, Alibaba Group, Shenzhen, China*
[*] *Corresponding author: dymeng@mail.xjtu.edu.cn*

NIPS 2019: U-Net-based with 7e6 params

Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures

Variational Denoising Network: Toward Blind Noise Modeling and Removal

Zongsheng Yue[1,2], Hongwei Yong[2], Qian Zhao[1], Lei Zhang[2,3], Deyu Meng[1,*]

[1] *School of Mathematics and Statistics, Xi'an Jiaotong University, Shaanxi, China*

[2] *Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong*

hina

u.cn

e6 params

Modulating Image Restoration with Continual Levels via Adaptive Feature Modification Layers

Jingwen He[1,*] Chao Dong[1,*] Yu Qiao[1,2,†]

[1] ShenZhen Key Lab of Computer Vision and Pattern Recognition, SIAT-SenseTime Joint Lab, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China

[2] The Chinese University of Hong Kong

CVPR 2019: DnCNN-based with 1.2e6 params

Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019
that illustrate these architectures

Meta-SR: A Magnification-Arbitrary Network for Super-Resolution

Xuecai Hu*[1,2] , Haoyuan Mu*[4], Xiangyu Zhang[3], Zilei Wang[1], Tieniu Tan[1,2], Jian Sun[3]
[1] University of Science and Technology of China
[2] Center for Research on Intelligent Perception and Computing, NLPR, CASIA
[3] Megvii Inc (Face++) [4] Tsinghua University
huxc@mail.ustc.edu.cn, muhy17@mails.tsinghua.edu.cn
{zhangxiangyu, sunjian}@megvii.com, zlwang@ustc.edu.cn, tnt@nlpr.ia.ac.cn

CVPR 2019: Huge network with 2e6 params

...vork: Toward Blind Noise ...d Removal

...n Zhao[1], Lei Zhang[2,3], Deyu Meng[1,*]
...'an Jiaotong University, Shaanxi, China
[2]Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong
...hina
...u.cn

...e6 params

Modulating Image Restoration with Continual Levels via Adaptive Feature Modification Layers

Jingwen He[1,*] Chao Dong[1,*] Yu Qiao[1,2,†]
[1]ShenZhen Key Lab of Computer Vision and Pattern Recognition, SIAT-SenseTime Joint Lab,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[2]The Chinese University of Hong Kong

CVPR 2019: DnCNN-based with 1.2e6 params

Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures

Meta-SR: A Magnification-Arbitrary Network for Super-Resolution

Xuecai Hu[*1,2] , Haoyuan Mu[*4], Xiangyu Zha
[1] University of Science an
[2] Center for Research on Intelligent Perce
[3] Megvii Inc (Face++) [4]
huxc@mail.ustc.edu.cn, muhy
{zhangxiangyu, sunjian}@megvii.com, zlw

CVPR 2019: Huge netw

3D Appearance Super-Resolution with Deep Learning

Yawei Li[1], Vagia Tsiminaki[2], Radu Timofte[1], Marc Pollefeys[2,3], and Luc van Gool[1]
[1]Computer Vision Lab, ETH Zurich, Switzerland
{yawei.li, radu.timofte, vangool}@vision.ee.ethz.ch
[2]Computer Vision and Geometry Group, ETH Zurich, Switzerland, [3]Microsoft, USA
{vagia.tsiminaki, marc.pollefeys}@inf.ethz.ch

CVPR 2019: Huge network with 4e6 params

e6 params

Modulating Imag
via Adaptive Feature Modification Layers

Jingwen He[1,*] Chao Dong[1,*] Yu Qiao[1,2,†]
[1]ShenZhen Key Lab of Computer Vision and Pattern Recognition, SIAT-SenseTime Joint Lab,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[2]The Chinese University of Hong Kong

CVPR 2019: DnCNN-based with 1.2e6 params

Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures

Meta-SR: A Magnification-Arbitrary Network for Super-Resolution

Xuecai Hu[*1,2] , Haoyuan Mu[*4], Xiangyu Zha
[1] University of Science an
[2] Center for Research on Intelligent Perce
[3] Megvii Inc (Face++) [4]
huxc@mail.ustc.edu.cn, muhy
{zhangxiangyu, sunjian}@megvii.com, zlw

CVP

3D Appearance Super-Resolution with Deep Learning

Yawei Li[1], Vagia Tsiminaki[2], Radu Timofte[1], Marc Pollefeys[2,3], and Luc van Gool[1]
[1]Computer Vision Lab, ETH Zurich, Switzerland
.ee.ethz.ch
tzerland, [3]Microsoft, USA
.ethz.ch

4e6 params

e6 params

High-Quality Self-Supervised Deep Image Denoising

Samuli Laine	Tero Karras	Jaakko Lehtinen	Timo Aila
NVIDIA*	NVIDIA	NVIDIA, Aalto University	NVIDIA

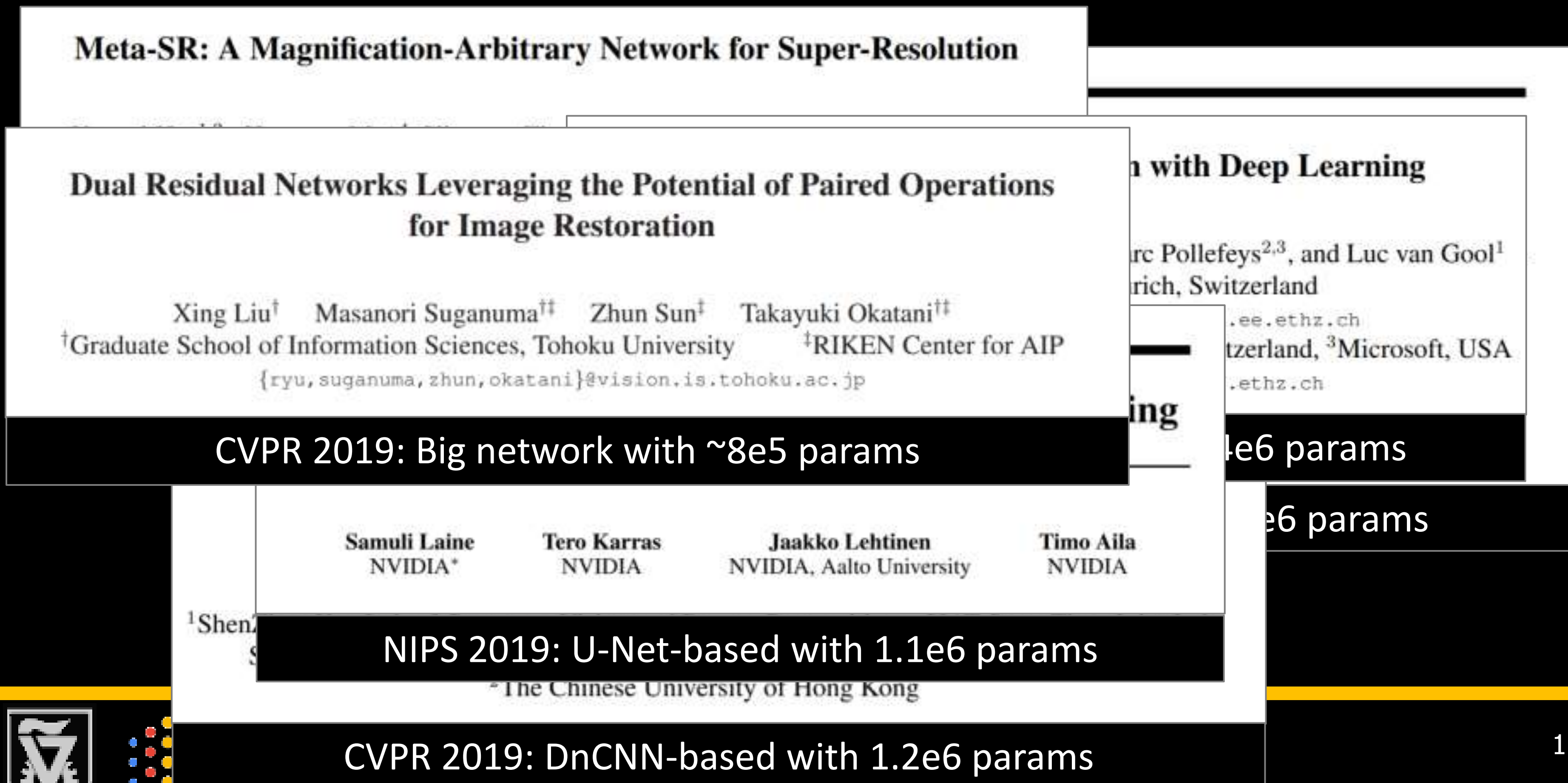
NIPS 2019: U-Net-based with 1.1e6 params

[1]Shen
S
[2]The Chinese University of Hong Kong

CVPR 2019: DnCNN-based with 1.2e6 params

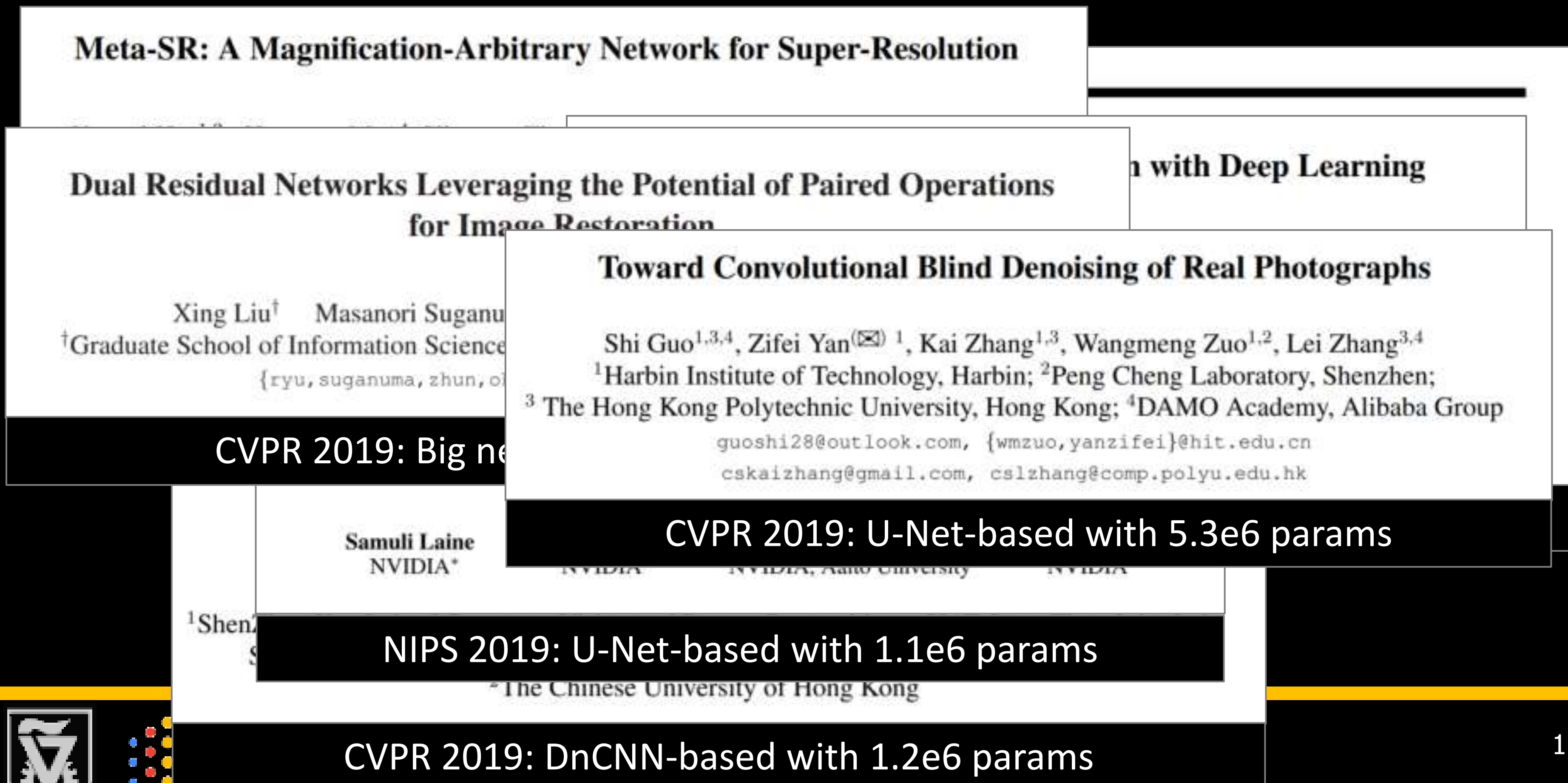
Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures



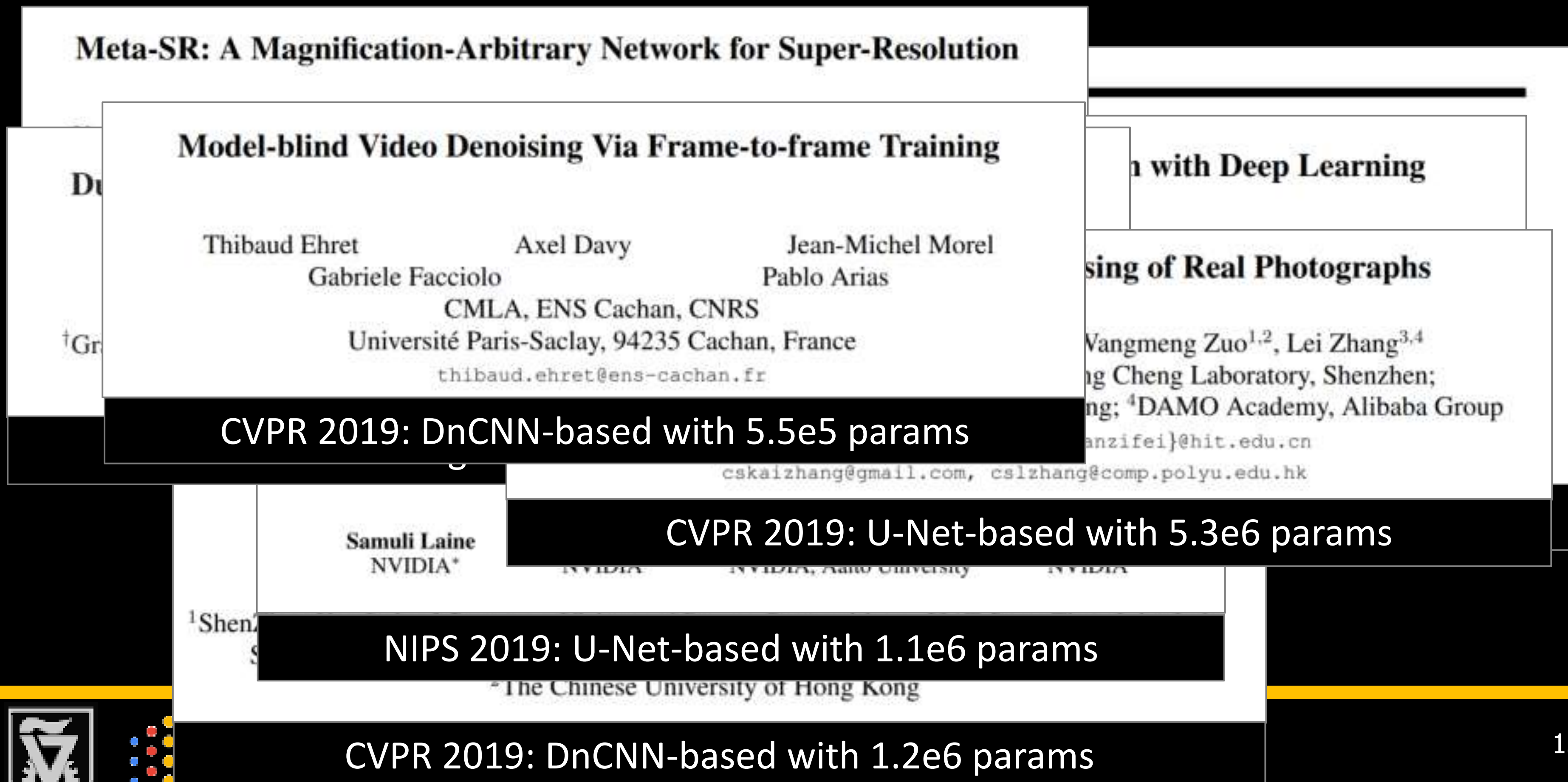
Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures



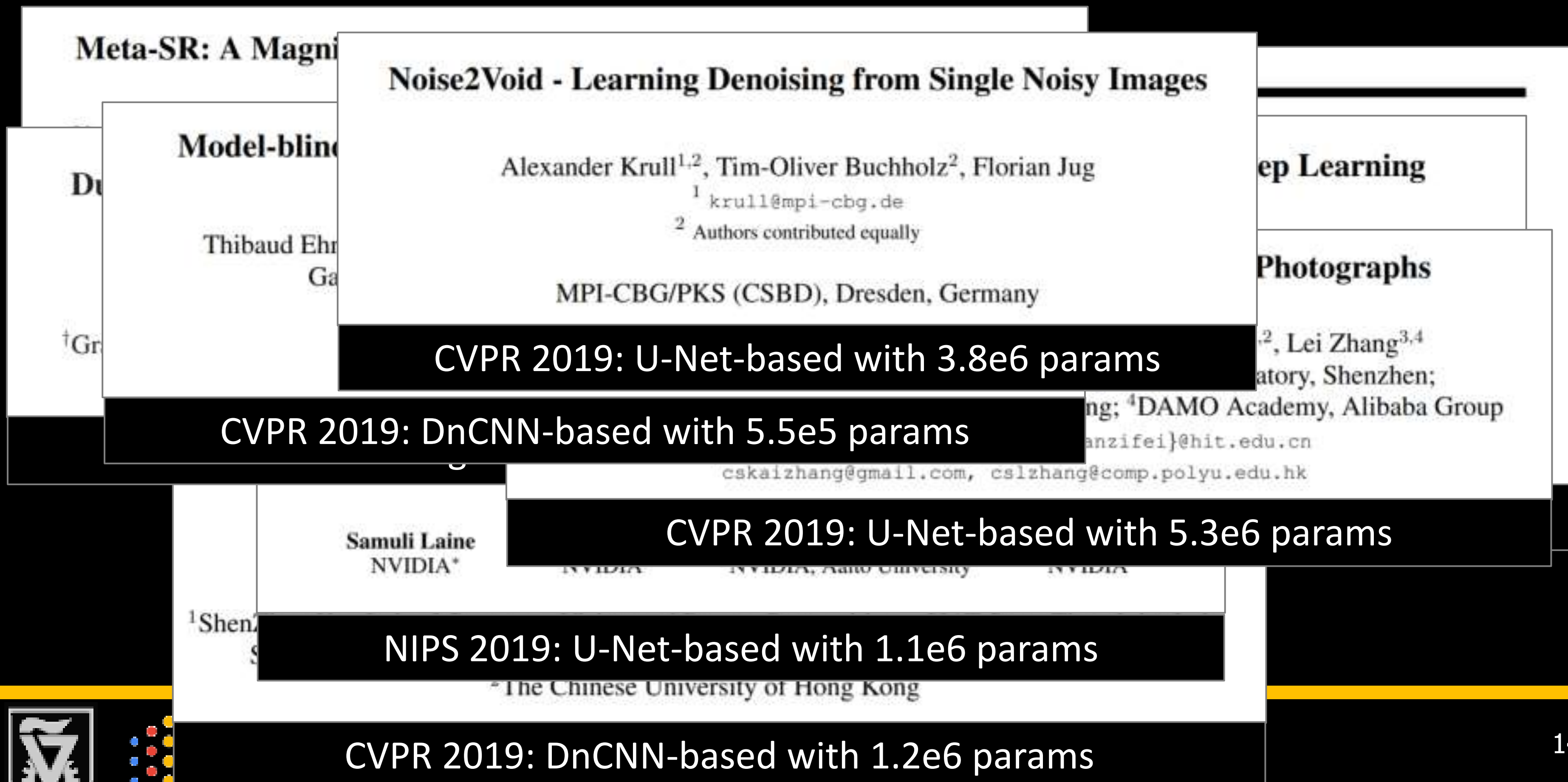
Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures



Few Recent Examples

Here are several paper examples from CVPR/NIPS 2019 that illustrate these architectures



Main Question #1

Bottom line:

- This brute-force approach for choosing the architecture seems to work rather well
- However, this approach typically tends to very heavy and cumbersome networks
- Lacking more insight, this approach produces black-boxes that are likely to hit a performance barrier soon (if not already)

Main Question #1 in this Talk:

Can we do better in choosing our architectures ?

Main Question #2:

Lets move to something seemingly totally different …

- ❑ Massive research activity in image processing during the past 3-5 decades has brought vast knowledge and knowhow
- ❑ The entrance of supervised deep-learning solutions in the past decade seems to have bypassed this knowledge altogether, offering a highly effective and totally different alternative path towards the design of solution for imaging tasks

Main Question #2 in this Talk:

Has the classic knowledge in Image processing became obsolete in the era of deep-learning?

On a Personal Note

- Allow me to be more specific and slightly more personal:
  - In the past 20 years I have been working quite extensively on the sparse representation model for visual data
  - Key idea: signals can be effectively represented as a sparse combination of atoms from a given dictionary
  - We and many others have shown the applicability of this model to various tasks, both in image processing and in other domains
  - I strongly believe that this model is key in explaining many of our algorithms/processes for handling data in general
- So, here is a refined version of Question 2:

Is the knowledge on sparse modeling of data useless in the era of deep-learning?

This Lecture

This Lecture focuses on the Above Two Seemingly Unrelated Questions

Question 1: Is there a systematic way to design deep-learning architectures?

Question 2: What about all the accumulated knowledge in image processing over the past 50 years? Has it become obsolete?

We argue that the two questions are strongly interconnected, and there is a common answer to both

This Lecture

This Lecture focuses on the Above Two Seemingly Unrelated Questions

Our Claim: We can do far better in choosing deep-learning architectures by relying systematically on the classics of image processing and sparse representations for their formation

The benefits in such architectures:

1. They are far more concise yet just as effective as leading methods
2. They are easier to train because they are lighter
3. They have the potential to break current performance barriers
4. They may bring better understanding and explainability
5. This gives a good feeling that our past work has not been in vain

This Lecture

This Lecture focuses on the Above Two Seemingly Unrelated Questions

- In this talk I would like to demonstrate the above by describing VERY BRIEFLY three of our recent papers, all addressing the image denoising problem:
  - Deep KSVD Denoising [Scetbon, Milanfar & Elad, arXiv:1909.13164, Sep. `19]
  - Non-Local & Multi-Scale Denoising [Vaksman, Milanfar & Elad, arXiv:1911.07167, Nov.`19]
  - Rethinking the CSC Model [Simon & Elad, NIPS `19]
- Our message: classic image denoising algorithms can be turned into differentiable and relatively concise schemes and those can be trained in a supervised fashion, leading to excellent results

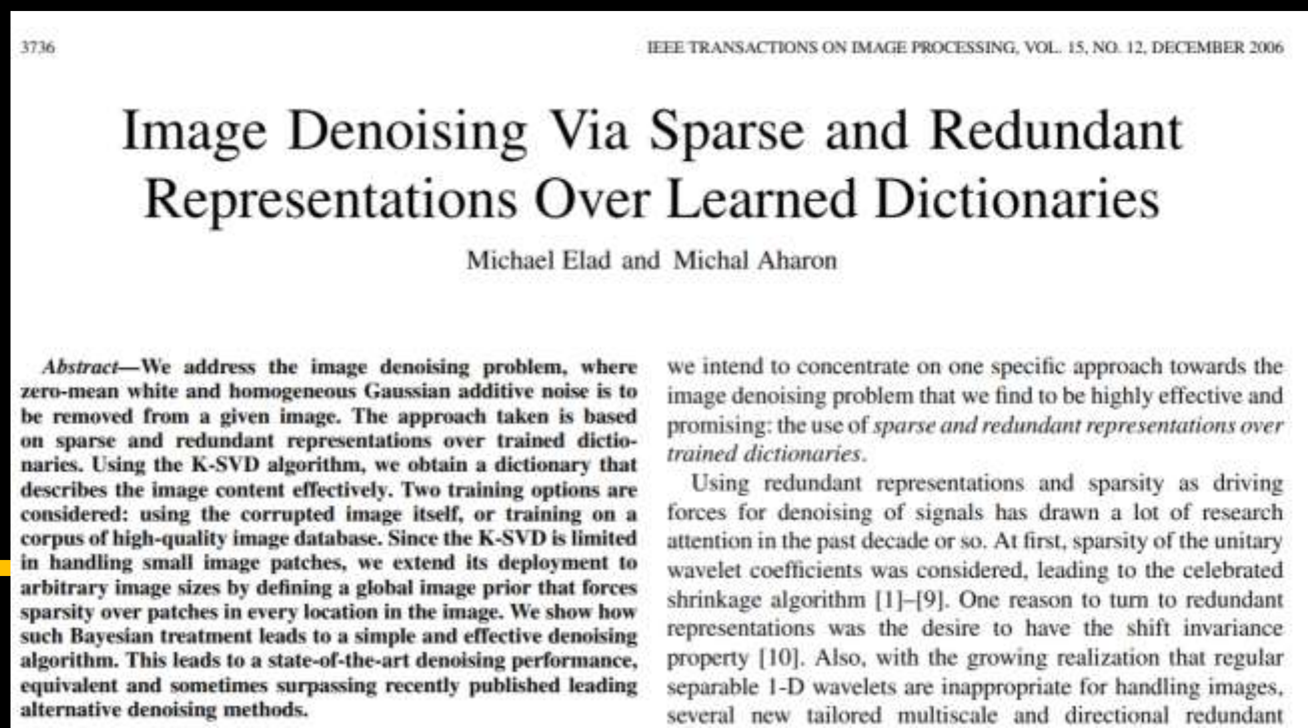
Deep KSVD Denoising

Meyer Scetbon

M. Scetbon, M. Elad, and P. Milanfar, Deep K-SVD Denoising, arXiv:1909.13164, Sep. `19

Paper #1: Deep K-SVD

- In 2006, we developed a new and highly effective image denoising alg. by relying on sparsity of image patches and a learned dictionary
- This was considered as state-of-the-art for whole 2 minutes until beaten by competition (BM3D, NCSR, TNRD, WNNM, …)
- Over the years, various improvements of it came up – e.g. exploiting joint sparsity [Mairal et. al. `09] or leveraging the EPLL [Sulam et. al. `15]
- … And recently we decided to revisit this method …



3736 IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 15, NO. 12, DECEMBER 2006

Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries

Michael Elad and Michal Aharon

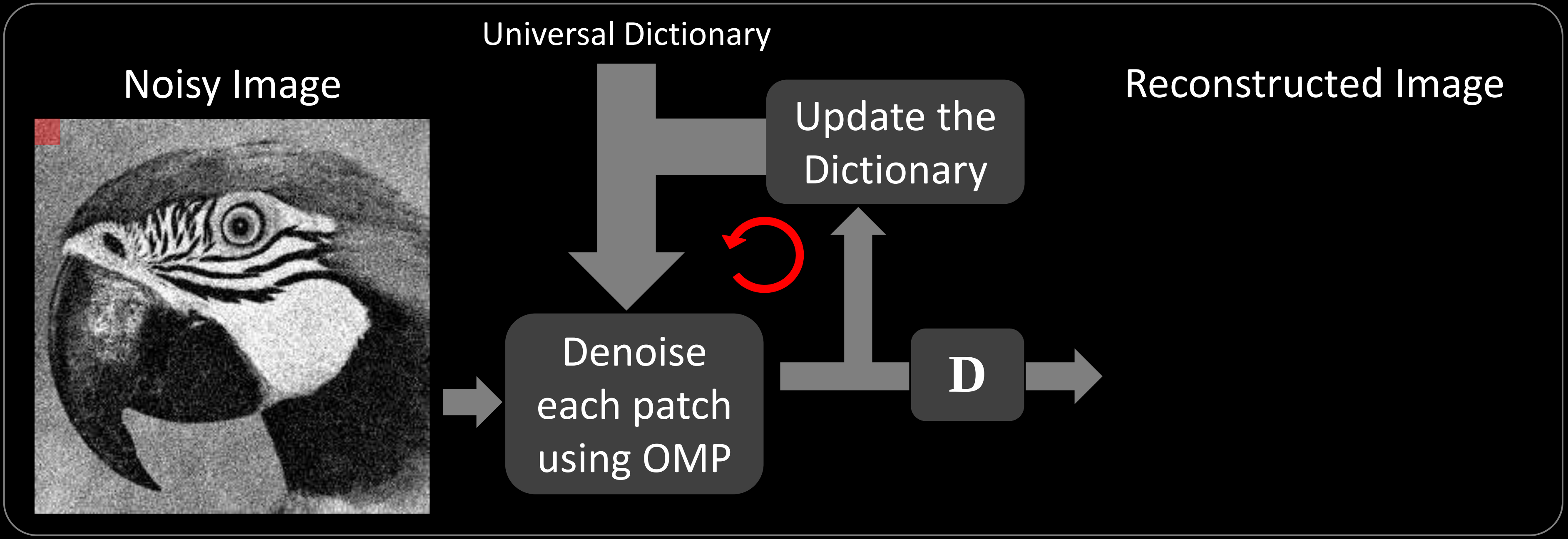
Abstract—We address the image denoising problem, where zero-mean white and homogeneous Gaussian additive noise is to be removed from a given image. The approach taken is based on sparse and redundant representations over trained dictionaries. Using the K-SVD algorithm, we obtain a dictionary that describes the image content effectively. Two training options are considered: using the corrupted image itself, or training on a corpus of high-quality image database. Since the K-SVD is limited in handling small image patches, we extend its deployment to arbitrary image sizes by defining a global image prior that forces sparsity over patches in every location in the image. We show how such Bayesian treatment leads to a simple and effective denoising algorithm. This leads to a state-of-the-art denoising performance, equivalent and sometimes surpassing recently published leading alternative denoising methods.

we intend to concentrate on one specific approach towards the image denoising problem that we find to be highly effective and promising: the use of *sparse and redundant representations over trained dictionaries*.

Using redundant representations and sparsity as driving forces for denoising of signals has drawn a lot of research attention in the past decade or so. At first, sparsity of the unitary wavelet coefficients was considered, leading to the celebrated shrinkage algorithm [1]–[9]. One reason to turn to redundant representations was the desire to have the shift invariance property [10]. Also, with the growing realization that regular separable 1-D wavelets are inappropriate for handling images, several new tailored multiscale and directional redundant

Paper #1: Deep K-SVD

So, how does the original K-SVD denoiser work?

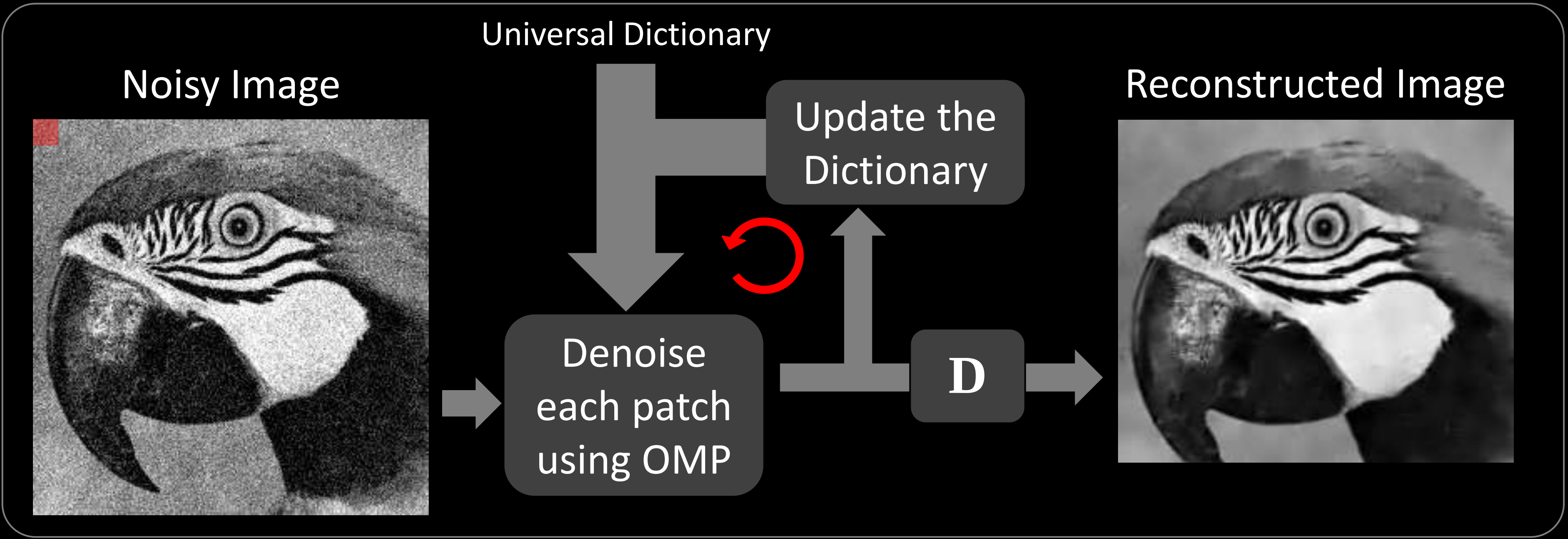


Core idea: Assume that all patches obey sparse modeling

$$\min_{\alpha} \|\alpha\|_0 \ \text{s.t.} \ \|\mathbf{D}\alpha - \mathbf{R}_i y\|_2 \leq T$$

Paper #1: Deep K-SVD

So, how does the original K-SVD denoiser work?



Core idea: Assume that all patches obey sparse modeling

$$\min_{\alpha} \|\alpha\|_0 \ \text{s.t.} \ \|\mathbf{D}\alpha - \mathbf{R}_i y\|_2 \leq T$$

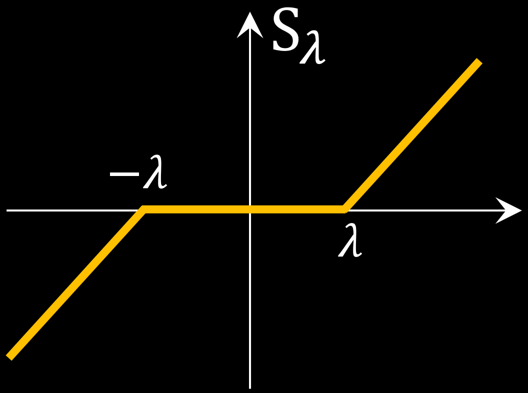
Paper #1: Deep K-SVD

Unfolding this Algorithm:

- All patches (with full overlaps) go through the same "pursuit" in parallel
- OMP problematic (L_0, greedy) $\rightarrow$ Use LISTA [Gregor & LeCun `00] (7 iterations):

$$\min_{\alpha} \ \|\alpha\|_1 + \lambda\|\mathbf{D}\alpha - \mathbf{R}_i y\|_2^2$$

$$\rightarrow \quad \alpha_{k+1} = S_\lambda\{\alpha_k + c\mathbf{D}^T(\mathbf{D}\alpha_k - \mathbf{R}_i y)\}$$ [ISTA]



- Each patch should get a dynamic # of non-zeros
  $\rightarrow$ Get an adaptive λ by another small network

Bottom Line:

- The dictionary and few other parameters are learned in a supervised fashion
- Our reference method to compare with is DnCNN (550K params) [Zhang, `17]
- Using 45K params, this elementary method gets within 0.1-0.2dB to DnCNN

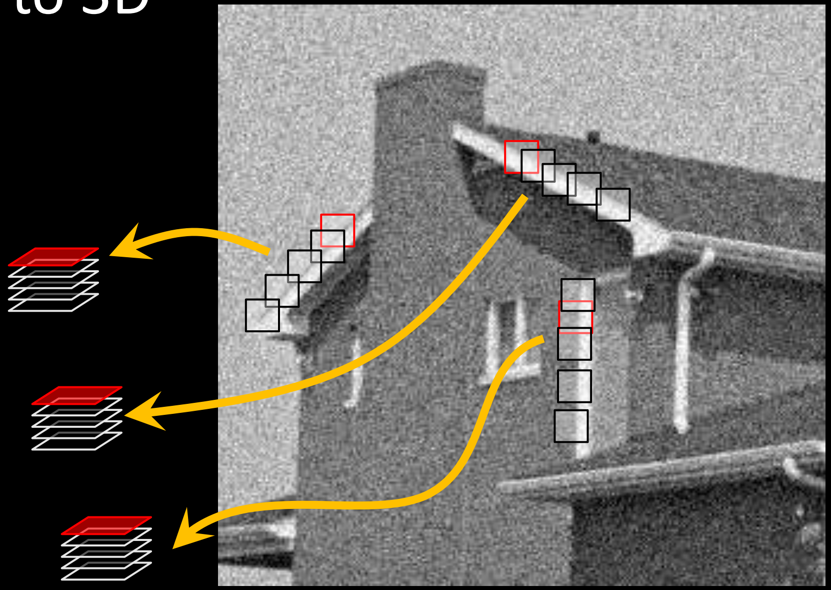
Non-Local & Multi-Scale Denoising

Grisha Vaksman

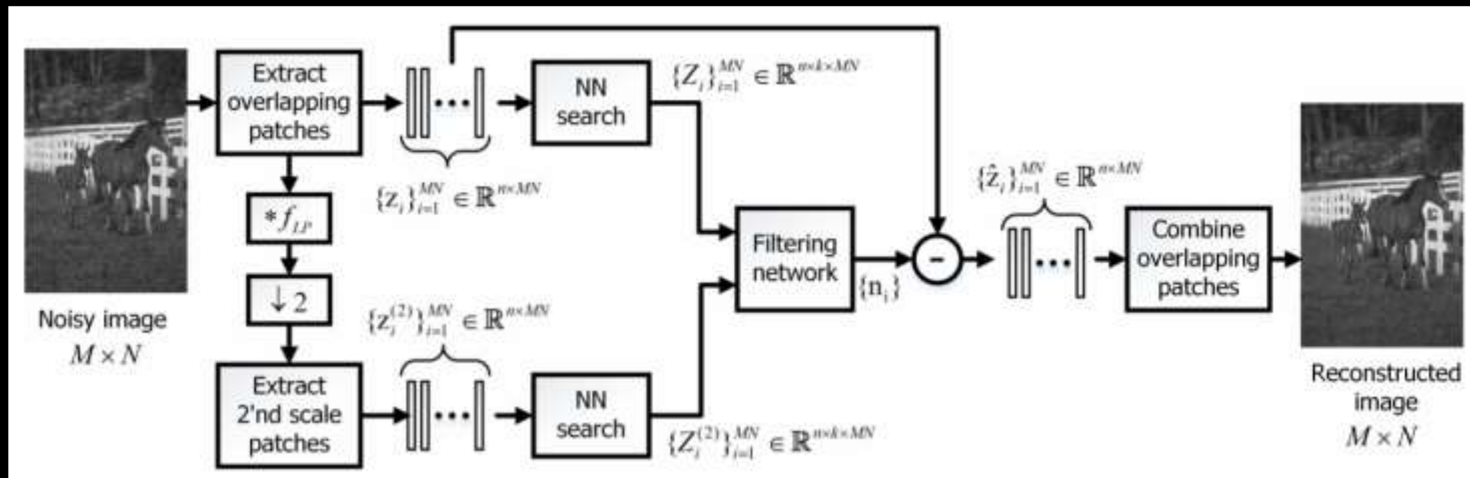
G. Vaksman, M. Elad and P. Milanfar, Low-Weight and Learnable Image Denoising, arXiv:1911.07167 , Nov. `19

Paper #2: Non-Local and Multi-Scale

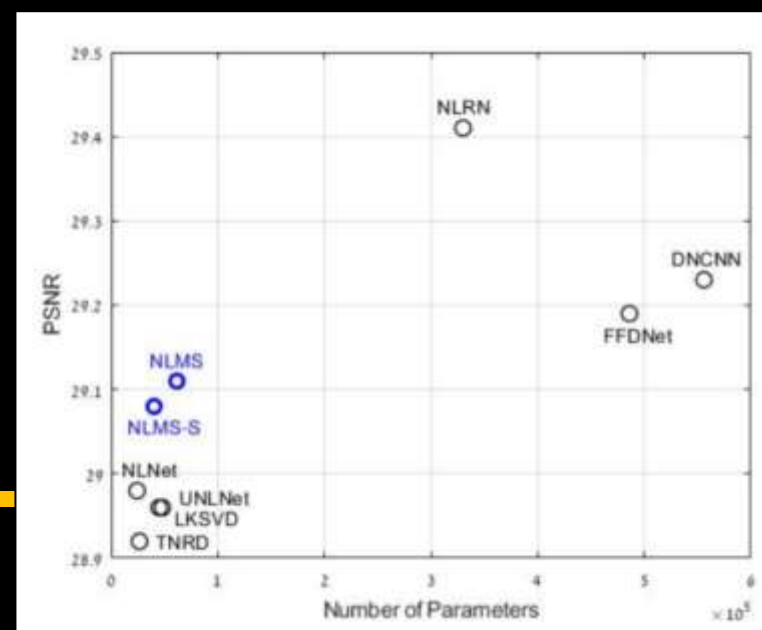
- Two key forces that the previous work has totally failed to use are (i) self-similarity and (ii) multi-scale connections
- BM3D [Dabov et. al 2006]: A highly effective denoiser based on sparsity and self-similarity
- Its core idea: Gather similar patches to 3D blocks and sparse code them jointly
- Our idea: Unfold this algorithm and augment it with a multi-scale treatment, and design its parameters via supervised learning
- This work has been inspired by [Lefkimmiatis `17] and [Lefkimmiatis `18]



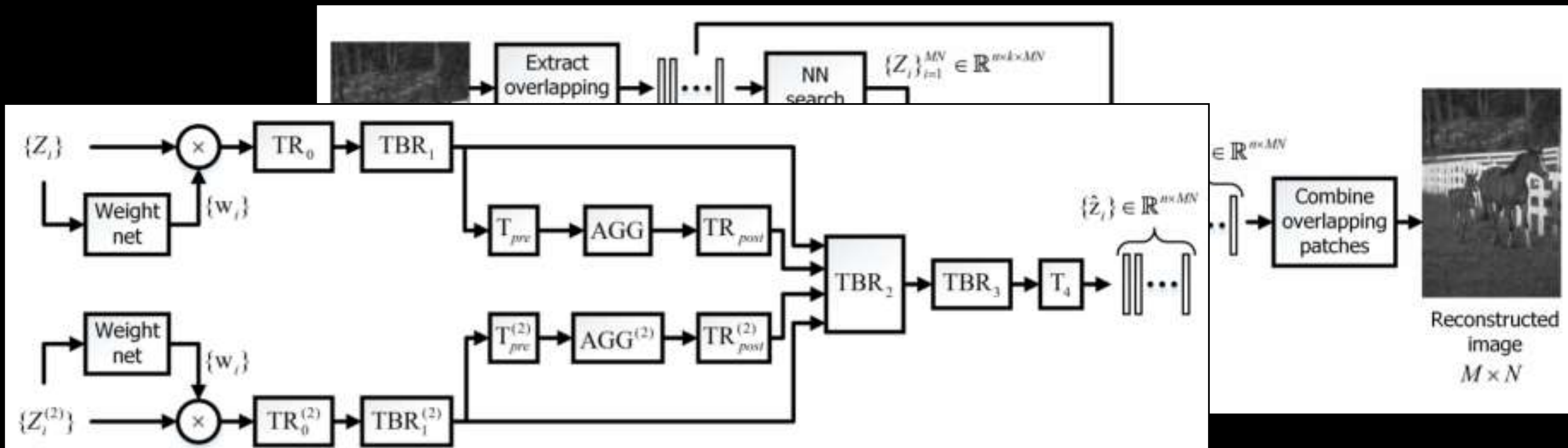
Paper #2: Non-Local and Multi-Scale



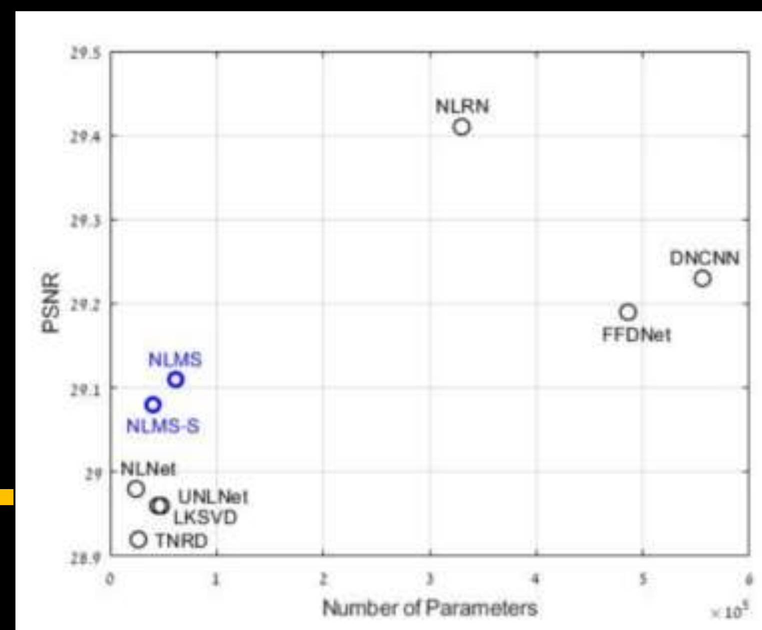
- ❑ The proposed architecture implements the ideas mentioned above in a simple a direct way
- ❑ This illustrates the performance vs. #of parameters for various networks



Paper #2: Non-Local and Multi-Scale

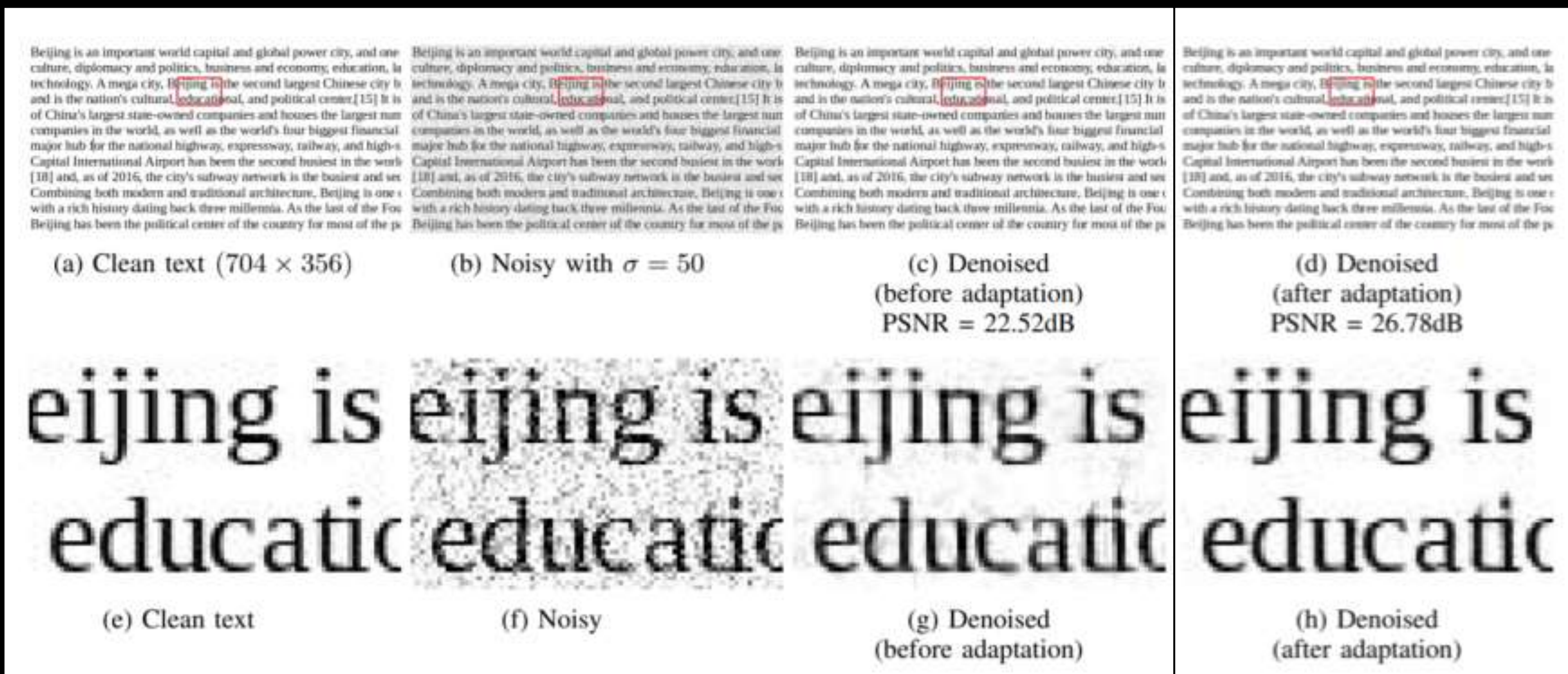


- The proposed architecture implements the ideas mentioned above in a simple a direct way
- This illustrates the performance vs. #of parameters for various networks



Paper #2: Non-Local and Multi-Scale

- Bottom Line: Using 60K learned parameters (instead of 550K), this method gets within 0.05-0.1dB to DnCNN
- An additional benefit: Fast and effective adaptation capability

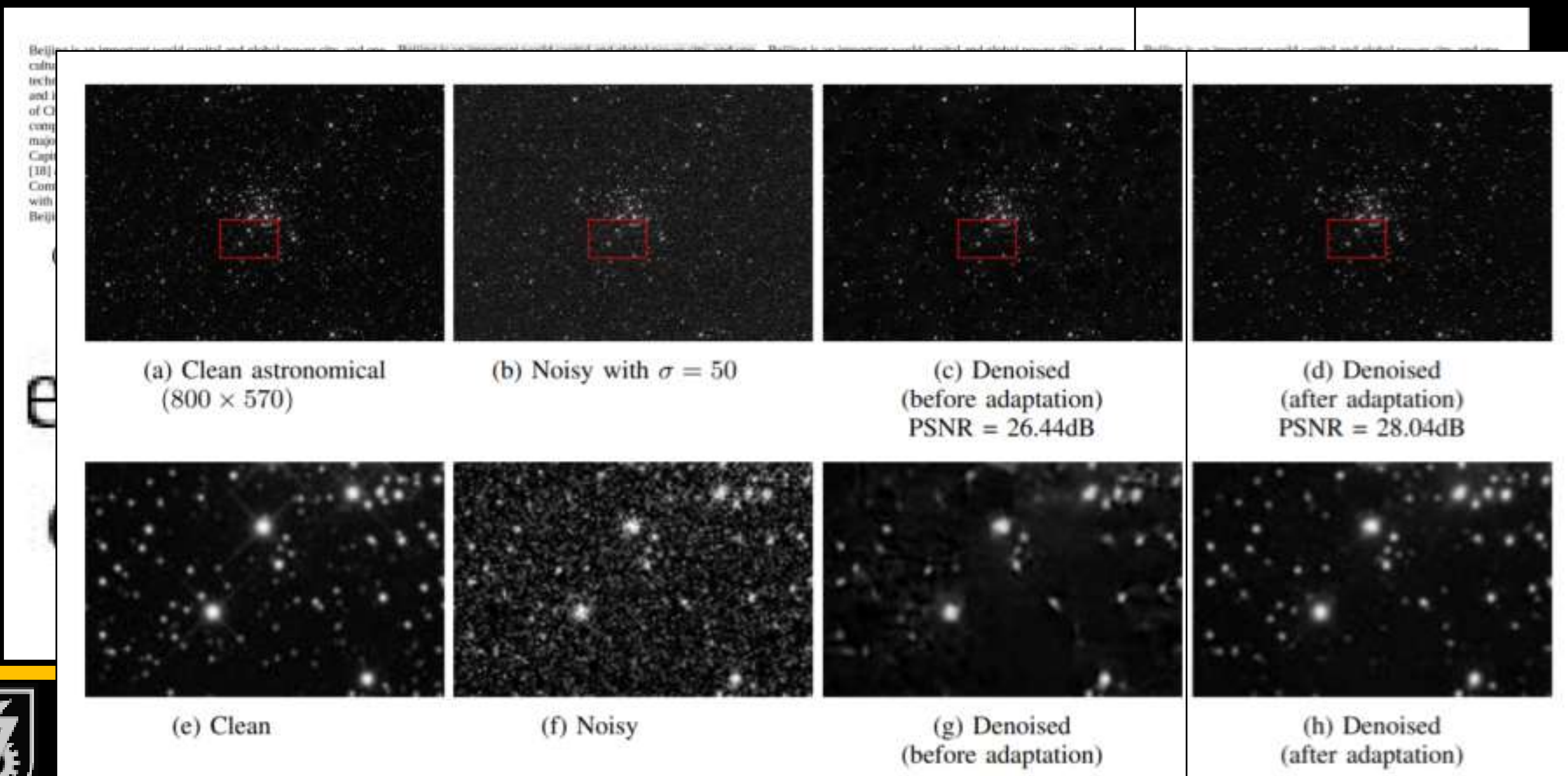


(a) Clean text (704 × 356) (b) Noisy with $\sigma = 50$ (c) Denoised (before adaptation) PSNR = 22.52dB (d) Denoised (after adaptation) PSNR = 26.78dB

(e) Clean text (f) Noisy (g) Denoised (before adaptation) (h) Denoised (after adaptation)

Paper #2: Non-Local and Multi-Scale

- Bottom Line: Using 60K learned parameters (instead of 550K), this method gets within 0.05-0.1dB to DnCNN
- An additional benefit: Fast and effective adaptation capability



Rethinking the CSC Model

D. Simon and M. Elad, Rethinking the CSC Model for Natural Images, NIPS 2019

Dror Simon

Paper #3: Rethinking CSC

Remember the earlier Pursuit task and ISTA?

$$\min_{\alpha} \|\alpha\|_1 + \lambda\|\mathbf{D}\alpha - \mathbf{R}_i y\|_2^2 \;\rightarrow\; \alpha_{k+1} = S_\lambda\{\alpha_k + c\mathbf{D}^T(\mathbf{D}\alpha_k - \mathbf{R}_i y)\}$$

Why work on patches? Lets apply this on the whole image!

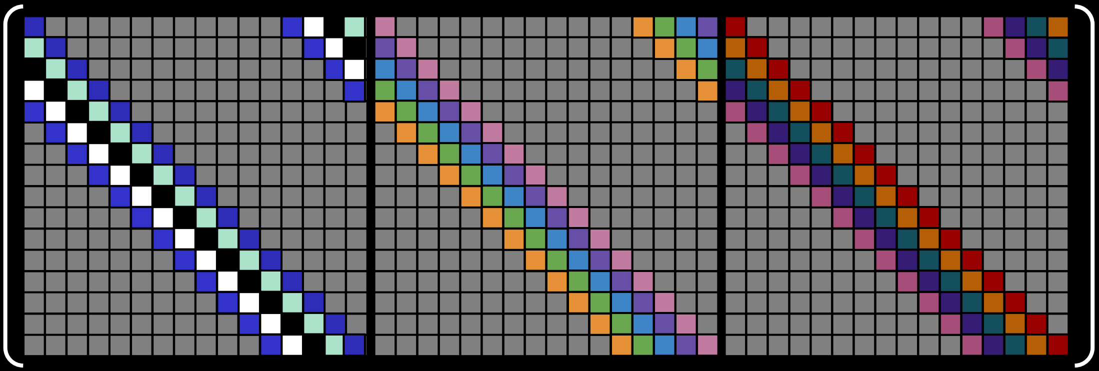
$$\min_{\alpha} \|\alpha\|_1 + \lambda\|\mathbf{D}\alpha - Y\|_2^2 \;\rightarrow\; \alpha_{k+1} = S_\lambda\{\alpha_k + c\mathbf{D}^T(\mathbf{D}\alpha_k - Y)\}$$

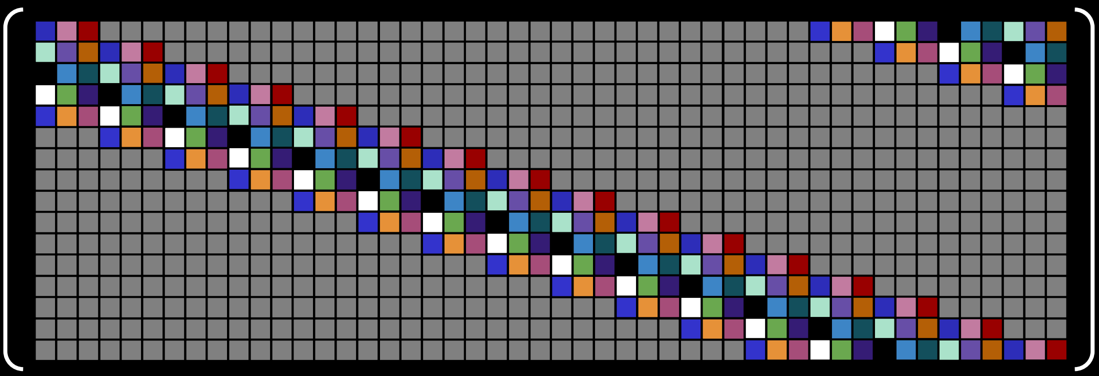
Great, but who is $\mathbf{D}$ in this case?

This brings us to the Convolutional Sparse Coding (CSC) Model

Paper #3: Rethinking CSC

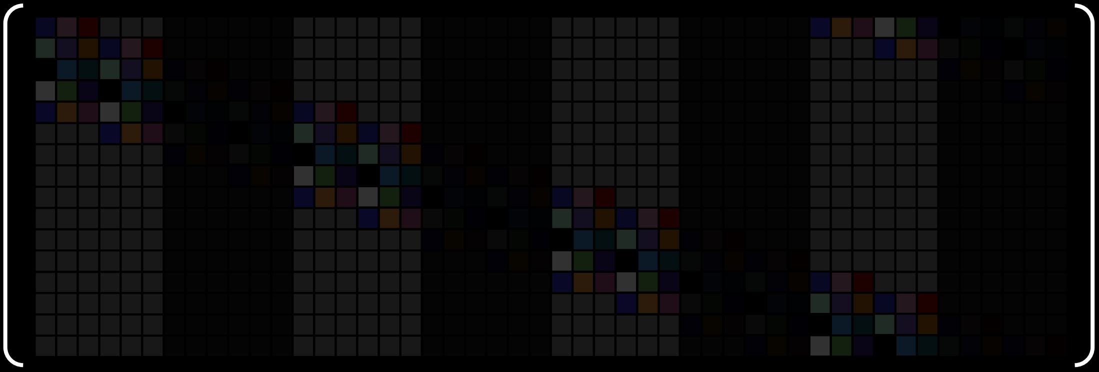
- CSC assumes a structured dictionary: $\mathbf{D}$ is built of m small filters
- Thus, multiplication by $\mathbf{D}$ and $\mathbf{D}^{\mathbf{T}}$ amount to convolutions
- Great! So lets apply LISTA on this pursuit and train it in a supervised way for best denoising results
- This is exactly the idea in [Giryes et. al. '18] and their results are (at best) getting close to BM3D
- So, are we stuck?





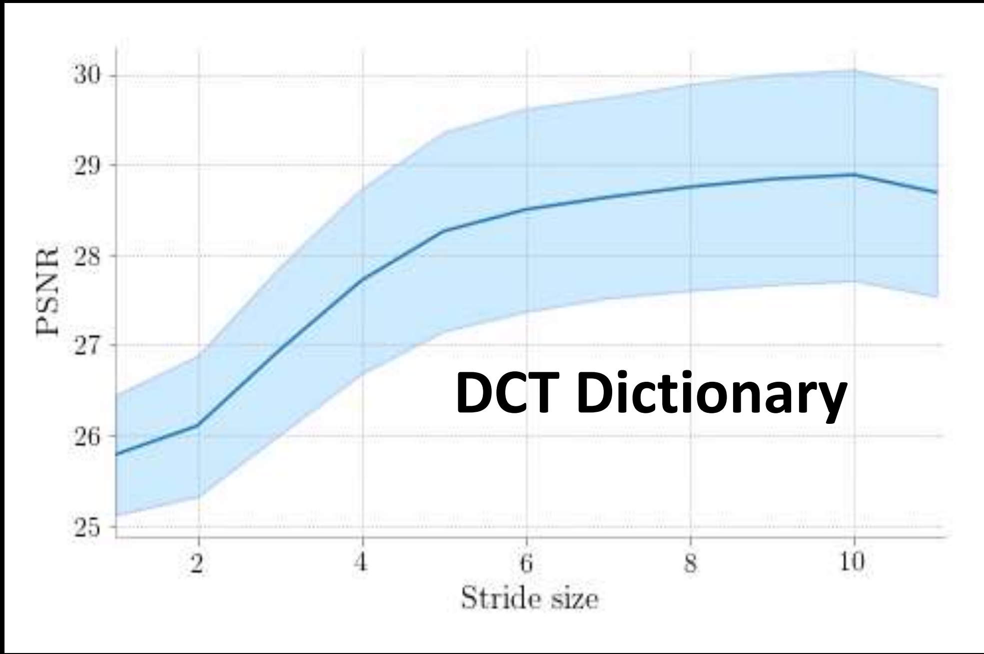
Paper #3: Rethinking CSC

- Dror's idea: Use the CSC while deploying an MMSE estimation
- Observation 1: The CSC dictionary has a horrible coherence
- Observation 2: Denoising could be improved by moving to MMSE
- Observation 3: Subsampling the dictionary, solving the pursuit for all offsets, and averaging the results $\rightarrow$ you get MMSE approx.
- Create a network along this idea and train it for denoising
- Bottom line: using 63K params, this algorithm works as good as DnCNN and even better



Paper #3: Rethinking CSC

- If the filter size is $n = 11$ then the stride (subsampling factor) could be anything in the range [1,11]:
  - $q = 1$: no subsampling – this is [Giryes et. al. '18] all over again
  - $q = 11$: this is a patch-averaging, just as in the K-SVD denoising
  - $q = 9/10$: performs best



- Side result: Using CSC with a stride generalizes the patch-based method that is so popular in image processing

Wrapping Up

Summary

- The right way to build solutions to imaging tasks goes as follows:

Modeling the Data → **Algorithm Development** → **Unfold this to a Network** → **Apply Supervised Learning**

- What should be taken into account for the algorithm’ design?
  - The degradation and noise statistics (“the physics”)
  - Prior on the image: (i) Non-Local self similarity; (ii) multi-scale connections; & (iii) Sparsity or other form of simplicity (e.g. low-rank)
  - The objective (e.g., MMSE)

- More broadly, I believe that sparse modeling of data is key
  - In explaining existing deep-learning architectures
  - In creation of new ones
  - In bringing theoretical understanding to deep-learning

Still Unanswered

Open Questions:

- When designing an algorithm (and thus a network) for solving inverse problems, should we consider MMSE or MAP?
- It will be great to see this advocated rationale breaking existing performance barriers – this is yet to happen
- What about using this rationale for supporting unsupervised solutions? Recall the K-SVD denoising with an adapted dictionary
- We mentioned in the beginning that this talk focuses on regression tasks in computational imaging. What about recognition or synthesis tasks?

Is this Becoming a Trend?

BTW, take a look at this recent work by Mairal

Revisiting Non Local Sparse Models for Image Restoration

Bruno Lecouat *
Inria
bruno.lecouat@inria.fr

Jean Ponce *
Inria
jean.ponce@inria.fr

Julien Mairal †
Inria
julien.mairal@inria.fr

January 29, 2020

Abstract

We propose a differentiable algorithm for image restoration inspired by the success of sparse models and self-similarity priors for natural images. Our approach builds upon the concept of joint sparsity between groups of similar image patches, and we show how this simple idea can be implemented in a differentiable architecture, allowing end-to-end training. The algorithm has the advantage of being interpretable, performing sparse decompositions of image patches, while being more parameter efficient than recent deep learning methods. We evaluate our algorithm on grayscale and color denoising, where we achieve competitive results, and on demoisaicking, where we outperform the most recent state-of-the-art deep learning model with 47 times less parameters and a much shallower architecture.

Is this Becoming a Trend?

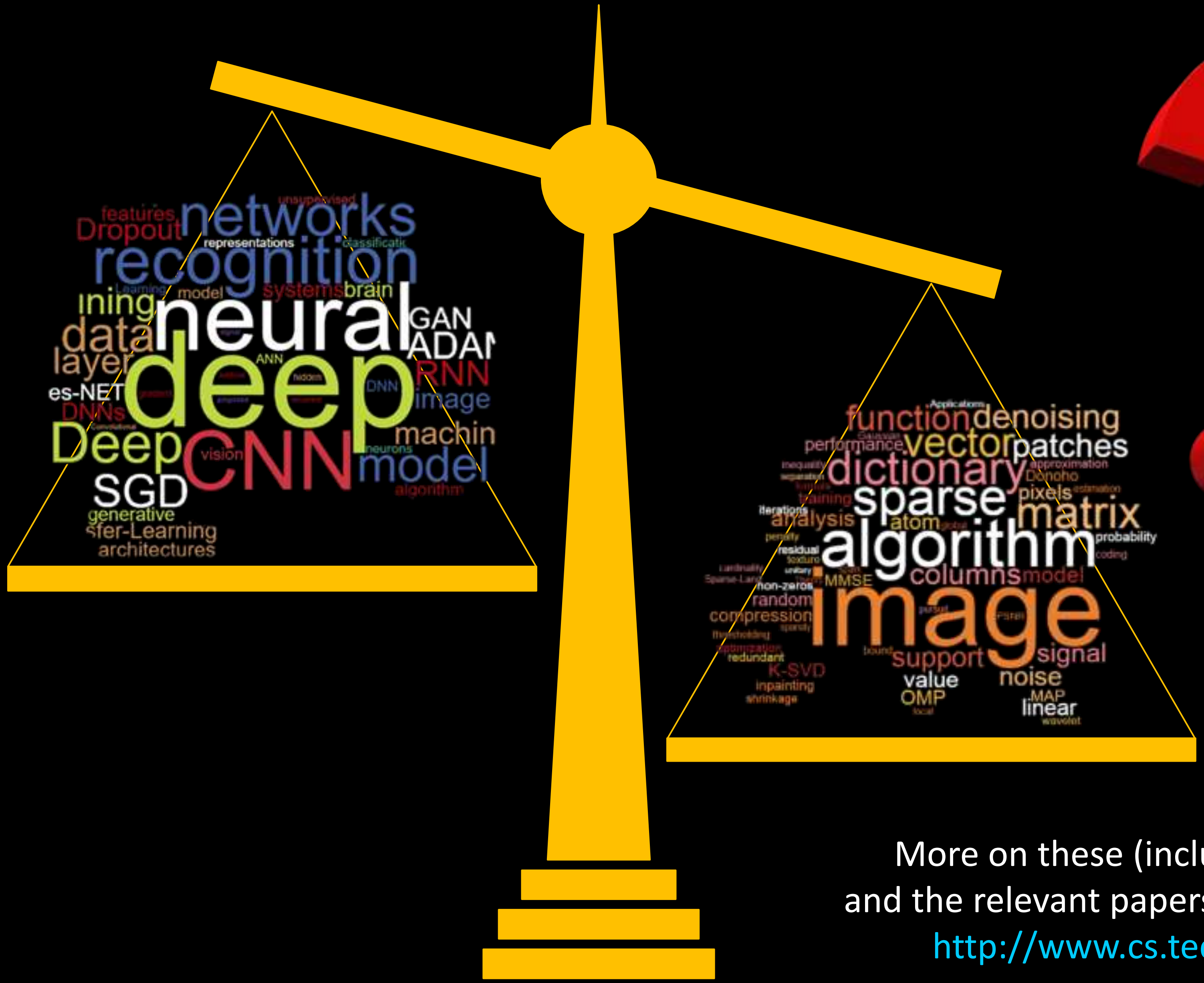
… this recent paper

Algorithm Unrolling: Interpretable, Efficient Deep Learning for Signal and Image Processing

Vishal Monga, *Senior Member, IEEE,* Yuelong Li, *Member, IEEE,* and Yonina C. Eldar, *Fellow, IEEE*

***Abstract*—Deep neural networks provide unprecedented performance gains in many real world problems in signal and image processing. Despite these gains, future development and practical deployment of deep networks is hindered by their black-box nature, i.e., lack of interpretability, and by the need for very large training sets. An emerging technique called algorithm unrolling or unfolding offers promise in eliminating these issues by providing a concrete and systematic connection between iterative algorithms that are used widely in signal processing and deep neural networks. Unrolling methods were first proposed to develop fast neural network approximations for sparse coding. More recently, this direction has attracted enormous attention and is rapidly growing both in theoretic investigations and practical applications. The growing popularity of unrolled deep networks is due in part to their potential in developing efficient, high-performance and yet interpretable network architectures from reasonable size training sets. In this article, we review**

model based analytic methods. In contrast to conventional iterative approaches where the models and priors are typically designed by analyzing the physical processes and handcrafting, deep learning approaches attempt to automatically discover model information and incorporate them by optimizing network parameters that are learned from real world training samples. Modern neural networks typically adopt a hierarchical architecture composed of many layers and comprise a large number of parameters (can be millions), and are thus capable of learning complicated mappings which are difficult to design explicitly. When training data is sufficient, this adaptivity enables deep networks to often overcome model inadequacies, especially when the underlying physical scenario is hard to characterize precisely.



More on these (including these slides
and the relevant papers) can be found in
http://www.cs.technion.ac.il/~elad