# Learning Discriminative Shrinkage Deep Networks for Image Deconvolution











Pin-Hung Kuo<sup>1</sup> Jinshan Pan<sup>2</sup> Shao-Yi Chien<sup>1</sup> Ming-Hsuan Yang<sup>3,4,5</sup> <sup>1</sup>National Taiwan University <sup>2</sup>Nanjing University of Science and Technology <sup>3</sup>Google Research <sup>4</sup>University of California, Merced <sup>5</sup>Yonsei University

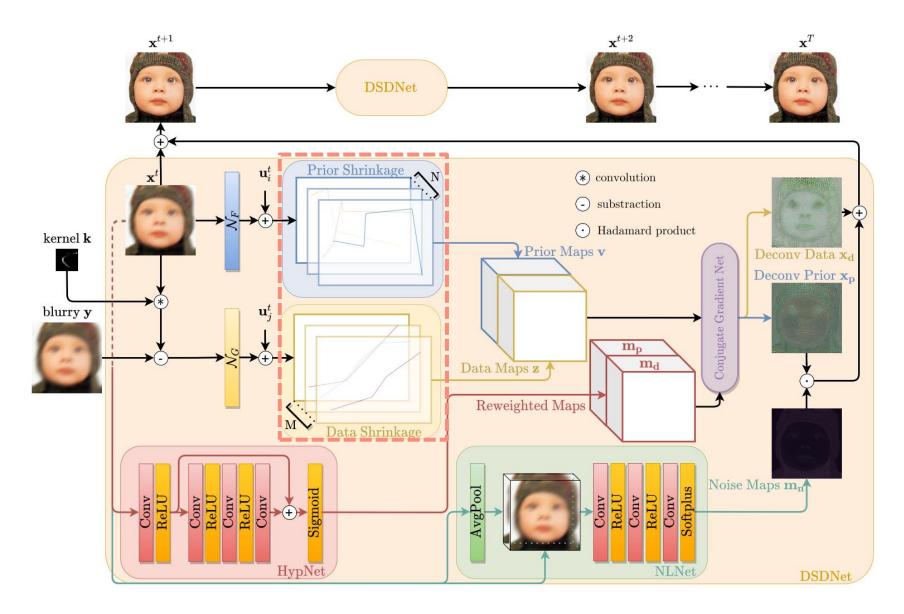


## Introduction

Goals: Design a deblurring deep learning model based on Maximum-a-Posteriori estimation. **Challenges:** 

- How to learn the **shrinkage functions** corresponding to various learned filters?
- With the regularization / data terms, how to solve the **deconvolution problem**?

# Architecture



# **ADMM**

We model the problem as:

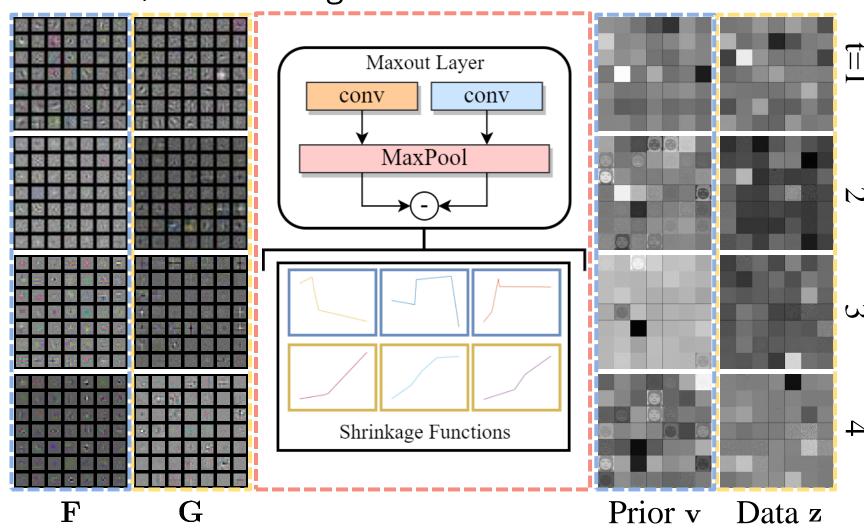
$$\min_{\mathbf{u}, \mathbf{v}, \mathbf{x}, \mathbf{z}} \sum_{i=1}^{N} R_i(\mathbf{v}_i) + \sum_{j=1+N}^{M+N} R_j(\mathbf{z}_j) \quad s.t. \quad \mathbf{F}_i \mathbf{x} = \mathbf{v}_i, \quad \mathbf{G}_j(\mathbf{y} - \mathbf{H}\mathbf{x}) = \mathbf{z}_j$$

and solve it by ADMM:

$$\begin{aligned} \mathbf{v}_{i}^{t+1} &= \mathbf{prox}_{\lambda_{i}R_{i}}(\mathbf{F}_{i}\mathbf{x}^{t} + \mathbf{u}_{i}^{t}), \\ \mathbf{z}_{j}^{t+1} &= \mathbf{prox}_{\lambda_{j}R_{j}}(\mathbf{G}_{j}(\mathbf{y} - \mathbf{H}\mathbf{x}^{t}) + \mathbf{u}_{j}^{t}), \\ &\left(\sum_{i=1}^{N} \rho_{i}\mathbf{F}_{i}^{\top}\mathbf{F}_{i} + \sum_{j=N+1}^{N+M} \rho_{j}\mathbf{H}^{\top}\mathbf{G}_{j}^{\top}\mathbf{G}_{j}\mathbf{H}\right)\mathbf{x}^{t+1} \\ &= \left(\sum_{i=1}^{N} \rho_{i}\mathbf{F}_{i}^{\top}(\mathbf{v}_{i}^{t+1} - \mathbf{u}_{i}^{t}) + \sum_{j=N+1}^{N+M} \rho_{j}\mathbf{H}^{\top}\mathbf{G}_{j}^{\top}(\mathbf{G}_{j}\mathbf{y} - \mathbf{z}_{j}^{t+1} + \mathbf{u}_{j}^{t})\right) \\ &\mathbf{u}_{i}^{t+1} = \mathbf{u}_{i}^{t} + \mathbf{F}_{i}\mathbf{x}^{t+1} - \mathbf{v}_{i}^{t+1}, \\ &\mathbf{u}_{j}^{t+1} = \mathbf{u}_{j}^{t} + \mathbf{G}_{j}(\mathbf{y} - \mathbf{H}\mathbf{x}^{t+1}) - \mathbf{z}_{j}^{t+1}. \end{aligned}$$

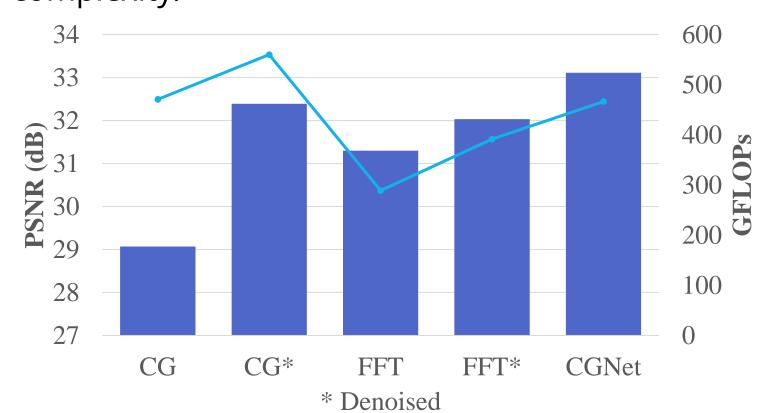
# Discriminative Shrinkage Function

As Maxout Layers can linearly approximate any function, the shrinkage functions are learned.



## CG Net

To solve the deconvolution problem, we design the CGNet to replace conventional CG or FFT. For FFT and CG, we test them w/o and w/ denoising. CGNet achieves the highest performance with reasonable complexity.

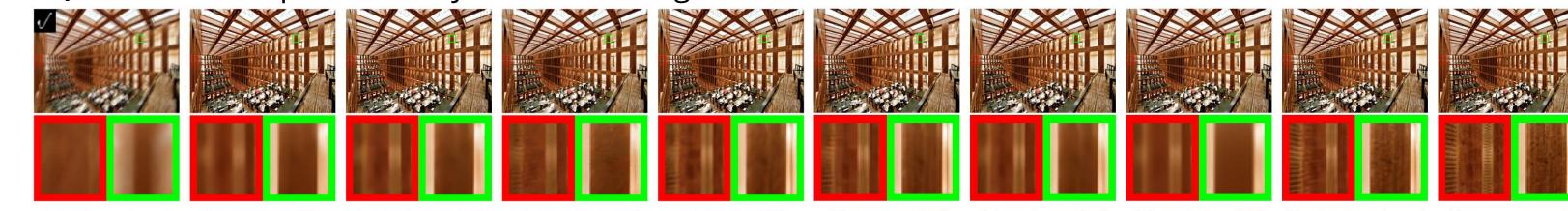


### Results

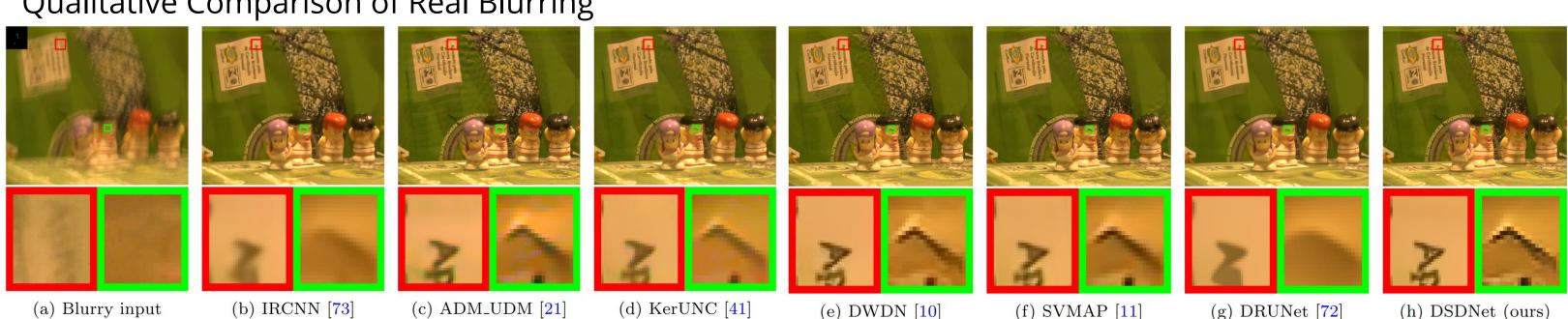
#### Quantitative Results on Benchmark Datasets

Dataset	noise	IRCNN [73] PSNR / SSIM	SFARL[48] PSNR / SSIM	ADM_UDM [21] PSNR / SSIM	CPCR [12] PSNR / SSIM	KerUNC [41] PSNR / SSIM	VEM [42] PSNR / SSIM	DWDN [10] PSNR / SSIM	SVMAP [11] PSNR / SSIM	DRUNet [72] PSNR / SSIM	DSDNet(Light) PSNR / SSIM	DSDNet(Full PSNR / SSIM
Levin [30]	3%	29.70 / 0.864	16.82 / 0.255	31.48 / 0.922 28.61 / 0.812 27.83 / 0.827	25.61 / 0.765	21.72 / 0.416	29.47 / 0.867	31.94 / 0.916	31.20 / 0.893	30.86 / 0.905	32.13 / 0.918	32.89 / 0.928
BSD100 [38]	1% 3% 5%	29.20 / 0.817 27.54 / 0.762 27.04 / 0.756	24.21 / 0.568 15.80 / 0.245 12.56 / 0.146	29.39 / 0.836 26.92 / 0.722 26.04 / 0.697	28.77 / 0.829 25.96 / 0.712 25.75 / 0.688	29.23 / 0.829 22.10 / 0.430 18.99 / 0.297	29.54 / 0.848 27.09 / 0.746 26.11 / 0.698	31.10 / 0.881 28.47 / 0.797 27.50 / 0.762	31.52 / 0.888 27.94 / 0.762 27.59 / 0.763	30.36 / 0.872 28.10 / 0.798 27.19 / 0.767	$\begin{array}{c} 31.50 \ / \ \underline{0.892} \\ \underline{28.73} \ / \ \underline{0.812} \\ \underline{27.64} \ / \ \underline{0.774} \end{array}$	32.01 / 0.898 29.08 / 0.820 27.96 / 0.788
Set 5 [3]	3%	28.66 / 0.813	15.50 / 0.211	30.52 / 0.868 27.64 / 0.709 26.75 / 0.756	27.94 / 0.799	21.39 / 0.376	28.40 / 0.804	29.54 / 0.838	28.78 / 0.812	29.21 / 0.841	29.94 / 0.843	30.40 / 0.85

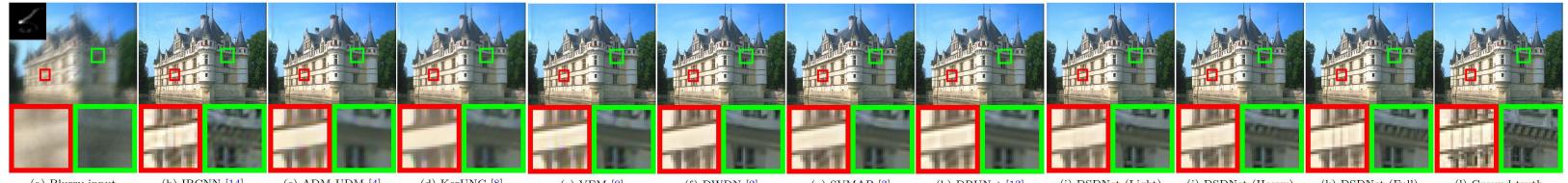
### Qualitative Comparison of Synthetic Blurring



Qualitative Comparison of Real Blurring



## Qualitative Comparison of Model Size



#### Speed vs Accuracy

33.5		• Full • Heavy
33		Light
32.5		Feather
BSNR (dB) 32 31.5		• SVMAP DWDN •
31		• VEM • DRUNet
30.5	•ADM_UDM	• KerUNC

Execution Time (sec)

#### Quantitative Results on Real Blurring

	IRCNN [73]	ADM_UDM [21]	KerUNC [41]	VEM [42]	DWDN [10]	SVMAP [11]	DRUNet [72]	DSDNet
BRISQUE	43.484	36.598	37.816	33.663	34.027	35.508	46.774	33.129
PIQE	78.700	67.605	65.674	44.942	51.348	56.032	81.074	49.788

## Specification of 4 Sizes

	Feather	Light	Heavy	Full
$\overline{T}$	2	3	3	4
M, N	$^{24}$	24	49	49





https://github.com/setsunil/DSDNet