Efficient Deep Image Denoising via Class Specific Convolution

Lu Xu[1], Jiawei Zhang[2], Xuanye Cheng[2], Feng Zhang[2], Xing Wei[3], Jimmy Ren[2,4]

lxu7@cse.cuhk.edu.hk, zhjw1988@gmail.com, jimmy.sj.ren@gmail.com

[1] The Chinese University of Hong Kong [2] SenseTime Research

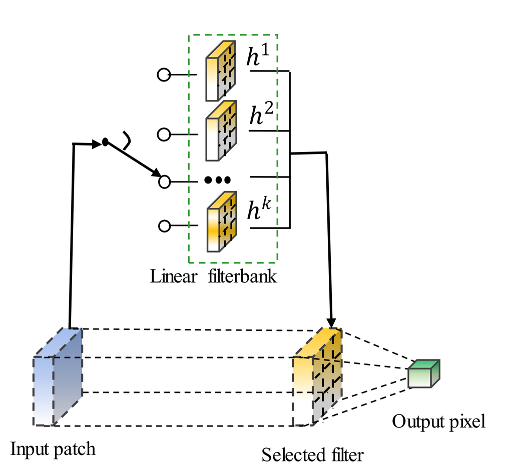
[3] Xi'an Jiaotong University [4] Qing Yuan Research Institute, Shanghai Jiao Tong University

Abstract

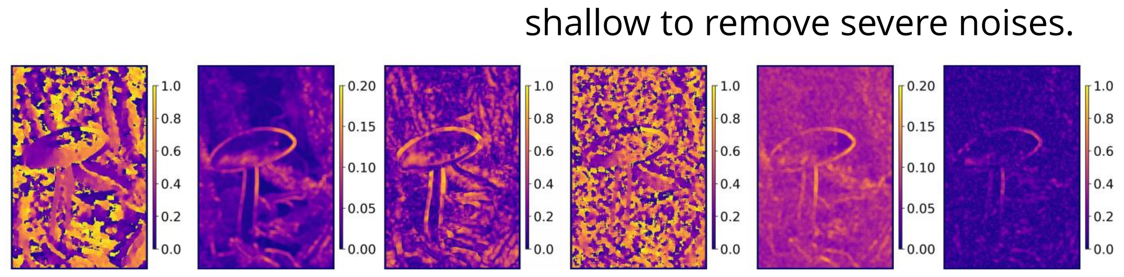
Deep neural networks have been widely used in image denoising during the past few years. Even though they achieve great success on this problem, they are computationally inefficient which makes them inappropriate to be implemented in mobile devices. In this paper, we propose an efficient deep neural network for image denoising based on pixel-wise classification. Despite using a computationally efficient network cannot effectively remove the noises from any content, it is still capable to denoise from a specific type of pattern or texture. The proposed method follows such a divide and conquer scheme. We first use an efficient U-net to pixel-wisely classify pixels in the noisy image based on the local gradient statistics. Then we replace part of the convolution layers in existing denoising networks by the proposed Class Specific Convolution layers (CSConv) which use different weights for different classes of pixels. Quantitative and qualitative evaluations on public datasets demonstrate that the proposed method can reduce the computational costs without sacrificing the performance compared to state-of-the-art algorithms.

Motivation

To achieve state-of-the-art performance, many deep network structures have been applied. However, the computational efficiency of the network remains to be improved for the sake of deployment on mobile devices.

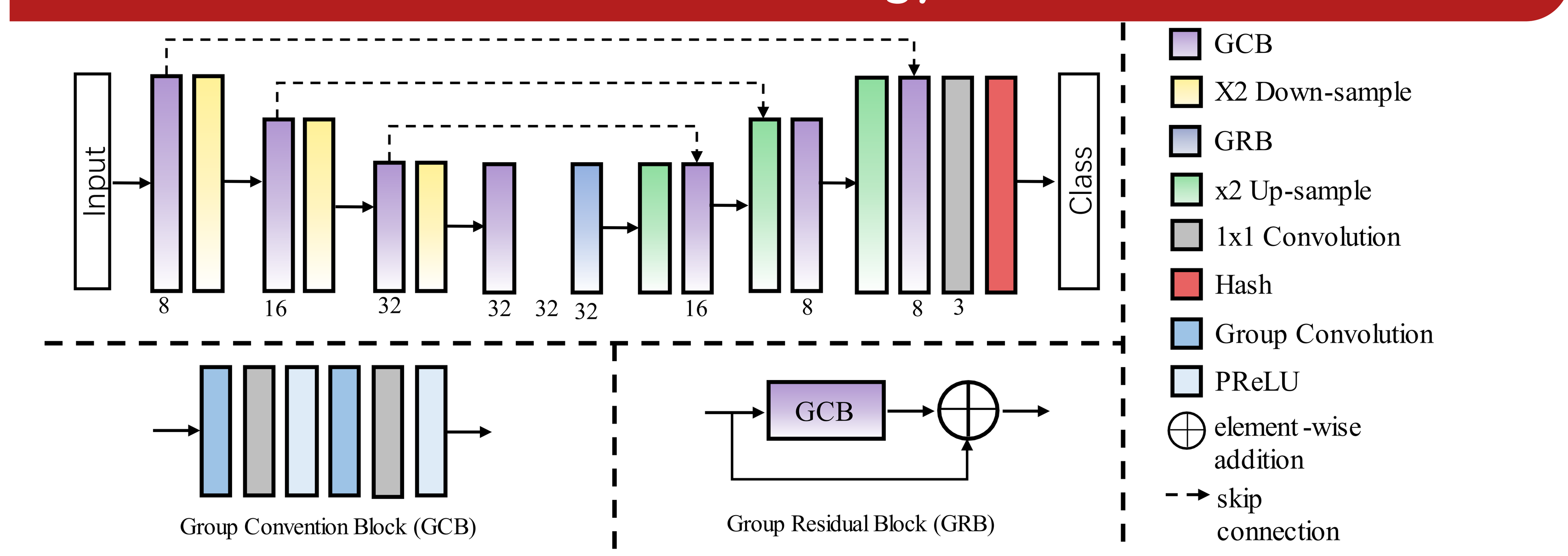


To address the computational efficiency issue, RAISR and BLADE classify the image patches into different buckets according to the local gradient statistics. Then only one specific linear filter is learned for every bucket via solving least squares to efficiently solve low-level vision tasks. However, RAISR and BLADE are equivalent to a single layer network which is too shallow to remove severe noises.

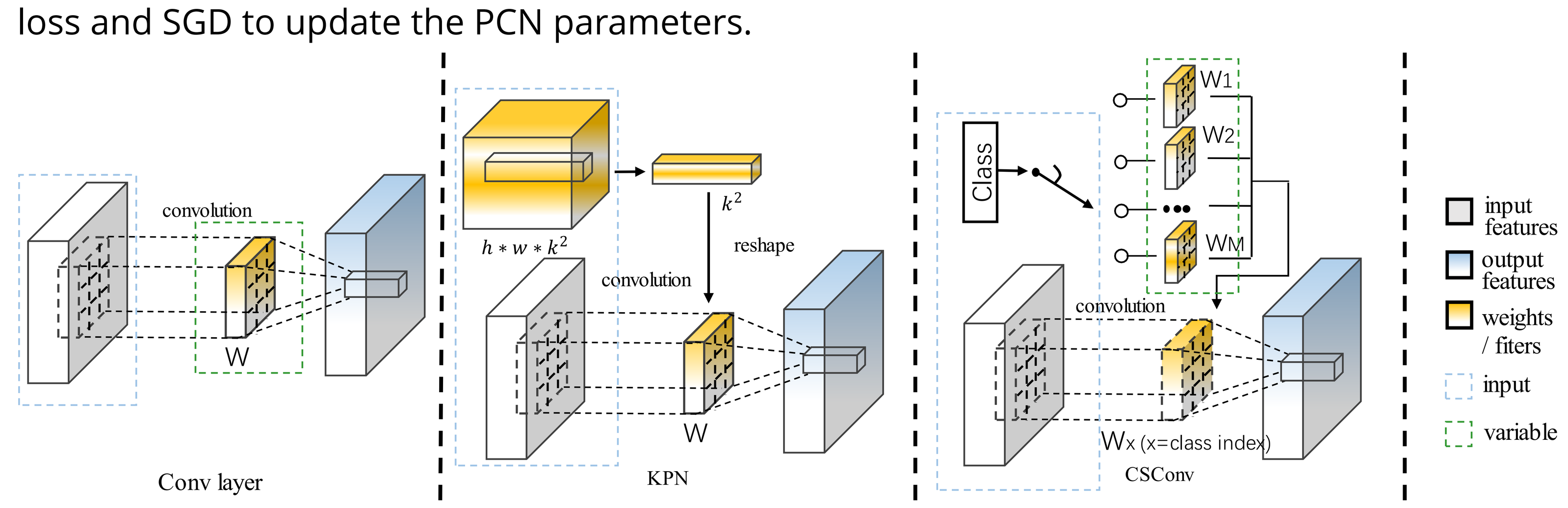


In addition, the local gradient statistics estimated from RAISR's eigenanalysis are not very accurate under noises, therefore it cannot distinguish different types of textures very well.

Methodology

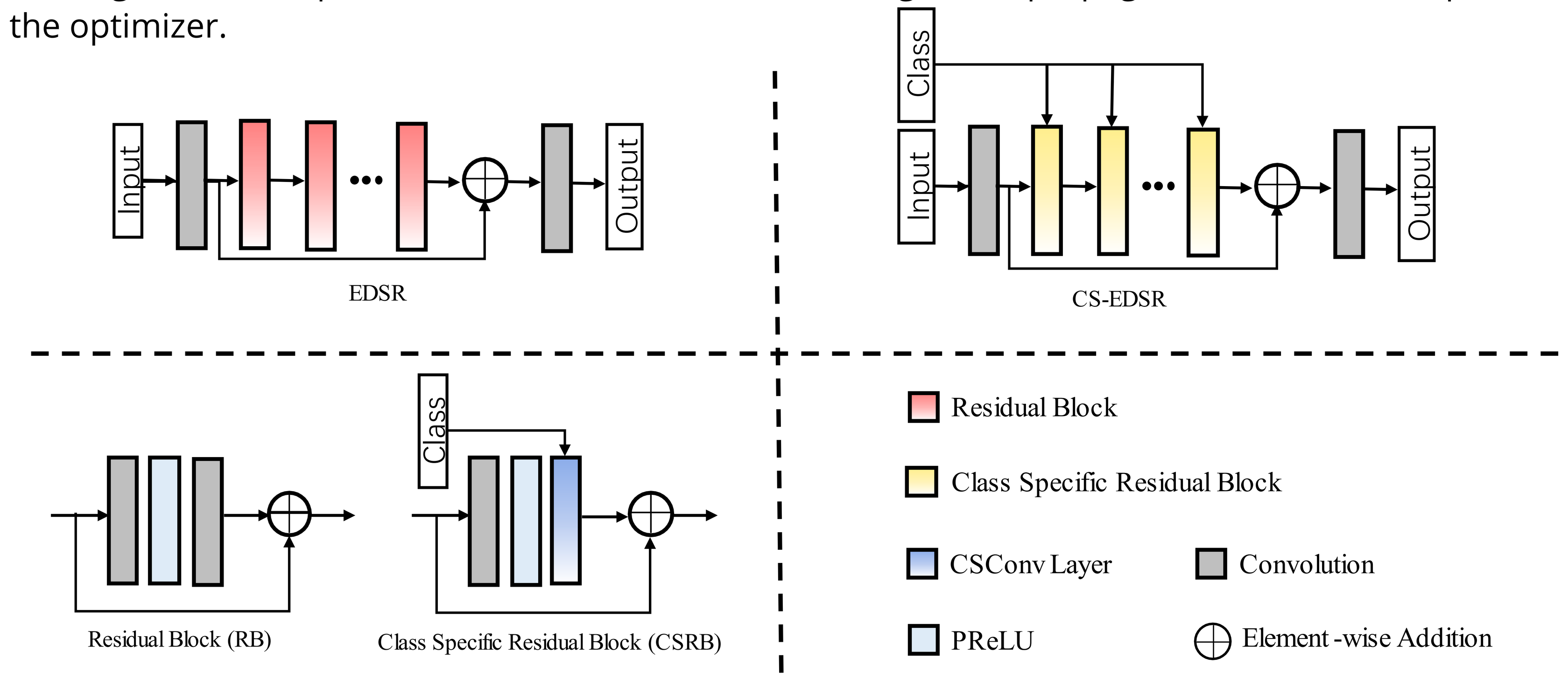


We propose a small Pixel-wise Classification Network (PCN) to predict the gradient statistics φ, λ and μ constrained by the estimated one from the ground truth clean image. PCN uses an architecture similar to U-Net. To train the PCN, we adopt an approach similar to RAISR to get the ground truth statistics. After PCN predicts the estimated noise-free statistics from noisy image input, we use L1 loss and SGD to update the PCN parameters.



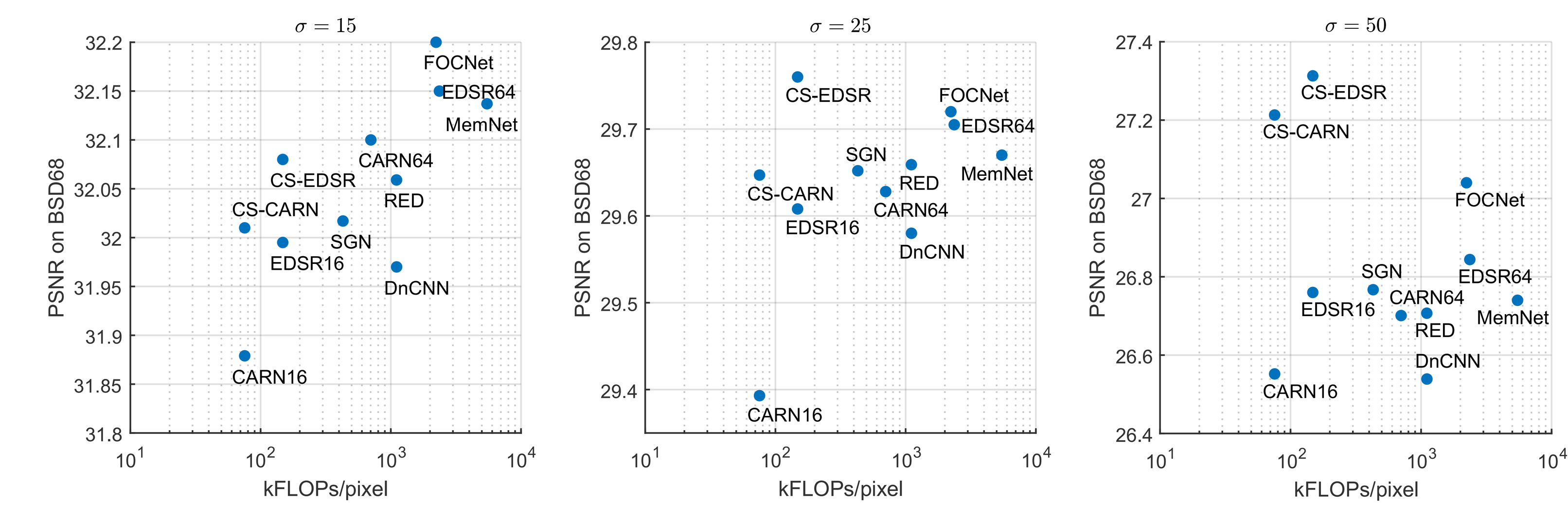
$$\hat{Q}(m,n,c)=\sum_{s=-r}^{r}\sum_{t=-r}^{r}\sum_{c'=0}^{c_{in}-1}[Q(m-s,n-t,c')\times W_i(s,t,c,c')]$$

We use the above PCN to classify pixels into different classes and then learn different weights for different classes in the proposed class specific convolution (CSConv). Specifically, the i-th learnable weights Wi of CSConv will be fetched from the filter bank W and used to filter the input feature Q(m, n, c) if the pixel at position (m, n) is classified as the i-th class from the noisy image. During training, the training loss will be passed to each chosen filter Wi through back propagation and then be updated by the optimizer.



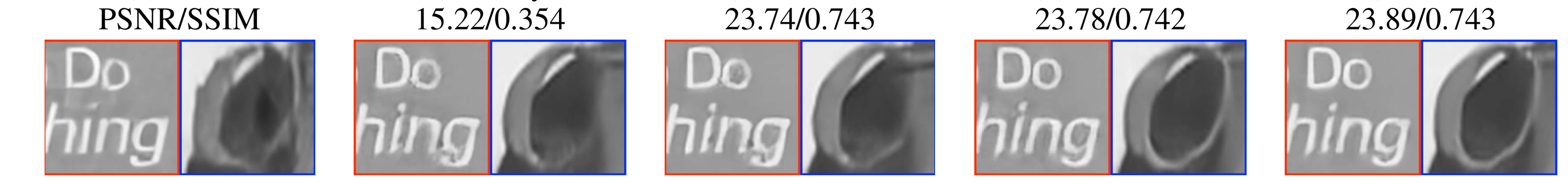
CSConv layers can be integrated into existing denoising network architectures by directly replacing the conv layers. We integrate the proposed CSConv into the EDSR and CARN architecture as the proposed CSConv-based denoising convolutional network (CSDN) in our experiments. To make the proposed CSDN more efficient, we reduce the filter channels by a factor of four.

Results



Dataset	σ	DnCNN	CARN16	EDSR16	CARN64	EDSR64	RED	MemNet	FOCNet	SGN	CS-CARN	CS-EDSR
BSD68	15	31.97/0.878	31.88/0.882	31.99/0.887	32.10/0.888	32.16/0.887	32.07/0.889	32.14/0.891	**32.20/0.893**	32.02/0.889	32.01/0.885	32.08/0.887
	25	29.58/0.811	29.36/0.812	29.61/0.822	29.63/0.818	29.70/0.822	29.66/0.824	29.67/0.825	29.72/0.819	29.65/0.824	29.65/0.820	**29.76/0.826**
	50	26.54/0.700	26.55/0.700	26.77/0.714	26.70/0.707	26.84/0.718	26.73/0.712	26.74/0.709	27.04/0.720	26.77/0.710	27.21/0.731	**27.31/0.738**
Urban100	15	32.85/0.920	32.21/0.917	32.75/0.921	32.95/0.920	33.10/0.931	32.59/0.923	32.50/0.919	**33.15/0.927**	32.57/0.911	32.76/0.921	32.93/0.925
	25	30.04/0.869	29.27/0.862	29.96/0.878	30.02/0.879	30.36/0.883	29.93/0.880	29.91/0.878	**30.64/0.887**	30.05/0.880	29.83/0.874	30.17/0.882
	50	26.30/0.772	25.77/0.755	26.44/0.781	26.33/0.774	26.85/0.794	26.29/0.779	26.34/0.778	**27.10**/0.803	26.35/0.781	26.76/0.799	27.07/**0.810**
DIV2K	15	33.93/0.901	33.75/0.901	33.93/0.905	34.04/0.906	**34.14/0.908**	34.04/0.907	33.99/0.905	-	34.02/0.902	33.95/0.904	34.03/0.906
	25	31.39/0.845	31.16/0.846	31.52/0.856	31.53/0.856	31.66/0.855	31.56/0.859	31.58/0.858	-	31.68/0.860	31.58/0.856	**31.74/0.861**
	50	28.25/0.753	28.18/0.751	28.51/0.765	28.40/0.759	28.65/0.766	28.53/0.769	28.43/0.753	-	28.53/0.771	29.02/0.782	**29.17/0.789**
Set12	15	33.16/0.887	32.90/0.893	33.11/0.897	33.16/0.896	33.34/**0.899**	33.11/0.891	33.15/0.897	**33.37**/0.896	33.07/0.881	33.14/0.896	33.18/0.898
	25	30.73/0.840	30.40/0.837	30.78/0.849	30.76/0.845	30.95/0.850	30.96/0.851	30.78/0.850	30.73/0.846	30.81/0.850	30.78/0.847	**30.97/0.853**
	50	27.44/0.747	27.28/0.737	27.64/0.755	27.58/0.748	27.86/0.762	27.50/0.766	27.48/0.749	27.99/0.765	27.62/0.758	28.16/0.773	**28.33/0.781**
kFLOPs/pixel		1108.2	72.6	145.1	702.7	2361.6	1106.4	5480.0	2225.7	429.1	75.5	148.0

(a) clean PSNR/SSIM (b) noisy 15.22/0.354 (c) DnCNN 23.74/0.743 (d) MemNet 23.78/0.742 (e) SGN 23.89/0.743



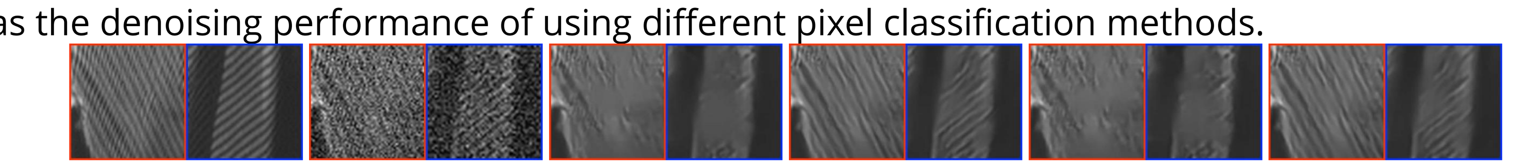
(f) FOCNet 24.04/0.749 (g) CARN64 23.87/0.739 (h) EDSR64 24.00/0.747 (i) CS-CARN 24.37/0.771 (j) CS-EDSR 24.46/0.778

In experiment, the proposed CS-EDSR and CS-CARN achieves competitive denoising performance on various datasets and noise levels, capable of better removing noises while preserving the correct shape and texture details while having much less computational cost than baselines.

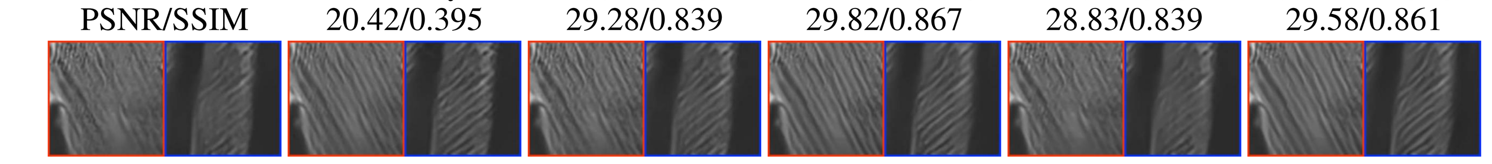
Dataset	PCN σ = 15	PCN σ = 25	PCN σ = 50	RAISR+N σ = 15	RAISR+N σ = 25	RAISR+N σ = 50
BSD68	0.030	0.031	0.035	0.052	0.062	0.075
Urban100	0.025	0.026	0.025	0.051	0.068	0.092
DIV2K	0.028	0.030	0.032	0.059	0.069	0.081
Set12	0.030	0.031	0.032	0.052	0.064	0.080

Dataset	EDSR PCN+N	EDSR RAISR+C	EDSR RAISR+N	CARN PCN+N	CARN RAISR+C	CARN RAISR+N
BSD68	29.76/0.826	30.22/0.847	29.42/0.814	29.65/0.820	29.90/0.838	29.36/0.812
Urban100	30.17/0.882	30.47/0.896	29.36/0.866	29.83/0.874	29.97/0.885	29.36/0.863
DIV2K	31.74/0.861	32.22/0.878	31.28/0.849	31.58/0.856	31.87/0.868	31.22/0.845
Set12	30.97/0.853	31.39/0.868	30.40/0.840	30.78/0.847	31.00/0.859	30.31/0.838

We verified the effectiveness of proposed Pixel-wise Classification network by comparison of mean square error of the estimated gradient statistics with RAISR statistics on noisy input, as well as the denoising performance of using different pixel classification methods.



(a) clean PSNR/SSIM (b) noisy 20.42/0.395 (c) B_8F_{16} 29.28/0.839 (d) CS-B_8F_{16} 29.82/0.867 (e) $B_{16}F_8$ 28.83/0.839 (f) CS-$B_{16}F_8$ 29.58/0.861



(g) $B_{16}F_{16}$ 30.01/0.871 (h) CS-$B_{16}F_{16}$ 30.21/0.877 (i) $B_{16}F_{32}$ 30.23/0.878 (j) CS-$B_{16}F_{32}$ 30.29/0.879 (k) $B_{32}F_{16}$ 30.16/0.873 (l) CS-$B_{32}F_{16}$ 30.28/0.879

We also compare the CSConv and vanilla convolution on EDSR architecture with different number of blocks and featuremap channels. When using fewer blocks or features, CS-EDSR still produces relatively clear details, whereas the results of EDSR are over-smoothed.

Conclusion & References

- We present a way of deep image denoising by divide and conquer
- A classification network is used for robust and efficient pixel classification
- CSConv layers are proposed for efficient denoising with spatially variant kernels
- Competitive performance is achieved with much less computational cost

[1] Romano, Isidoro, and Milanfar, RAISR: Rapid and accurate image super resolution, TCI 2016.
[2] Getreuer et al., BLADE: Filter Learning for General Purpose Image Processing, ICCP 2018.