A review of an old dilemma: demosaicking first, or denoising first? Supplementary Material

Qiyu Jin
School of mathematical science,
Inner Mongolia University
qyjin2015@aliyun.com

Gabriele Facciolo
Centre Borelli,
ENS Paris-Saclay, CNRS
facciolo@cmla.ens-cachan.fr

Jean-Michel Morel
Centre Borelli,
ENS Paris-Saclay, CNRS
moreljeanmichel@gmail.com

Evaluation of A&DMand B&DNschemes. Figures 1 and 2 present the results (corresponding to Fig.2 in the main article) comparing the schemes: denoising then demosaicking ($DN\&DM$), demosaicking then denoising ($DM\&DN$), and demosaicking then denoising with a noise parameter set to $1.5\sigma_0$ ($DM\&1.5DN$). Two different different denoising methods are considered CBM3D (Dabov *et al.* 2007) and nlBayes (Lebrun *et al.* 2013).

We see that for both CBM3D and nlBayes the $DN\&DM$ schemes lead to a loss of detail, while $DM\&DN$ has a strong residual noise. Denoising after demosaicking with a noise parameter equal to $C\,\sigma_0$ with $C = 1.5$, while noise in the raw image has standard deviation $\sigma_0 = 20$ preserves much more detail ($DM\&1.5DN$).

Selection of the optimal noise scaling factor for B&DNschemes. Table 1, which is a more complete version of Table 1 in the main article, allows to confirm the choice of the factor $C = 1.5$ in the $DM\&DN$ schemes for different levels of noise and for different choices of demosaicking algorithms.

Analysis of demosaicking noise. Table 2 and 3 are more complete versions of Tables 3 and 4 in the main article as they include the RI method. Table 2 shows that the standard deviation of the demosaicking noise is higher in the Y direction, and lower in the chromatic directions C_1 and C_2. Table 3 shows that after demosaicking the RGB color channels are more correlated for more sophisticated methods.

Evaluation of demosaicking algorithms for the B&DNschemes. Lastly, Table 4 presents a more complete version of Table 5 in the main article. It confirms that the $DM\&1.5DN$ schemes yield the best results for different noise levels for both denoising algorithms CBM3D and nlBayes.

Additional results. In Figures 3 and 4 we present additional results obtained on the Kodak and Imax datasets, respectively. Note that, compared with the best $DN\&DM$ scheme (BM3D+RCNN), the proposed $DM\&1.5DN$ schemes (RCNN+CBM3D, RCNN+nlBayes, and MLRI+CBM3D) yield the best results terms of CPSNR and detail preservation. The results are comparable to those of the joint demosaicking and denoising CNN (JCNN) by (Gharbi *et al.* 2016).

In Figure 3 the detail in the upper left corner we see that the result of BM3D+RCNN is too smooth, MLRI+CBM3D introduced some artifacts, but RCNN+CBM3D and RCNN+nlBayes preserve the details and don't introduce artifacts. In the detail in the lower left corner, BM3D+RCNN smooths all the details while RCNN+CBM3D, RCNN+nlBayes and MLRI+CBM3D retain much more detail.

In Figure 4, the detail in the upper right corner highlights the chessboard effect introduced by JCNN and BM3D+RCNN, the restored images using $DM\&DN$ scheme-based methods don't have any chessboard Effect.

Results on real raw images from the SIDD dataset. We evaluate the proposed framework on real images from the Smartphone Image Denoising Dataset (SIDD) (Abdelhamed *et al.* 2018), which provides pairs of raw noisy images and ground truth photo-finished images. We evaluate the best performing methods from the synthetic image evaluation, namely RCNN+CBM3D for the $DM\&1.5DN$ schemes and BM3D+RCNN for the $DN\&DM$ scheme, as well as the JCNN method as reference.

Table 5 reports the CPSNRs for every image as well as the estimated whitened noise level σ_0. In Fig. 5 we compare some results.

The authors of (Abdelhamed *et al.* 2018) provide a simple pipeline for converting raw images to photo-finished ones. The steps of the pipeline are:

Simple raw to photo-finished pipeline without denoising

1. black level correction and dark frame removal
2. white balance
3. demosaicking
4. colorspace conversion from camera to XYZ
5. tone curve

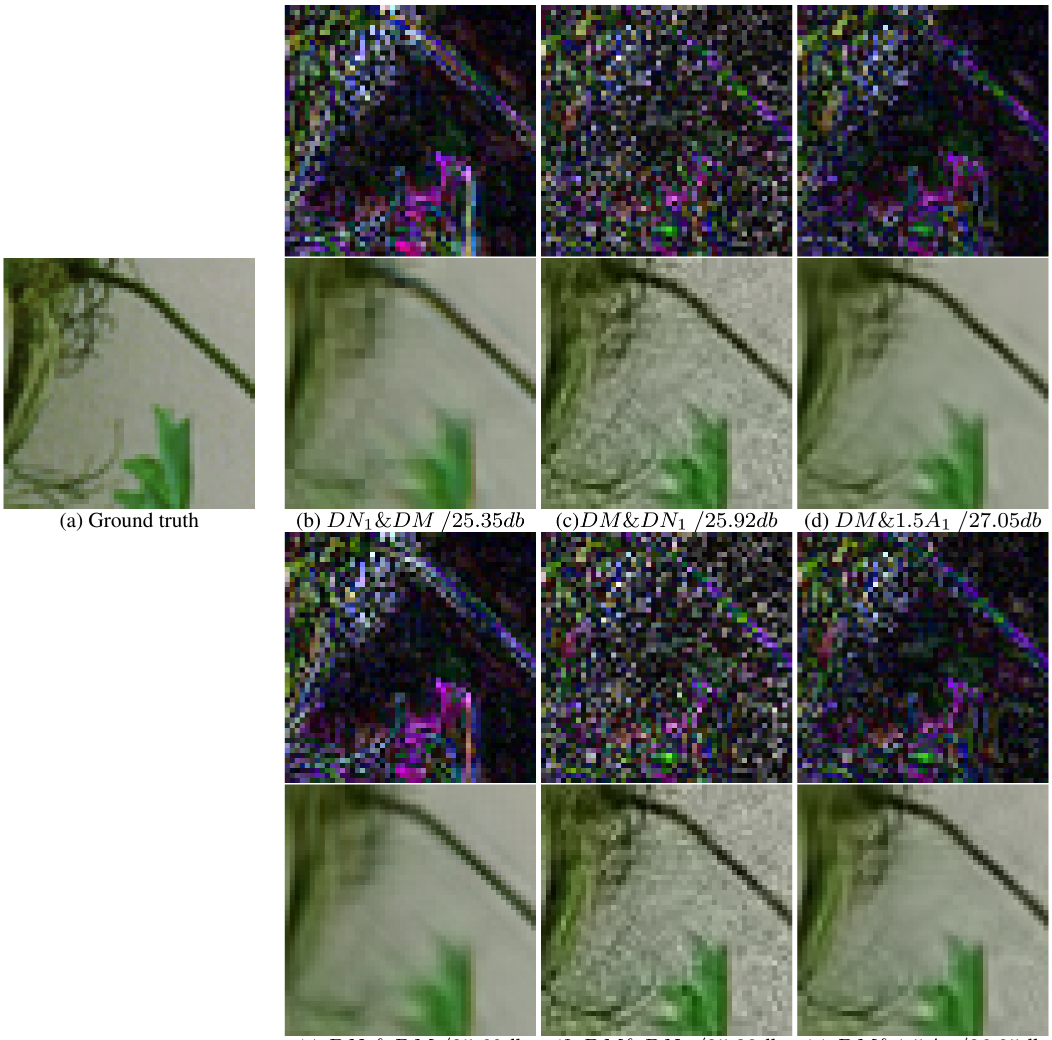
In our experiments we adapted the pipeline by replacing the demosaicking step by our $DN\&DM$ or $DM\&1.5DN$ method. The noise level is estimated using the algorithm by (Ponomarenko *et al.* 2007) on variance stabilized images (VST). We adopt a simple VST, a squared root. These adaptations are detailed in the blocks below.

Pipeline for $DN\&DM$

1. black level correction and dark frame removal
2. white balance
   - 3.1 estimate noise: σ_0.
     A VST (squared root) must be applied to the image, but not propagated to the next step.
   - 3.3 apply VST;
     apply CFA denoiser with parameter $1.5\sigma_0$;
     undo the VST;
   - 3.2 apply demosaicking
4. colorspace conversion from camera to XYZ
5. tone curve

Pipeline for $DM\&1.5DN$

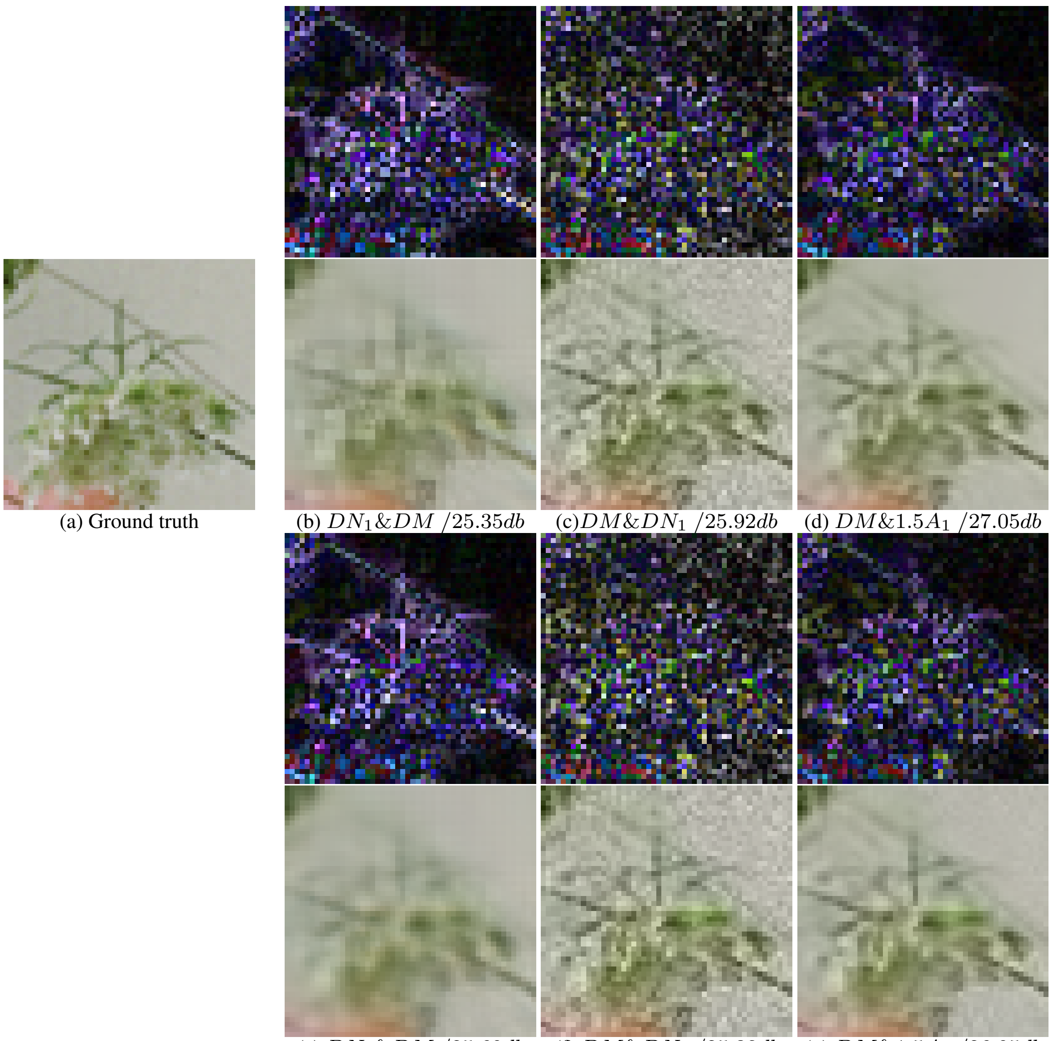
1. black level correction and dark frame removal
2. white balance
   - 3.1 estimate noise: σ_0.
     A VST (squared root) must be applied to the image, but not propagated to the next step.
   - 3.2 apply demosaicking
   - 3.3 apply VST;
     apply color denoiser with parameter $1.5\sigma_0$;
     undo the VST;
4. colorspace conversion from camera to XYZ
5. tone curve



(a) Ground truth | (b) $DN_1\&DM$ /25.35db | (c)$DM\&DN_1$ /25.92db | (d) $DM\&1.5A_1$ /27.05db

(e) $DN_2\&DM$ /25.69db | (f) $DM\&DN_2$ /25.38db | (g) $DM\&1.5A_2$ /26.95db

Figure 1. Image detail 1 with noise level $\sigma_0 = 20$. In each experiment: below, the denoised image, above, the difference with original that should contain mainly noise. DN_1: nlBayes denoising; DN_2: CBM3D denoising; DM: demosaicking (here we use RCNN). $1.5DN$ means that if noise level is σ_0, the input noise level parameter of denoising method DN is $\sigma = 1.5\sigma_0$; $DN_2\&DM$: is computed using the BM3D-CFA method (Danielyan *et al.* 2009).



(a) Ground truth

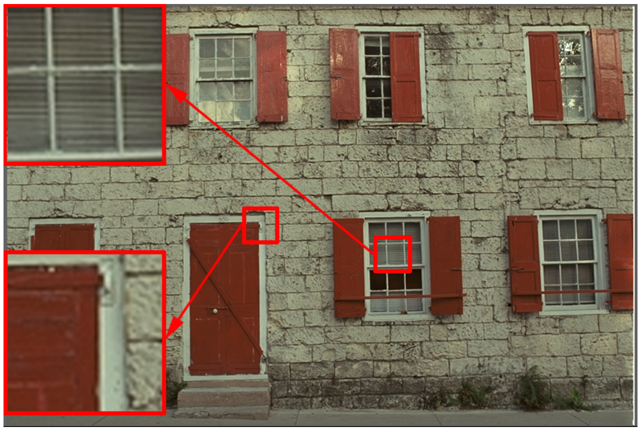
(b) $DN_1\&DM$ /$25.35db$ (c)$DM\&DN_1$ /$25.92db$ (d) $DM\&1.5A_1$ /$27.05db$

(e) $DN_2\&DM$ /$25.69db$ (f) $DM\&DN_2$ /$25.38db$ (g) $DM\&1.5A_2$ /$26.95db$

Figure 2. Image detail 2 with noise level $\sigma_0 = 20$. In each experiment: below, the denoised image, above, the difference with original that should contain mainly noise. DN_1: nlBayes denoising; DN_2: CBM3D denoising; DM: demosaicking (here we use RCNN). $1.5DN$ means that if noise level is σ_0, the input noise level parameter of denoising method DN is $\sigma = 1.5\sigma_0$; $DN_2\&DM$: is computed using the BM3D-CFA method (Danielyan *et al.* 2009).

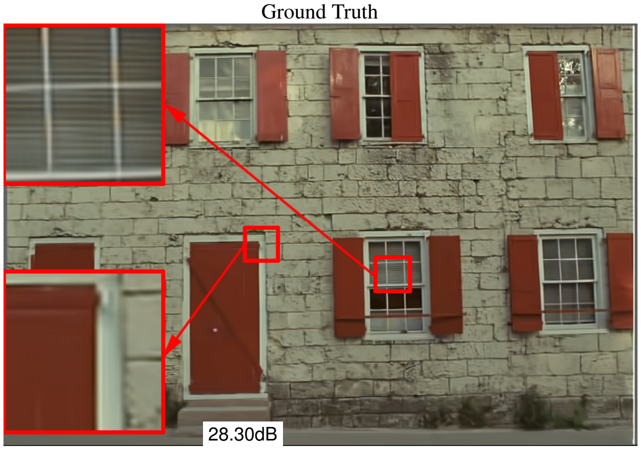
Table 1. Denoising after demosaicking $DM\&DN$, where DM are different algorithms (in columns) and DN is CBM3D with noise parameter equal to $C\,\sigma_0$, while noise in the raw image has standard deviation $\sigma_0 = 5, 10, 20, 40$. Each row shows the CPSNR result for C ranging from 1.0 to 1.9. Each column correspond to a different demosaicking method DM. The best result of each column is in **red**, the second best is in **green** and the third is in **blue**. The best factor C is clearly $C \simeq 1.5$. This seems to imply that the "demosaicked noise" has a 1.5 times higher standard deviation than the initial noise. This is actually true for the Y component of the noise, as it is shown in the main article.

σ_0	factor	HA	GBTF	RI	MLRI	ARI	LSSC	RCNN
	1.0	33.50	32.89	34.13	34.13	34.63	33.60	34.79
	1.1	33.66	33.07	34.28	34.29	34.75	33.76	35.01
	1.2	33.79	33.21	34.39	34.43	34.83	33.89	35.19
	1.3	33.88	33.31	34.47	34.52	34.89	33.99	35.30
	1.4	33.96	33.40	**34.52**	34.58	**34.92**	34.06	**35.37**
5	1.5	33.99	33.44	**34.54**	**34.60**	**34.92**	34.10	**35.39**
	1.6	**34.02**	**33.47**	**34.54**	**34.60**	**34.91**	**34.12**	**35.38**
	1.7	**34.02**	**33.48**	**34.52**	**34.59**	34.89	**34.13**	35.35
	1.8	**34.02**	**33.48**	34.49	34.56	34.85	**34.12**	35.30
	1.9	34.00	33.46	34.44	34.51	34.80	34.10	35.24
	1.0	31.44	30.76	31.76	31.52	32.06	31.02	31.38
	1.1	31.71	31.13	32.02	31.84	32.26	31.38	31.88
	1.2	31.91	31.41	32.20	32.08	32.40	31.65	32.27
	1.3	32.05	31.60	32.31	32.24	32.48	31.84	32.54
	1.4	32.13	31.73	**32.37**	**32.33**	**32.52**	31.98	**32.70**
10	1.5	**32.16**	**31.77**	**32.37**	**32.36**	**32.52**	**32.03**	**32.74**
	1.6	**32.17**	**31.79**	**32.35**	**32.34**	**32.49**	**32.05**	**32.74**
	1.7	**32.14**	**31.77**	32.29	32.30	32.43	**32.03**	32.69
	1.8	32.10	31.72	32.23	32.24	32.37	31.99	32.61
	1.9	32.05	31.67	32.15	32.17	32.29	31.94	32.52
	1.0	28.15	27.58	28.46	27.95	28.70	27.19	27.28
	1.1	28.56	28.15	28.83	28.44	28.98	27.89	28.05
	1.2	28.85	28.55	29.08	28.80	29.18	28.43	28.67
	1.3	29.05	28.81	29.23	29.03	29.29	28.78	29.09
	1.4	29.18	28.96	**29.31**	29.17	**29.35**	29.00	29.34
20	1.5	**29.23**	**29.00**	**29.32**	**29.22**	**29.35**	**29.06**	**29.41**
	1.6	**29.25**	**29.01**	**29.30**	**29.23**	**29.33**	**29.06**	**29.41**
	1.7	**29.25**	**28.97**	29.26	**29.20**	29.29	**29.02**	**29.36**
	1.8	29.22	28.92	29.20	29.15	29.23	28.95	29.28
	1.9	29.17	28.85	29.13	29.08	29.17	28.88	29.20
	1.0	24.00	23.62	24.42	23.79	24.67	23.18	23.05
	1.1	24.86	24.68	25.19	24.81	25.23	24.48	24.47
	1.2	25.18	25.05	25.39	25.13	25.40	24.99	25.08
	1.3	25.37	25.24	25.49	25.30	25.48	25.25	25.40
	1.4	25.49	**25.32**	**25.53**	**25.38**	**25.51**	**25.34**	**25.52**
40	1.5	**25.53**	**25.32**	**25.52**	**25.39**	**25.51**	**25.34**	**25.52**
	1.6	**25.54**	**25.29**	**25.50**	**25.37**	**25.49**	**25.30**	**25.47**
	1.7	**25.53**	25.25	25.46	25.33	25.46	25.25	25.41
	1.8	25.51	25.20	25.41	25.28	25.42	25.19	25.35
	1.9	25.48	25.14	25.37	25.23	25.39	25.13	25.28

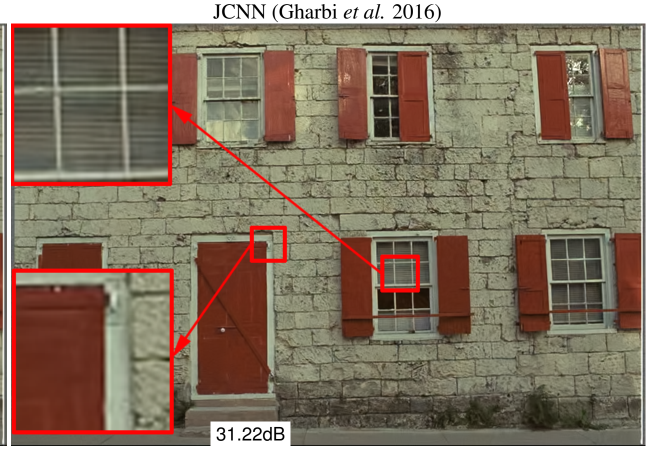


Ground Truth

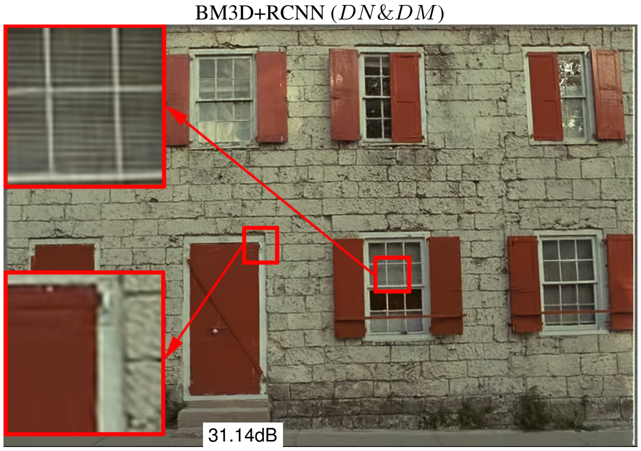
JCNN (Gharbi *et al.* 2016)



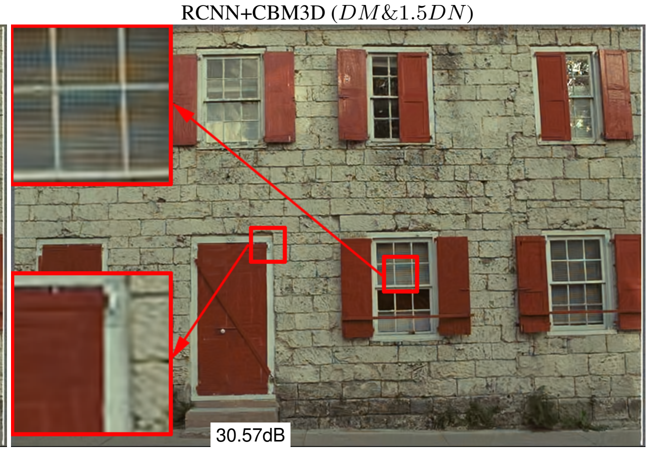
BM3D+RCNN ($DN\&DM$)



RCNN+CBM3D ($DM\&1.5DN$)



RCNN+nlBayes ($DM\&1.5DN$)



MLRI+CBM3D ($DM\&1.5DN$)

Figure 3. Demosaicking and denoising results on an image from the Kodak dataset with $\sigma = 10$. We compare an $DN\&DM$ scheme BM3D+RCNN (Danielyan *et al.* 2009), with three $DM\&1.5DN$ RCNN+CBM3D, RCNN+nlBayes and MLRI+CBM3D. As a reference we also include the result of a joint CNN method JCNN (Gharbi *et al.* 2016). Notice that JCNN does not work for σ larger than 20.

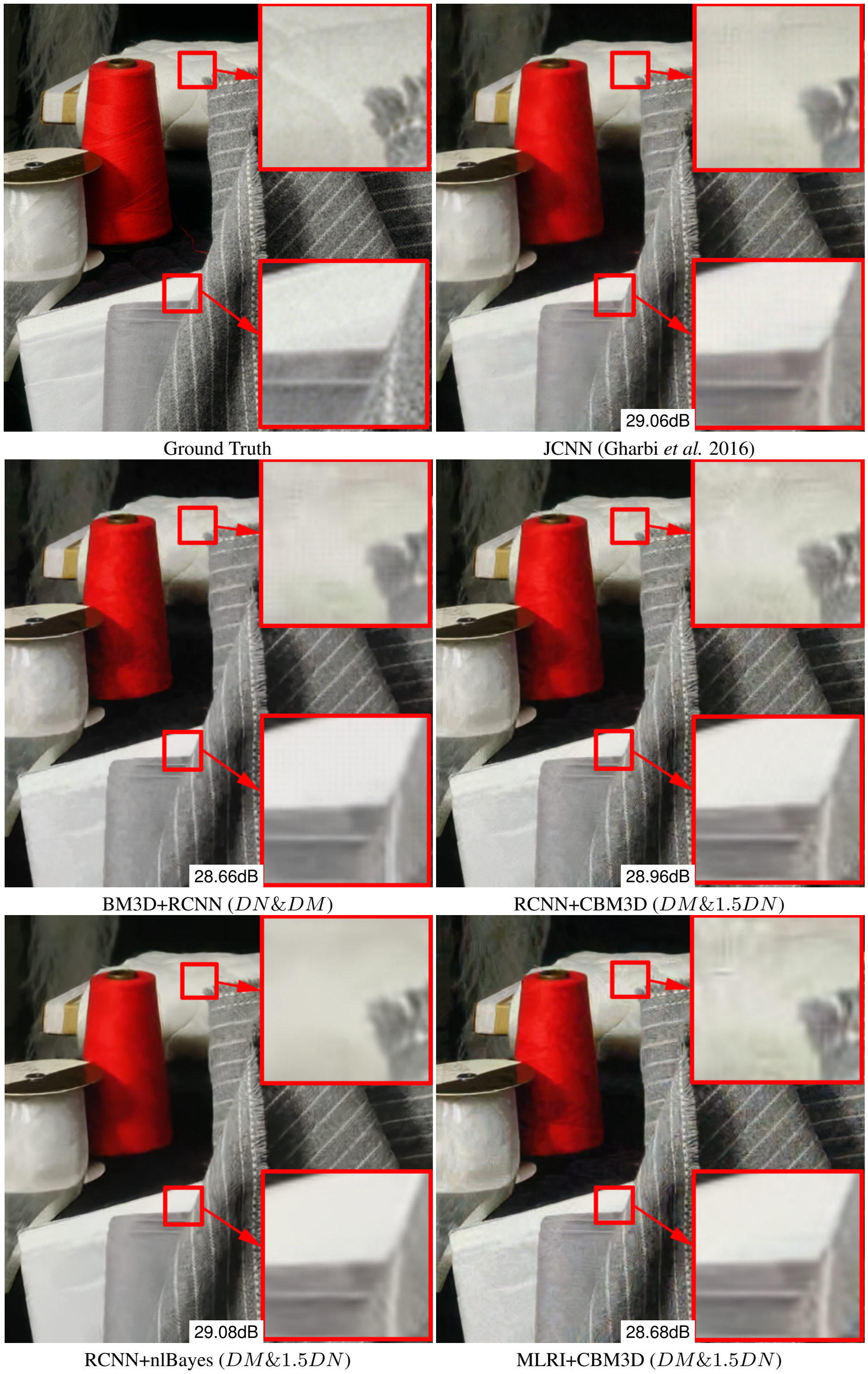


Figure 4. Demosaicking and denoising results on an image from the Imax dataset with $\sigma = 20$. We compare an $DN\&DM$ scheme BM3D+RCNN (Danielyan *et al.* 2009), with three $DM\&1.5DN$ RCNN+CBM3D, RCNN+nlBayes and MLRI+CBM3D. As a reference we also include the result of a joint CNN method JCNN (Gharbi *et al.* 2016). Notice that JCNN does not work for σ larger than 20.

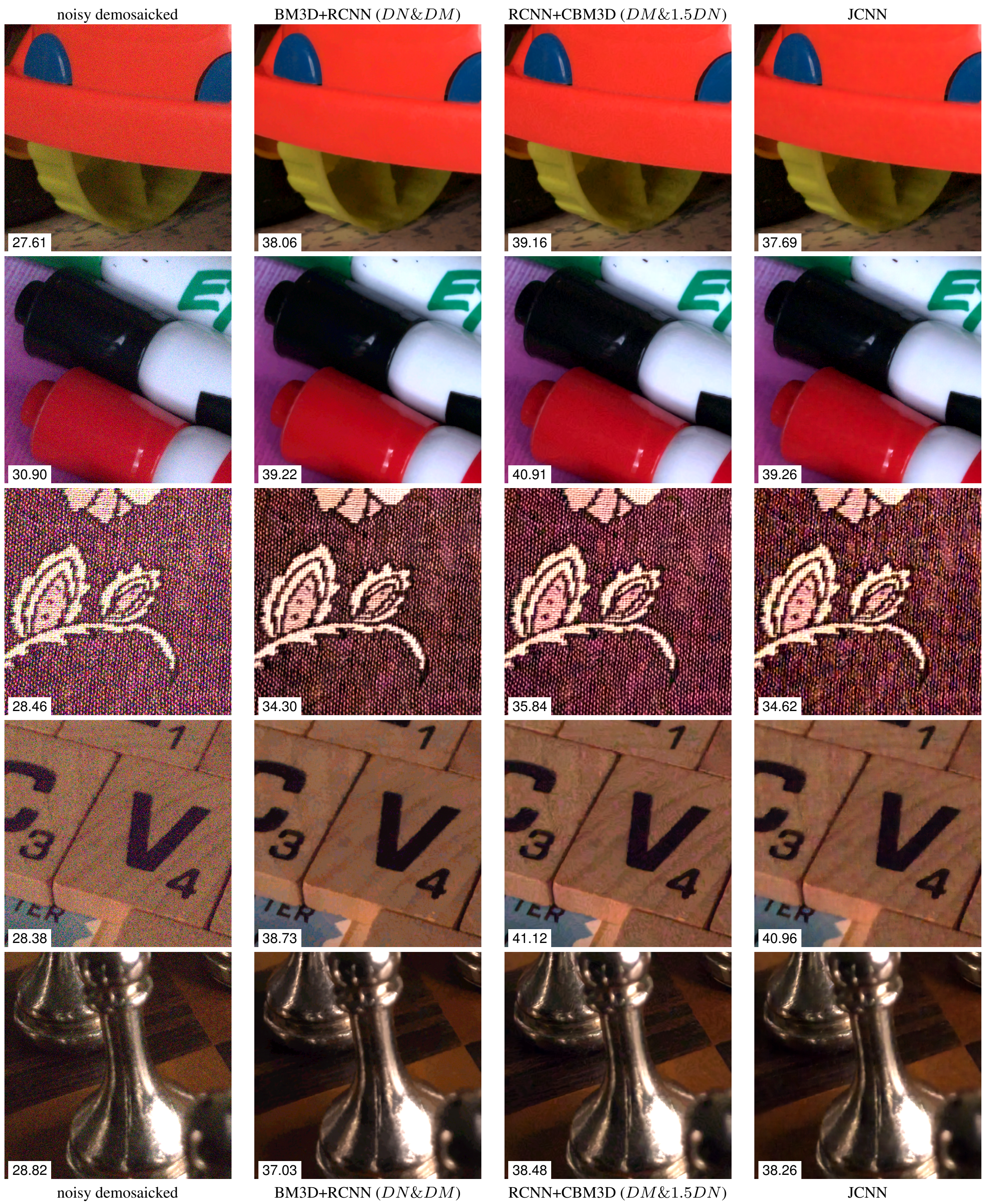


Figure 5. Results on images from the SIDD dataset (Abdelhamed *et al.* 2018). We compare the $DM\&1.5DN$ scheme RCNN+CBM3D, the $DN\&DM$ scheme BM3D+RCNN, and JCNN.

Table 2. Variance and covariance of (R, G, B) and (Y, U, V) (each first row) and the corresponding correlations (each second row) between pixels (i, j) and $(i + s, j + t)$, $s, t = 0, 1, 2$ first for AWGN (a) with standard deviation $\sigma = 20$, then for its demosaicked versions by RI (b), MLRI (c) and RCNN (d).

	(i,j)	*(i,j+1)*	*(i,j+2)*	*(i+1,j)*	*(i+1,j+1)*	*(i+1,j+2)*	*(i+2,j)*	*(i+2,j+1)*	*(i+2,j+2)*
R	400.6	0.6	0.4	0.7	0.1	0.7	0.3	0.2	0.8
G	401.7	0.5	1.1	0.1	0.3	0.9	1.0	0.6	0.4
B	400.2	1.2	0.1	0.5	0.6	0.0	1.9	0.3	1.9
Y	399.6	1.1	0.1	0.3	0.1	0.9	0.2	0.5	1.2
C_1	401.5	0.1	0.8	0.6	0.3	0.3	0.9	0.5	1.3
C_2	401.4	0.2	1.8	0.9	0.2	1.0	0.6	0.2	0.2

(a) AWG noise

	(i,j)	*(i,j+1)*	*(i,j+2)*	*(i+1,j)*	*(i+1,j+1)*	*(i+1,j+2)*	*(i+2,j)*	*(i+2,j+1)*	*(i+2,j+2)*
R	336.4	126.8	19.4	129.9	52.9	21.6	20.7	22.4	18.7
G	295.5	92.5	0.5	95.6	20.6	1.8	0.7	1.5	4.3
B	350.5	125.9	18.1	130.4	50.7	20.8	20.0	20.9	17.5
Y	715.6	170.9	32.3	178.6	2.6	5.4	34.0	7.1	20.5
C_1	168.4	108.3	41.3	110.1	73.4	28.2	44.1	29.4	9.7
C_2	98.3	66.0	27.9	67.3	48.1	21.4	29.9	22.4	10.4

(b) RI

	(i,j)	*(i,j+1)*	*(i,j+2)*	*(i+1,j)*	*(i+1,j+1)*	*(i+1,j+2)*	*(i+2,j)*	*(i+2,j+1)*	*(i+2,j+2)*
R	361.4	128.4	18.9	130.5	46.4	20.6	21.6	21.5	19.8
G	298.9	93.0	0.5	95.1	19.1	0.9	1.0	0.5	3.8
B	370.9	127.8	19.3	130.4	46.0	20.6	21.2	20.3	19.0
Y	772.2	177.7	33.0	181.3	9.6	9.2	32.6	10.9	21.4
C_1	164.8	107.1	43.7	108.8	72.8	29.3	46.1	30.2	10.1
C_2	94.3	64.4	28.1	65.8	48.2	21.9	30.3	23.1	11.1

(c) MLRI

	(i,j)	*(i,j+1)*	*(i,j+2)*	*(i+1,j)*	*(i+1,j+1)*	*(i+1,j+2)*	*(i+2,j)*	*(i+2,j+1)*	*(i+2,j+2)*
R	359.9	47.8	5.0	51.9	21.8	17.8	5.1	19.4	9.2
G	354.8	32.6	4.4	36.3	5.8	8.4	6.4	8.8	0.6
B	356.0	49.6	6.3	53.7	23.6	18.8	7.3	19.4	9.2
Y	972.3	69.0	20.8	76.4	3.6	18.6	28.9	17.3	2.2
C_1	55.1	33.8	15.3	36.0	26.1	14.6	19.0	16.6	11.8
C_2	43.3	27.3	12.3	29.4	21.5	11.7	16.0	13.7	9.4

(d) RCNN

Table 3. Covariances (each first row) and *correlations* (each second row) of the three color channels (R, G, B) of the demosaicked noise, when the initial CFA white noise satisfies $\sigma_0 = 20$.

	R	G	B
R	336.44	206.29	175.01
	1.0000	*0.6542*	*0.5097*
G	206.29	295.54	200.96
	0.6542	*1.0000*	*0.6244*
B	175.01	200.96	350.46
	0.5097	*0.6244*	*1.0000*

(a) RI

	R	G	B
R	361.42	224.39	201.41
	1.0000	*0.6826*	*0.5501*
G	224.39	298.94	216.86
	0.6826	*1.0000*	*0.6512*
B	201.41	216.86	370.92
	0.5501	*0.6512*	*1.0000*

(b) MLRI

	R	G	B
R	359.90	320.44	302.85
	1.0000	*0.8967*	*0.8461*
G	320.44	354.83	299.85
	0.8967	*1.0000*	*0.8437*
B	302.85	299.85	355.99
	0.8461	*0.8437*	*1.0000*

(c) RCNN

	R	G	B
R	334.84	297.31	275.28
	1.0000	*0.8675*	*0.8181*
G	297.31	350.81	270.32
	0.8675	*1.0000*	*0.7848*
B	275.28	270.32	338.17
	0.8181	*0.7848*	*1.0000*

(d) JCNN

Table 4. Comparison in CPSNR(dB) of average restoration performance between $DN\&DM$ and $DM\&DN$ for a different noise levels. and DM means demosaicking. We test two denoisers DN_1 denotes nlBayes denoising; DN_2 denotes CBM3D denoising and $1.5DN$ means that if noise level is σ_0, the noise level parameter for the denoising method DN_x is $\sigma = 1.5\sigma_0$. Both denoisers can be adapted to handle mosaics in the $DN\&DM$ schemes, using BM3D-CFA (Danielyan *et al.* 2009) for DN_1 and a 4-channel nlBayes (LeBrun *et al.* 2013). The best result of each column is marked with a $\boxed{\text{box}}$. The best result of each line is in red and the second best one is in green.

σ_0	Algorithm	HA	RI	MLRI	ARI	RCNN
5	$DN_1\&B$	33.13	33.92	34.09	**34.39**	**34.98**
	$DM\&DN_1$	33.76	34.36	34.38	**34.86**	**35.15**
	$DM\&1.5DN_1$	33.99	34.51	34.58	**34.90**	**35.34**
	$DN_2\&DM$	33.53	34.16	34.16	**34.67**	**34.82**
	$DM\&DN_2$	33.50	34.13	34.13	**34.63**	**34.79**
	$DM\&1.5A_2$	34.00	34.54	34.61	**34.93**	**35.39**
10	$DN_1\&B$	31.04	31.32	31.36	**31.46**	**31.73**
	$DM\&A_1$	31.82	**32.13**	31.97	**32.41**	32.05
	$DM\&1.5DN_1$	32.11	32.28	32.29	**32.44**	**32.67**
	$DN_2\&DM$	31.41	**31.74**	31.51	**32.06**	31.36
	$DM\&A_2$	31.44	**31.76**	31.52	**32.06**	31.38
	$DM\&1.5A_2$	32.17	32.38	32.36	**32.52**	**32.75**
20	$DN_1\&DM$	28.17	28.17	28.17	**28.18**	**28.28**
	$DM\&DN_1$	28.67	**28.99**	28.57	**29.21**	28.02
	$DM\&1.5DN_1$	29.29	29.26	29.22	**29.31**	**29.36**
	$DN_2\&DM$	28.11	**28.45**	27.97	**28.69**	27.27
	$DM\&DN_2$	28.15	**28.46**	27.95	**28.70**	27.28
	$DM\&1.5A_2$	29.24	29.32	29.22	**29.36**	**29.41**
40	$DN_1\&DM$	**24.67**	24.62	24.62	24.62	**24.64**
	$DM\&DN_1$	24.98	**25.33**	24.82	**25.47**	24.10
	$DM\&1.5DN_1$	**25.44**	25.27	25.19	**25.29**	25.24
	$DN_2\&DM$	23.94	**24.45**	23.81	**24.65**	22.83
	$DM\&DN_2$	24.00	**24.42**	23.79	**24.67**	23.05
	$DM\&1.5A_2$	**25.53**	**25.53**	25.39	25.51	25.52

Table 5. CPSNR results on the a sample of 14 images extracted from the SIDD dataset (Abdelhamed *et al.* 2018). The proposed $DM\&1.5DN$ scheme RCNN+CBM3D outperforms the $DN\&DM$ scheme BM3D+RCNN.

Image ID	σ_0 (scaled to 255)	noisy	RCNN+CBM3D	BM3D+RCNN	JCNN
0028_001_IP_00100_00160_5500_N	9.45	28.50	41.67	39.00	40.87
0039_002_IP_00100_00180_5500_L	5.70	30.90	40.91	39.22	39.26
0044_002_IP_00100_00180_5500_N	8.17	31.68	38.81	38.41	36.61
0091_004_IP_00320_00080_3200_L	9.22	28.82	38.48	37.03	38.26
0110_005_IP_00100_00100_5500_L	8.98	30.71	40.94	39.97	39.29
0116_005_IP_00800_01520_5500_N	8.36	28.12	39.23	37.91	37.55
0134_006_IP_00100_00100_5500_N	9.32	31.69	39.00	38.62	37.33
0136_006_IP_00800_00800_5500_N	9.58	27.61	39.16	38.06	37.69
0159_007_IP_00100_00100_3200_L	5.65	32.60	42.19	41.20	41.14
0161_007_IP_00800_00800_3200_L	5.65	28.38	41.12	38.73	40.96
0185_008_IP_00400_00400_3200_L	5.74	28.46	35.84	34.30	34.62
0188_008_IP_00100_00100_3200_N	5.74	33.09	35.59	35.77	34.54
0193_009_IP_00800_02000_3200_N	5.34	28.25	40.01	37.83	39.59
0197_009_IP_00100_00200_5500_L	10.43	30.35	42.00	38.58	41.86