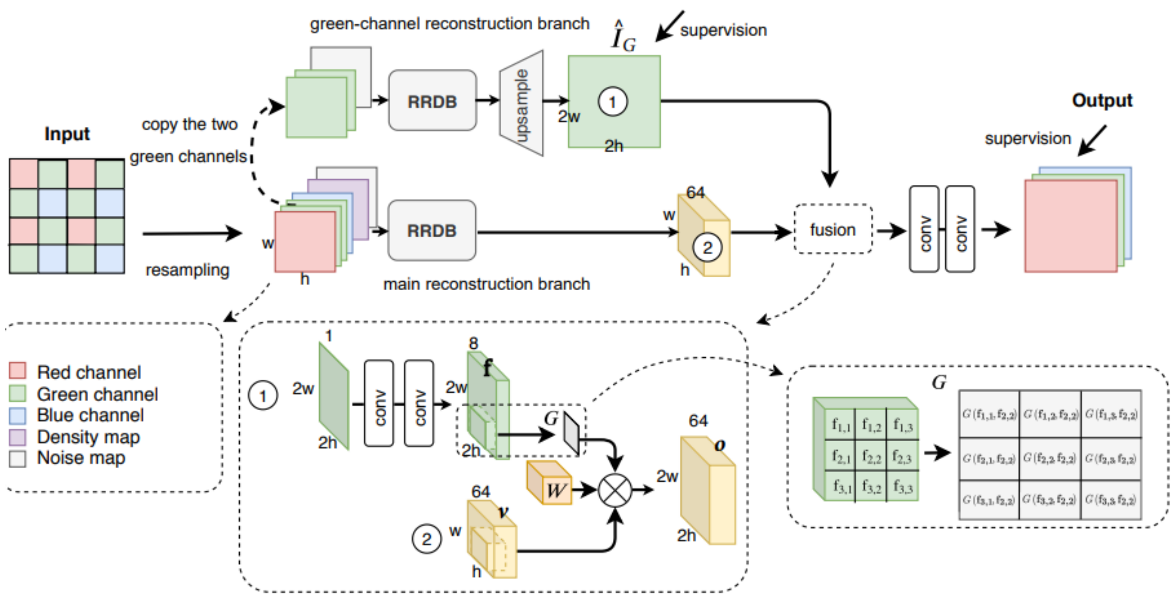
IML 세미나

2022-01-06

김태현

Joint Demosaicing and Denoising with Self Guidance

Lin Liu[1,2] Xu Jia[2*] Jianzhuang Liu[2] Qi Tian[2]
[1]CAS Key Laboratory of GIPAS, University of Science and Technology of China
[2]Noah's Ark Lab, Huawei Technologies

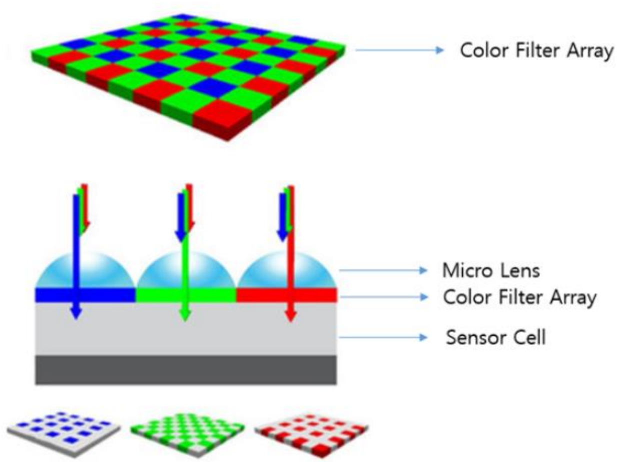


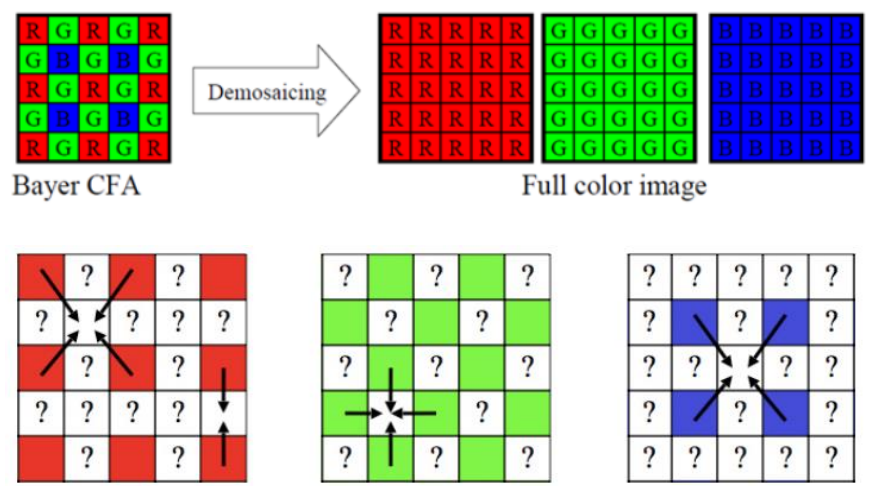
- Joint Demosaicing and Denoising
- Self Guidance

Novelty

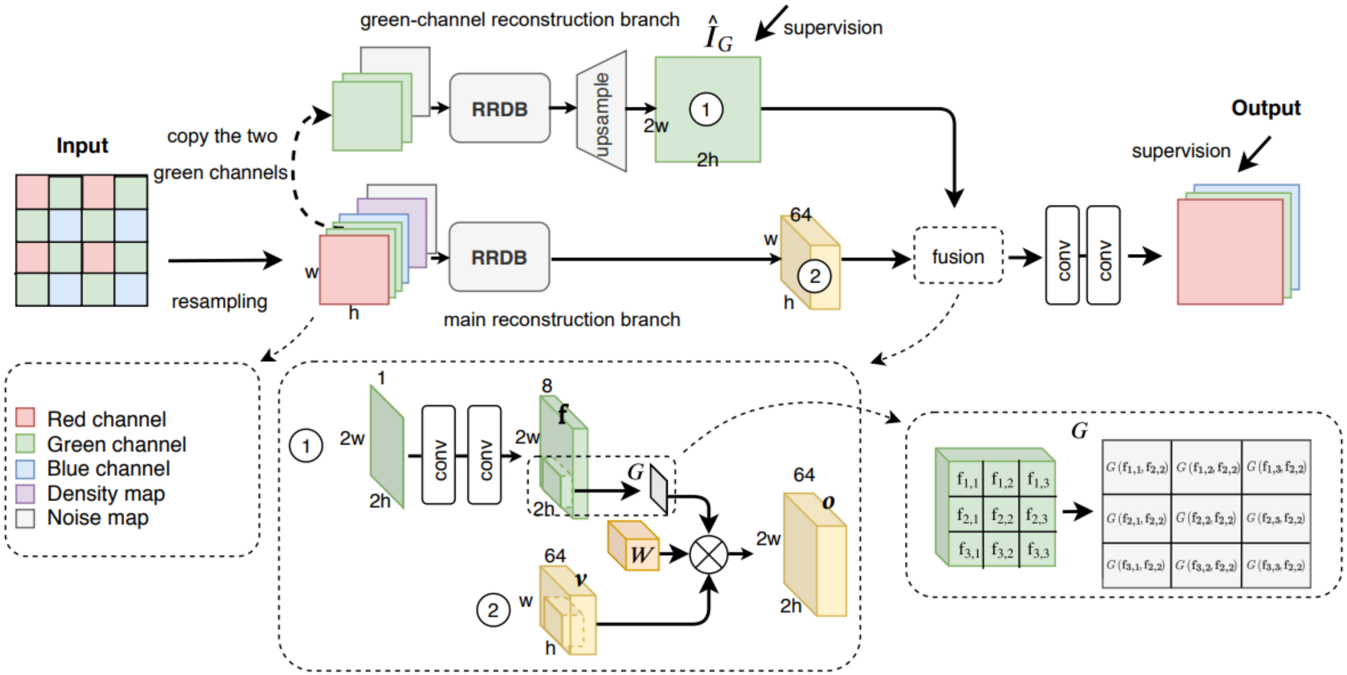
- Joint demosaicing and denoising을 위한 **density-map guidance** and **green-channel guidanc**가 있는 SGNet을 제안한다.
- **Density** 맵을 이용하여 영역 별로 잡음 제거, **Green Channel**을 이용하여 Demosaicing 도움.
- 텍스처를 복구하고 노이즈를 동시에 제거하기 위해 **adaptive-threshold edge loss** 와 **edge-aware smoothness loss**이라는 두 가지 loss를 제안한다.

센서 -> ADC -> DSP -> JPEG, H.264





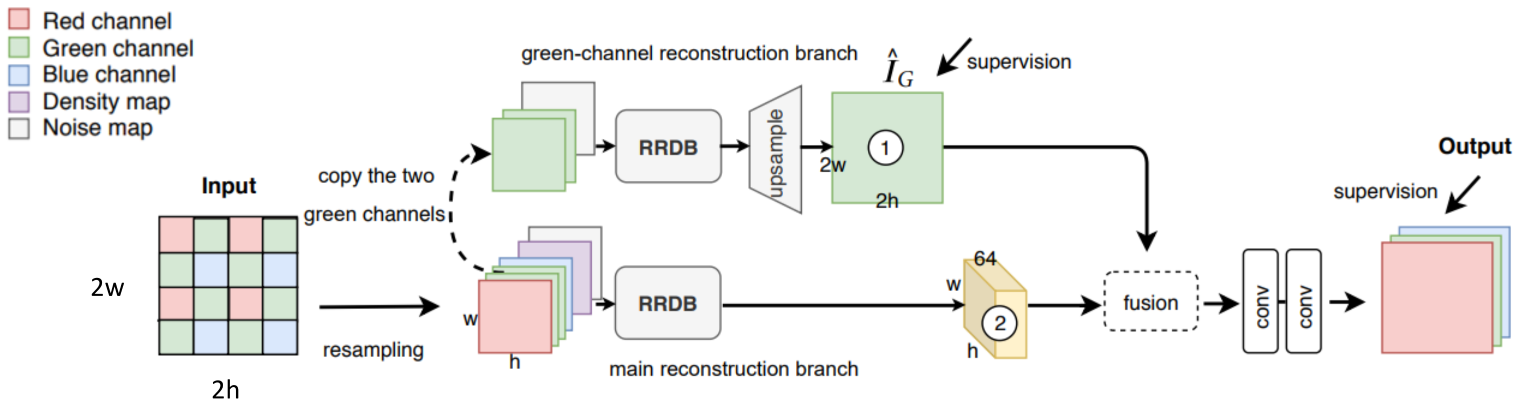
Proposed Method



1. Green-chaneel reconstruction branch
2. Main reconstruction branch
3. Loss : $L = L_{edge} + L_{smooth} + L_{L1} + L_G$

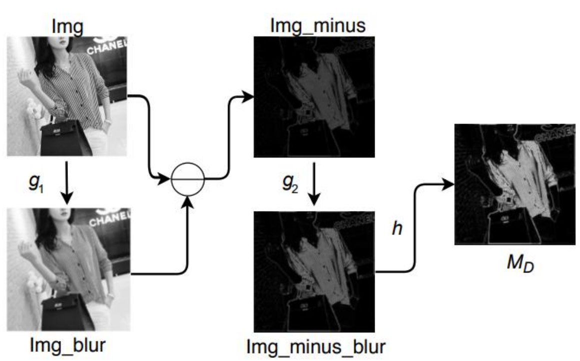
   Adaptive-threshold edge loss, Edge-aware smoothness loss

Proposed Method



1. Input RGGB raw 영상 (2Hx2Wx1) -> I_{G1}^{Bayer}, I_{G2}^{Bayer}, I_{R}^{Bayer}, I_{B}^{Bayer} (HxWx4) 분해
2. Density Map (M_D) 사용 -> 영역 별로 difficulty level이 다름을 알려주기 위함.

   Dense Texture : High frequency, Less Texture : Low frequency

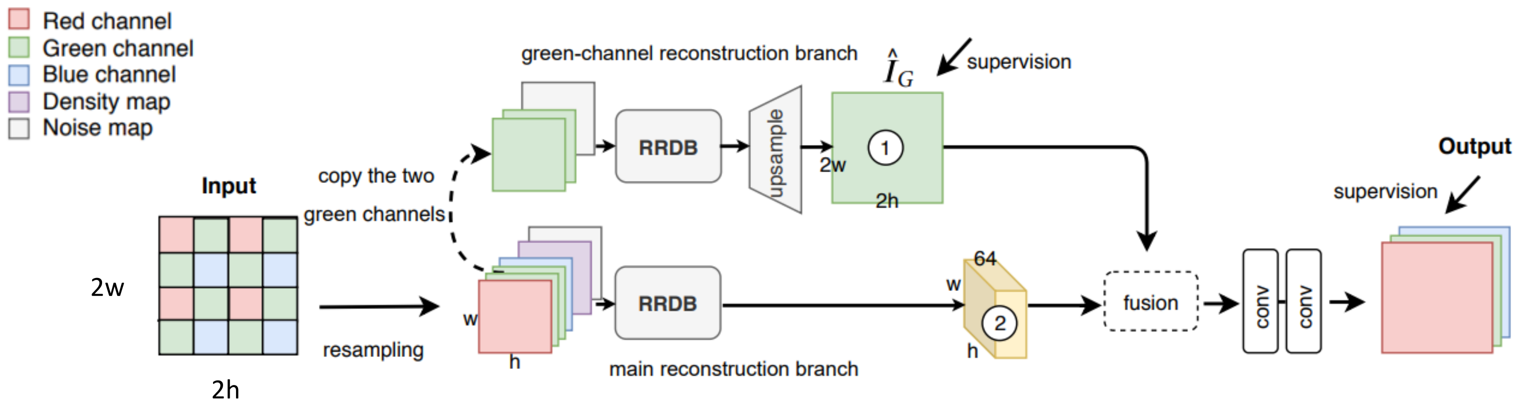


$$M_D = h(g_2(I_{gray} - g1(I_{gray}; K1); K2)$$

$$\mathrm{h(X)} = \frac{X - \min(X)}{\max(X) - \min(X) + \epsilon}$$

$$I_{gray} = \left(\left(I_{G1}^{Bayer} + I_{G1}^{Bayer}\right)/2 + I_{R}^{Bayer} + I_{B}^{Bayer}\right)/3$$

Proposed Method

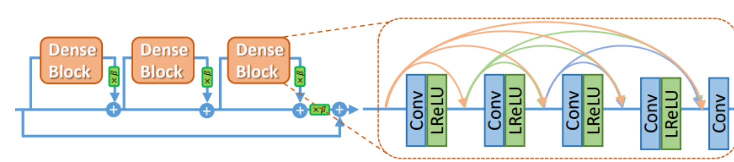


Input : $I_{G1}^{Bayer}, I_{G2}^{Bayer}, I_{R}^{Bayer}, I_{B}^{Bayer}, M_D, I^{noise}$

I^{noise} : noise map (Gausian noise level)

$\hat{I}_G$: 복원된 그린 채널 (2H x 2W x 1)

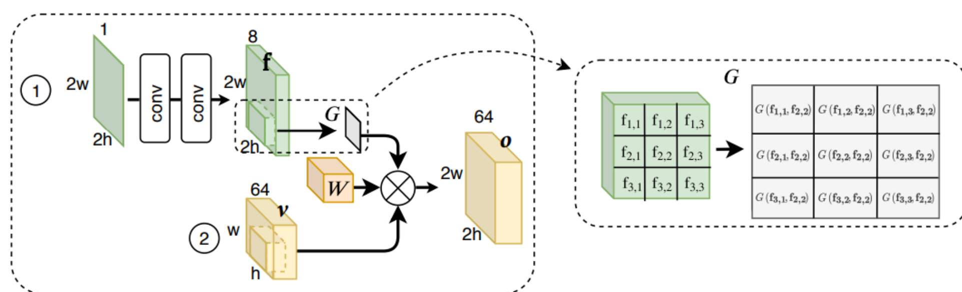
Residual in Residual Dense Block (RRDB)



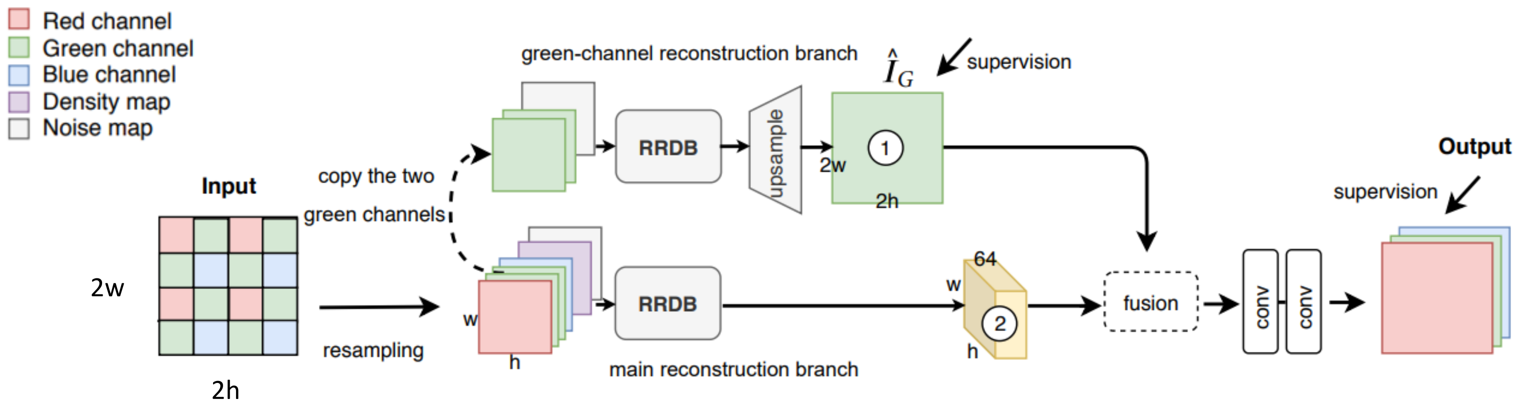
Fusion : **Pixel-Adaptive Convolution**

$$\mathbf{o}_i = \sum_{j \in \Omega(i)} G(\mathbf{f}_i, \mathbf{f}_j)\, \mathbf{W}_{p_{i,j}} \mathbf{v}_j + \mathbf{b}$$

$$G(\mathbf{f}_i, \mathbf{f}_j) = \exp\left(-\frac{1}{2}(\mathbf{f}_i - \mathbf{f}_j)^\top (\mathbf{f}_i - \mathbf{f}_j)\right)$$



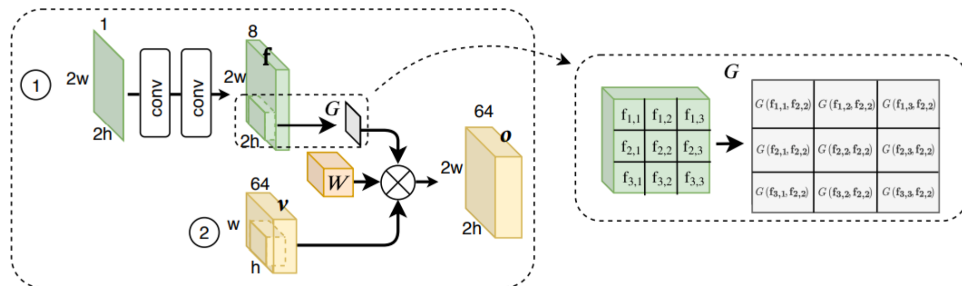
Proposed Method



Fusion : **Pixel-Adaptive Convolution**

$$\mathbf{o}_i = \sum_{j\in\Omega(i)} G\left(\mathbf{f}_i,\mathbf{f}_j\right)\mathbf{W}_{p_{i,j}}\mathbf{v}_j + \mathbf{b}$$

$$G\left(\mathbf{f}_i,\mathbf{f}_j\right) = \exp\left(-\frac{1}{2}\left(\mathbf{f}_i-\mathbf{f}_j\right)^{\top}\left(\mathbf{f}_i-\mathbf{f}_j\right)\right)$$

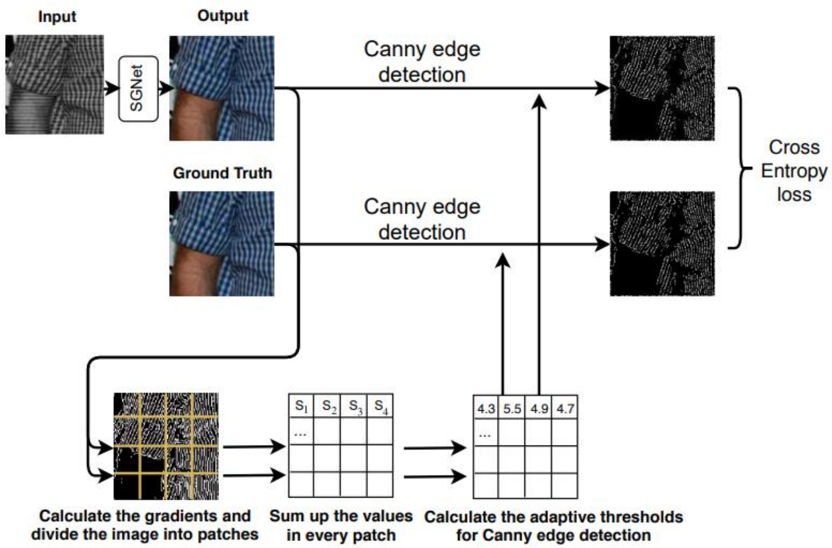


G : Gaussian function, 두 위치 간의 거리에 따라 다른 가중치

Ω_i : sxs convolution window, $\mathbf{v}_j \in \mathbb{R}^{c'}, j = 1,2,\ldots,H\times W$ $\mathbf{o}_i \in \mathbb{R}^{c}, i = 1,2,\ldots,2H\times 2W$

W : convolution weights, W $\in \mathbb{R}^{c'\times c\times s\times s}$

Proposed Method



$$\theta_i = \theta_0 + k \frac{s_{P_i}}{\max(s_{P_1}, s_{P_2}, \ldots, s_{P_n})}, \theta_0, k > 0,$$

$$L_{edge} = \sum_{i=1}^{n} -\beta * p(E(P_i^T;\theta_i)) * log(p(E(P_i^O;\theta_i))) - (1-\beta) * (1 - p(E(P_i^T;\theta_i))) * log(1 - p(E(P_i^O;\theta_i))),$$

$$\beta = |E(I^T)|/|I^T| \qquad \text{Loss} = -\frac{1}{\text{output size}} \sum_{i=1}^{\text{output size}} y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log(1 - \hat{y}_i)$$

- **Adaptive-threshold edge loss** : 고주파 디테일이 많은 지역에 가중치를 두기 위한 loss

P_i : 영상 패치들, S_{P_i} : 패치 픽셀 값들의 합, T : target, O : 네트워크 output

θ_i : Adaptive thresholds

$p(E(P_i\ ;\ \theta_i))$: Probability of being edges for a certain patch

$|E(I^T)|$: the number of edge pixels

- $L = L_{edge} + L_{smooth} + L_{L1} + L_G$

Proposed Method

- Edge-aware smoothness loss : 텍스처 영역에서 edge를 유지하면서 부드러운 영역에서 노이즈를 동시에 제거하는 방법을 학습함.

$$L_{smooth} = \|\nabla O \circ \exp(-\lambda \nabla O)\| \qquad \nabla O = \|\nabla O_h\| + \|\nabla O_v\|$$

∇O : 수평과 수직 그래디언트의 총합

λ : parameter balancing the strength of edge-awareness.

- L1 loss : GT와 네트워크의 출력 사이의 L1
- Green loss : $\hat{I}_G$ 와 GT의 그린 채널 사이의 L1
- $L = L_{edge} + L_{smooth} + L_{L1} + L_G$

$$\mathcal{L}(y - \hat{y}) = \sum_{i=1}^{n} |y - \hat{y}_i|$$

Experimental Results

Method	σ	Dense texture PSNR	Dense texture LPIPS	Sparse texture PSNR	Sparse texture LPIPS	MIT moire PSNR	MIT moire SSIM	MIT moire LPIPS	Urban100 PSNR	Urban100 SSIM	Urban100 LPIPS
FlexISP [11]	5	36.64	–	38.82	–	29.06	0.8206	–	30.37	0.8832	–
SEM [18]		38.55	–	38.40	–	27.46	0.8292	–	27.19	0.7813	–
ADMM [35]		39.97	–	43.78	–	28.58	0.7923	–	28.57	0.8578	–
Deepjoint [5]		45.90	0.01782	47.48	0.02218	31.82	0.9015	0.0692	34.04	0.9510	0.0304
Kokkinos [19]		44.77	0.02308	47.40	0.02158	31.94	0.8882	0.0711	34.07	0.9358	0.0437
CDM* [36]		45.49	0.02050	47.24	0.02281	30.36	0.8807	0.0757	32.09	0.9311	0.0484
SGNet		**46.75**	**0.01433**	**47.88**	**0.01917**	**32.15**	**0.9043**	**0.0691**	**34.54**	**0.9533**	**0.0299**
FlexISP [11]	10	32.66	–	33.62	–	26.61	0.7491	–	27.51	0.8196	–
SEM [18]		31.26	–	31.17	–	25.45	0.7531	–	25.36	0.7094	–
ADMM [35]		37.64	–	40.25	–	28.26	0.7720	–	27.48	0.8388	–
Deepjoint [5]		43.26	0.04017	45.17	0.04139	29.75	0.8561	0.1066	31.60	0.9152	0.0610
Kokkinos [19]		41.95	0.05499	44.86	0.04328	30.01	0.8123	0.1132	31.73	0.8912	0.0710
CDM* [36]		42.46	0.05393	44.70	0.04535	28.63	0.8286	0.1304	30.03	0.8934	0.0832
SGNet		**44.23**	**0.03280**	**45.56**	**0.03764**	**30.09**	**0.8619**	**0.1034**	**32.14**	**0.9229**	**0.0546**
FlexISP [11]	15	29.67	–	30.48	–	24.91	0.6851	–	25.55	0.7642	–
SEM [18]		25.98	–	27.01	–	23.23	0.6527	–	23.25	0.6156	–
ADMM [35]		34.87	–	36.78	–	27.58	0.7497	–	28.37	0.8440	–
Deepjoint [5]		41.42	0.06447	43.54	0.05783	28.22	0.8088	0.1506	29.73	0.8802	0.0929
Kokkinos [19]		40.16	0.08046	43.11	0.06103	28.28	0.7693	0.1764	29.87	0.8451	0.1054
CDM* [36]		40.51	0.08306	42.98	0.06549	27.23	0.7775	0.1875	28.34	0.8543	0.1262
SGNet		**42.32**	**0.05491**	**43.94**	**0.05456**	**28.60**	**0.8188**	**0.1412**	**30.37**	**0.8923**	**0.0793**

Table 1: Quantitative comparison for joint demosaicing and denoising on real test sets (Dense texture and Sparse texture) and synthetic test sets (MIT moire and Urban100). PSNR, SSIM: higher is better; LPIPS: lower is better.

σ	Dense texture 5	Dense texture 10	Dense texture 15	Sparse texture 5	Sparse texture 10	Sparse texture 15
Metrics	PSNR/LPIPS	PSNR/LPIPS	PSNR/LPIPS	PSNR/LPIPS	PSNR/LPIPS	PSNR/LPIPS
w/o green branch	45.43/0.02373	42.90/0.04381	41.21/0.06654	47.15/0.01995	44.30/0.03935	42.44/0.05953
Concatenation guidance	45.93/0.02240	43.31/0.04182	42.05/0.06006	47.46/0.01885	45.16/0.03846	43.75/0.05782
Adaptive guidance	**46.75/0.01433**	**44.23/0.03280**	**42.32/0.05491**	**47.88/0.01917**	**45.56/ 0.03764**	**43.94/0.05456**

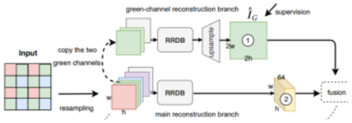


Table 2: Ablation study of the green-channel guidance method on Pixelshift200.

Experimental Results

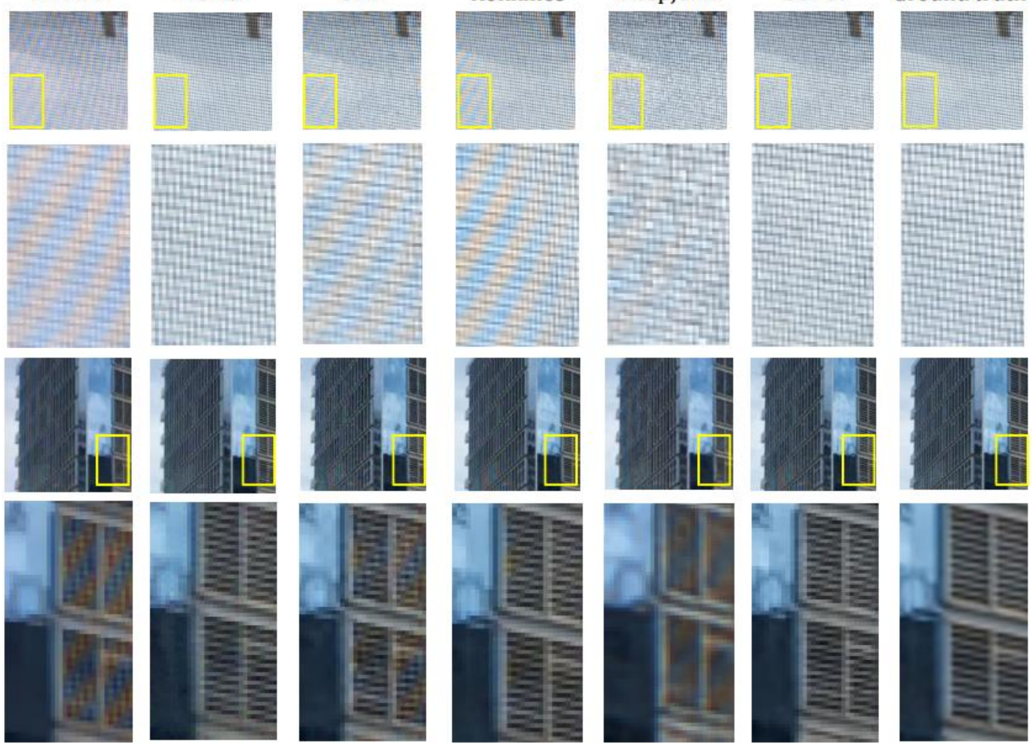


Figure 5: Visual comparison between state-of-the-arts and our method for joint demosaicing and denoising.

Method	σ	Dense texture PSNR/LPIPS	Sparse texture PSNR/LPIPS
w/o L_{smooth}		44.01/0.03574	45.40/0.03933
w/o L_{edge}	10	43.92/0.03752	45.42/0.03905
w/o M_D		43.86/0.04073	45.37/0.03982
Full model		**44.23/0.03280**	**45.56/0.03764**

Table 3: Ablation study on Pixelshift200.

KOREA UNIVERSITY