

# Pan-LUT: Efficient Pan-sharpening via Learnable Look-Up Tables

## Supplementary Material

Zhongnan Cai<sup>1\*</sup> Yingying Wang<sup>1\*</sup> Hui Zheng<sup>1</sup> Panwang Pan<sup>2</sup>  
 Zixu Lin<sup>1</sup> Ge Meng<sup>1</sup> Chenxin Li<sup>3</sup> Chunming He<sup>4</sup>  
 Jiaxin Xie<sup>1</sup> Yunlong Lin<sup>1†</sup> Junbin Lu<sup>5</sup> Yue Huang<sup>1</sup> Xinghao Ding<sup>1‡</sup>  
<sup>1</sup>Key Laboratory of Multimedia Trusted Perception and Efficient Computing,  
 Ministry of Education of China, Xiamen University, Xiamen, Fujian, China  
<sup>2</sup>ByteDance <sup>3</sup>The Chinese University of Hong Kong  
<sup>4</sup>Duke University <sup>5</sup>University of Washington

The following contents are provided in the materials:

- More Technical Details.
- More Experimental Details.
- Discussion.

### 1 More Technical Details.

To assist the understanding of the proposed LUT, we first briefly review the preliminary about 3DLUT and trilinear interpolation. Then we delve into the descriptions of PGLUT, SDLUT, and AOLUT in detail. Furthermore, we also provide more details about loss function used in Pan-LUT.

#### 1.1 Preliminary: 3D Lookup Tables

In this study, the 3D Look-Up Table (3D-LUT) is represented as a discrete sampling of a complete 3D color transformation function, encompassing a total of  $N^3$  sampling points in the space, where  $N$  denotes the number of bins in each dimension. Each sampling point defines a set of input color coordinates  $\{I_{(i,j,k)}^r, I_{(i,j,k)}^g, I_{(i,j,k)}^b\}$  and their corresponding output values  $\{O_{(i,j,k)}^r, O_{(i,j,k)}^g, O_{(i,j,k)}^b\}$ . Once sampling elements within the 3D-LUT, the input color looks up its nearest eight sampling points based on its index and calculate its transformed output by trilinear interpolation. As shown in the top part of Figure 1, which visually demonstrates the lookup process, to identify the eight nearest adjacent elements surrounding a

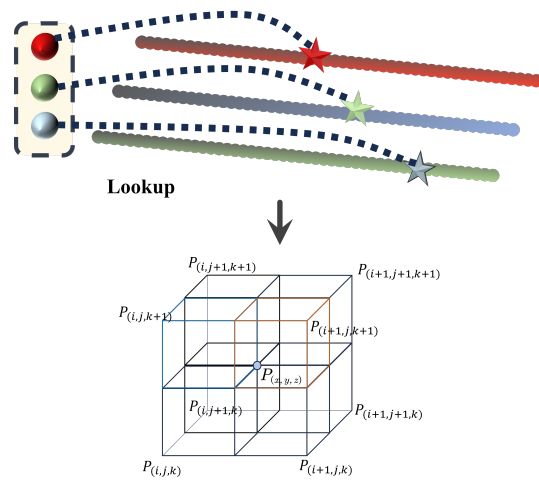


Figure 1: Procedures of the lookup and trilinear interpolation.

\*Equal Contribution.

†Project Leader.

‡Corresponding author.

specific input index in a 3D-LUT, we first calculate this index  $(i, j, k)$  using the input RGB values  $\{I_{(x,y,z)}^r, I_{(x,y,z)}^g, I_{(x,y,z)}^b\}$ . This lookup process unfolds as follows:

$$\begin{aligned} x &= \frac{I_{(x,y,z)}^r}{\Delta}, y = \frac{I_{(x,y,z)}^g}{\Delta}, z = \frac{I_{(x,y,z)}^b}{\Delta}, \\ i &= \lfloor x \rfloor, j = \lfloor y \rfloor, k = \lfloor z \rfloor, \end{aligned} \quad (1)$$

where  $\Delta = V_{max}/N$ ,  $V_{max}$  denotes the maximum color value.  $\lfloor \cdot \rfloor$  signifies the floor function. Then, the offset between the input precise index  $(x, y, z)$  and the corresponding computed sampling point  $(i, j, k)$  can be calculated as:

$$d_x = x - i, d_y = y - j, d_z = z - k. \quad (2)$$

As shown in the bottom part of Figure 1, we derive the interpolated output as follows:

$$\begin{aligned} O_{(x,y,z)}^c &= (1 - d_x)(1 - d_y)(1 - d_z)O_{(i,j,k)}^c \\ &\quad + d_x(1 - d_y)(1 - d_z)O_{(i+1,j,k)}^c \\ &\quad + (1 - d_x)d_y(1 - d_z)O_{(i,j+1,k)}^c \\ &\quad + (1 - d_x)(1 - d_y)d_zO_{(i,j,k+1)}^c \\ &\quad + d_xd_y(1 - d_z)O_{(i+1,j+1,k)}^c \\ &\quad + (1 - d_x)d_yd_zO_{(i,j+1,k+1)}^c \\ &\quad + d_x(1 - d_y)d_zO_{(i+1,j,k+1)}^c \\ &\quad + d_xd_yd_zO_{(i+1,j+1,k+1)}^c \end{aligned}, \quad (3)$$

where  $c$  is an element of the set  $r, g, b$ . Notably, this interpolation operation is differentiable, allowing for the update of LUTs during end-to-end training.

## 2 More Experimental Details

### 2.1 Datasets

We conduct experiments over the widely-used datasets including WorldView-II (WV2), GaoFen2 (GF2) and WorldView-III (WV3). Specifically, WorldView-II encompasses industrial areas and natural landscapes, GaoFen 2 encompasses mountains and rivers, while WorldView-III predominantly features urban roads and urban scenes. The PAN images are cropped into patches with the size of  $128 \times 128 \times 1$ , while the MS images are cropped into patches with the size of  $32 \times 32 \times 4$ . The detailed composition of each dataset is reported in Table 1.

Table 1: The details of each satellite image dataset.

Dataset	Training number	Testing number
<b>WorldView-III</b>	2150	200
<b>WorldView-II</b>	760	80
<b>GaoFen2</b>	2712	200

### 2.2 Additional Quantitative Comparisons.

Additionally, we conduct an assessment on the full-resolution GF2 dataset. As shown in Table 2, the proposed method outperforms several deep learning-based approaches, such as PSCINN and PanFlow, across all performance metrics. This demonstrates its robust generalization capability in real-world scenarios.

**Evaluation on full-resolution scene.** As shown in Figure 2, 3, 4, 5, we provide additional visual comparison on the full-resolution WorldView-II satellite dataset with both traditional and deep learning methods. Specifically, the resolution of the PAN images in this dataset is  $1024 \times 1024$  and the resolution of the MS images in this dataset is  $256 \times 256$ .

Table 2: Evaluation on the real-world full-resolution scenes from GaoFen2 dataset. The best values are highlighted by red. The up or down arrow indicates higher or lower metric corresponding to better results.

Metrics	Brovey	IHS	PNN	MSDCNN	GPPNN	SFDI	UCGAN	PanFlow	PSCINN	Pan-Mamba	TA-DiffHQ	Ours
$D_\lambda \downarrow$	0.1378	0.0770	0.0746	0.0734	0.0782	0.0681	0.0820	0.0861	0.0729	0.0666	<b>0.0503</b>	0.0762
$D_S \downarrow$	0.2605	0.2985	0.1164	0.1151	0.1253	0.1119	0.1320	0.1684	0.1175	0.1172	<b>0.1010</b>	0.1233
QNR $\uparrow$	0.6390	0.6485	0.8191	0.8251	0.8073	0.8466	0.7982	0.7607	0.8196	0.8341	<b>0.8537</b>	0.8111

### 3 Discussion

#### 3.1 Discussion on Memory Requirements.

The LUT-based methods [1] [3] face a dilemma: increasing the number of LUTs to enhance performance leads to rapidly growing storage requirements, which are ultimately limited by the available on-device memory. This constraint hinders the deployment of such models on edge devices. For example, the space complexity of 3DLUT-based methods [2] [3] is  $O(ND^3)$ , where  $N$  is the number of LUTs and  $D$  is the table length. In contrast, the default sizes of our PGLUT, SDLUT and AOLUT are 9, 9 and 9, respectively. The parameters of a single PGLUT, SDLUT and AOLUT are 295K ( $5 \times 9^5$ ), 26K ( $4 \times 9^4$ ) and 236K ( $4 \times 9^5$ ), respectively. This observation suggests that the effectiveness of our proposed method is not dependent on consuming extensive storage resources to increase the LUT size. We provide the parameter count for each LUT of different sizes in Tabel ??, which can be calculated as follows:

$$\begin{aligned}
 Param_{PGLUT} &= 5N^5, \\
 Param_{SDLUT} &= 4N^4, \\
 Param_{AOLUT} &= 4N^5.
 \end{aligned} \tag{4}$$

#### 3.2 Discussion on Efficiency Comparison.

To process high-resolution images, DNN-based methods often resort to dividing the image into patches, performing inference on each patch, and stitching the results. However, to ensure fairness in experimental comparisons and avoid artifacts introduced by patch-based processing, we employ single-pass inference on the full-resolution images.

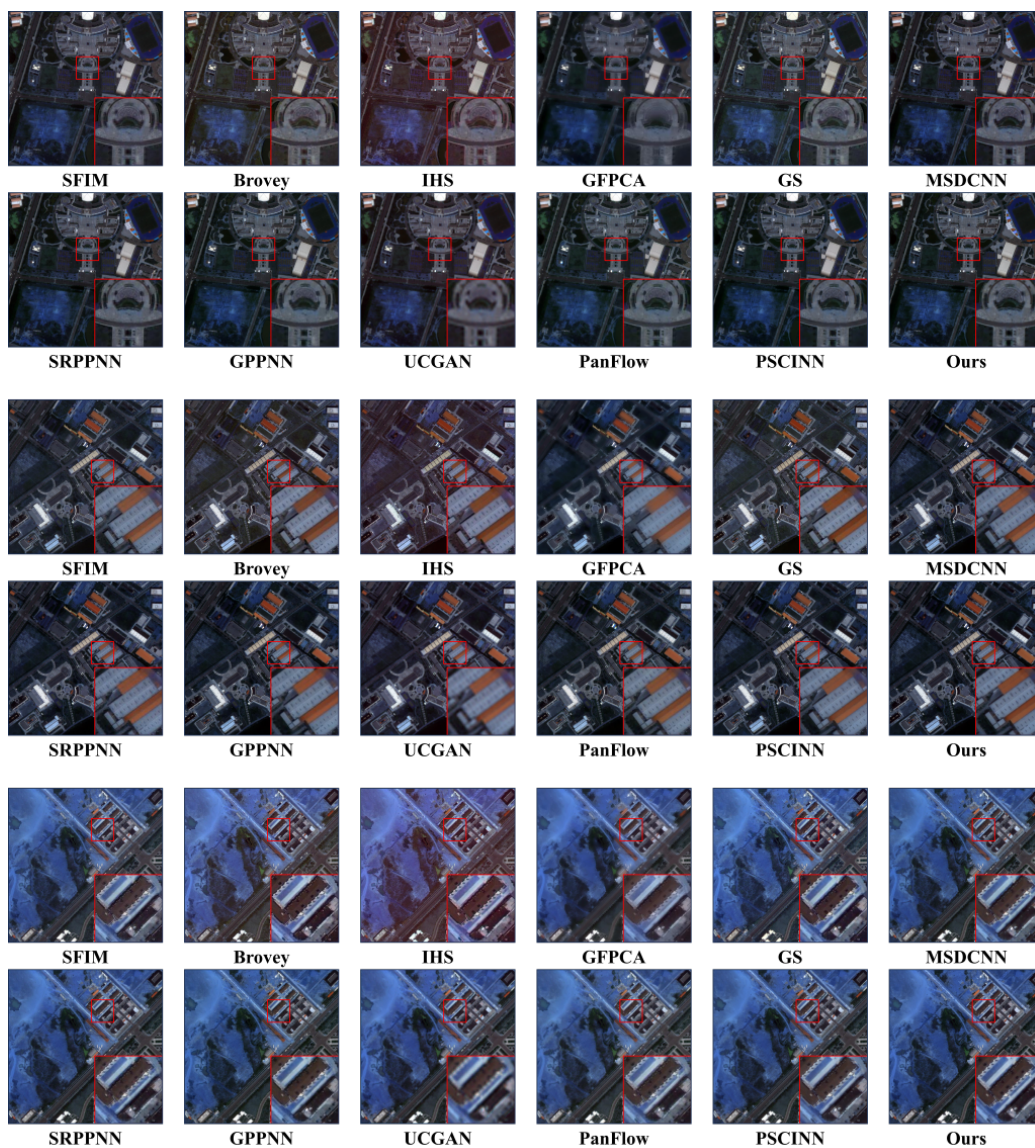


Figure 2: Visual comparison on the real full-resolution scenes from the WorldView-II dataset. For a more detailed examination of the results, we zoomed-in view on specific parts of the images.



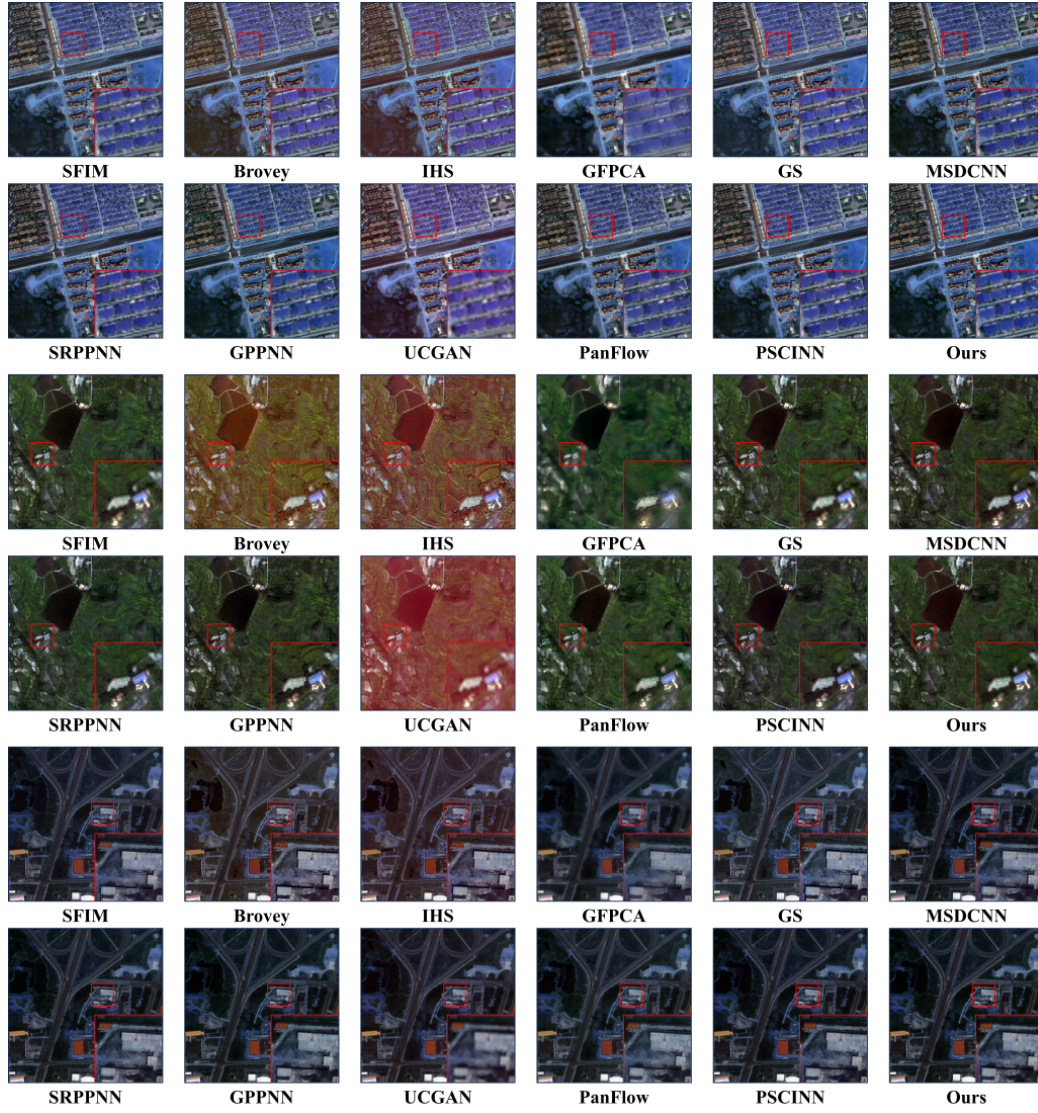


Figure 3: Visual comparison on the real full-resolution scenes from the WorldView-II dataset. For a more detailed examination of the results, we zoomed-in view on specific parts of the images.

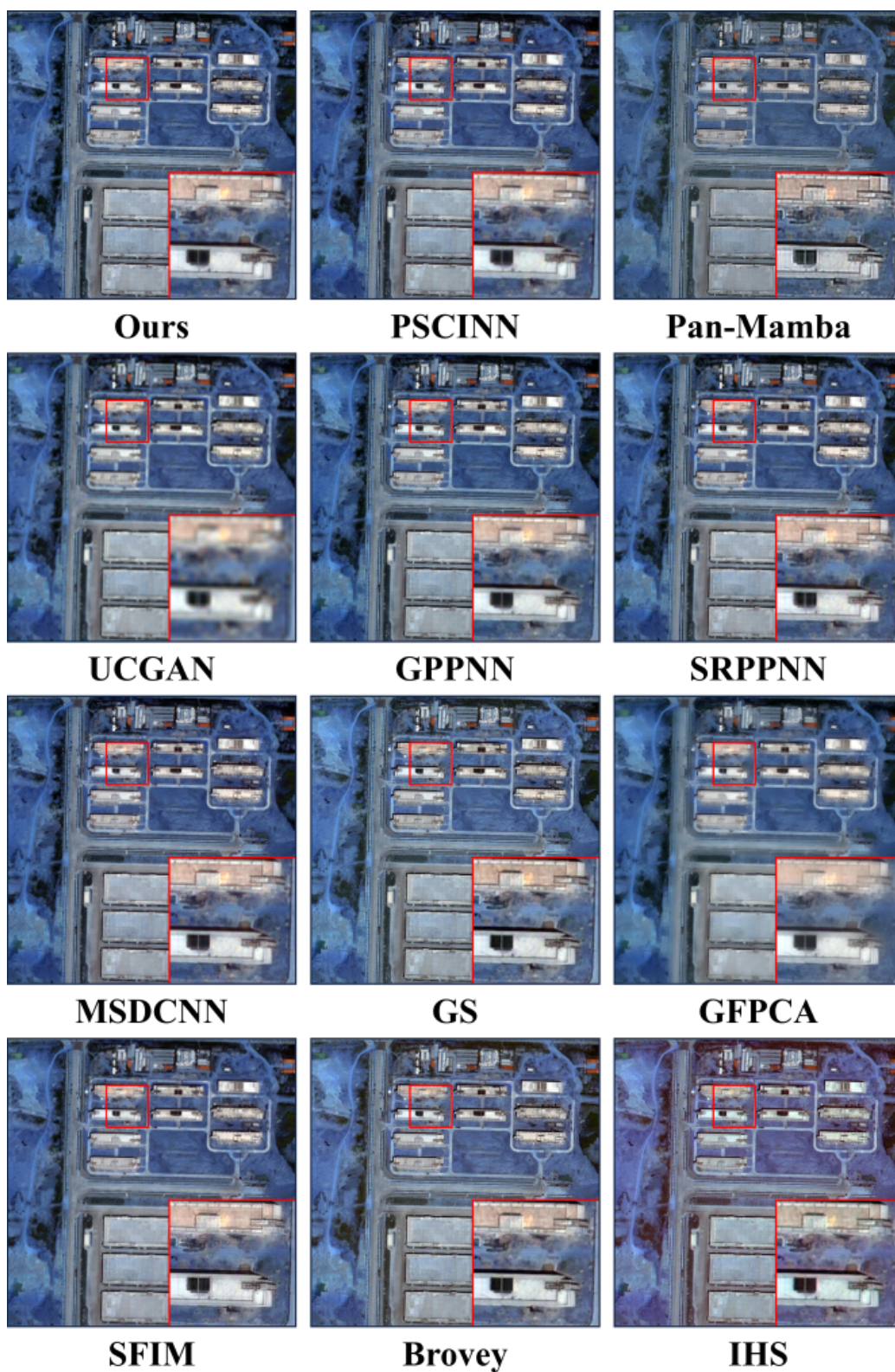


Figure 4: Visual comparison on the real full-resolution scenes from the WorldView-II dataset. For a more detailed examination of the results, we zoomed-in view on specific parts of the images.



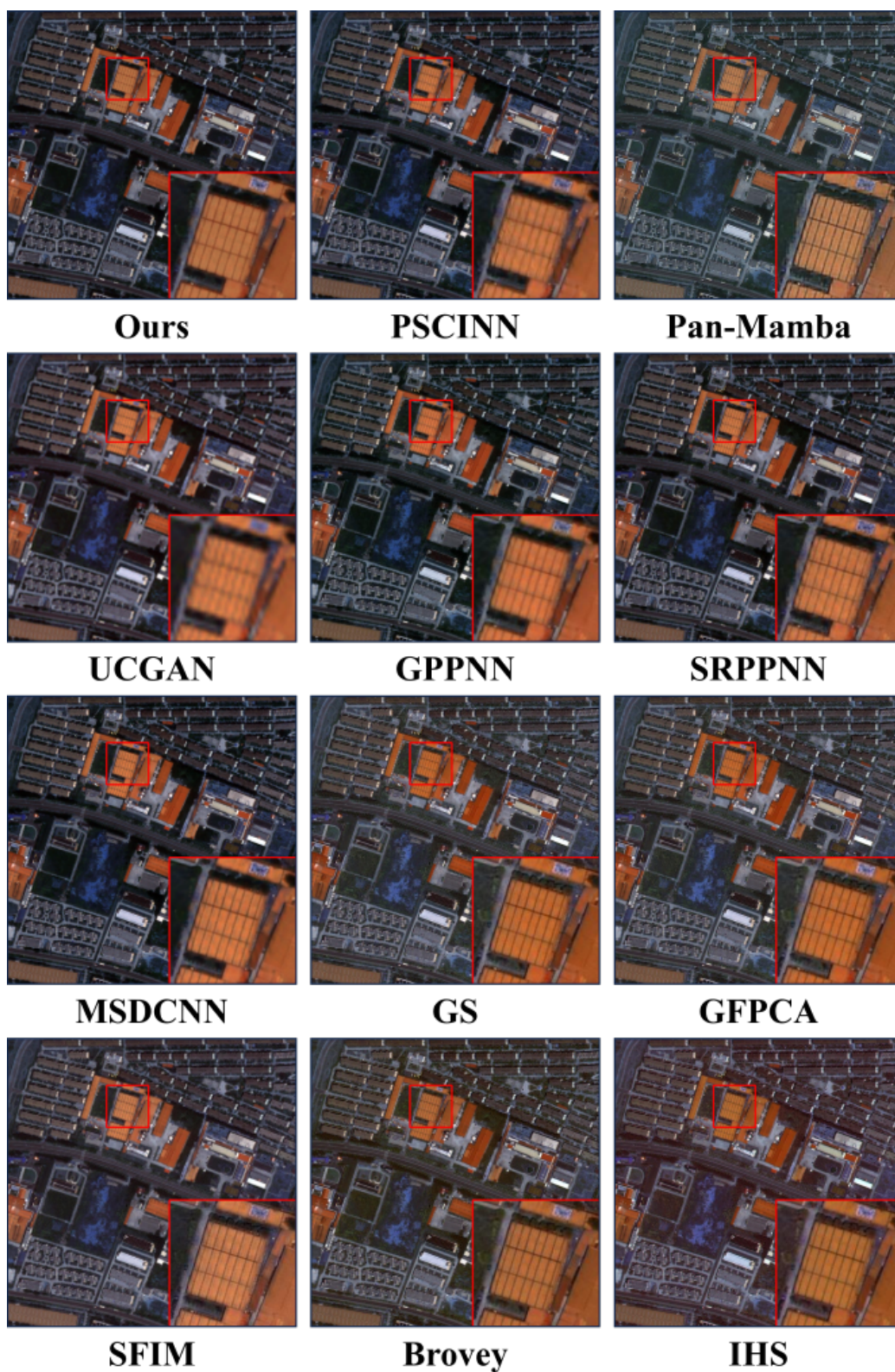


Figure 5: Visual comparison on the real full-resolution scenes from the WorldView-II dataset. For a more detailed examination of the results, we zoomed-in view on specific parts of the images.

## References

- [1] Y. Jo and S. Joo Kim. Practical single-image super-resolution using look-up table. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2021.
- [2] T. Wang, Y. Li, J. Peng, Y. Ma, X. Wang, F. Song, and Y. Yan. Real-time image enhancer via learnable spatial-aware 3d lookup tables. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2471–2480, 2021.
- [3] H. Zeng, J. Cai, L. Li, Z. Cao, and L. Zhang. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1–1, Jan 2020.