Supplementary Material for "Residual Degradation Learning Unfolding Framework with Mixing Priors across Spectral and Spatial for Compressive Spectral Imaging"

In this supplementary material, we first show the derivation of the formula for the proximal gradient descent (PGD) algorithm. Then, we provide the RGB images of the test scene and the visualization results of the spatial interaction. Furthermore, we offer a detailed description of the stage interaction and block interaction. Finally, we demonstrate more visual comparison results on both simulation data and real data.

The benchmark methods in our comparison include three model-based hyperspectral image (HSI) reconstruction methods (i.e., TwIST [1], GAP-TV [15] and DeSCI [10]) and four deep learning based methods (i.e., DGSMP [7], HDNet [6], MST [3] and CST [9]). The peak-signal-to-noise (PSNR) and the structural similarity index (SSIM) [14] are employed to evaluate the performance of competing HSI reconstruction methods.

1. The formulation of proximal gradient descent

Consider the unconstrained minimization problem of a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$:

$$\min\{f(x) : x \in \mathbb{R}^n\}. \tag{1}$$

One of the simplest methods for solving eq. (1) is the gradient algorithm which generates a sequence $\{x_k\}$ via:

$$x_0 \in \mathbb{R}^n, x_k = x_{k-1} - \rho \nabla f(x_{k-1}), \tag{2}$$

where ∇ is the differential operator, weighted by the step size $\rho > 0$. It is very well known that the gradient iteration (eq. (2)) can be viewed as a proximal regularization of the linearized function f at x_{k-1}, and written equivalently as

$$\begin{aligned} x_k = \underset{x}{\operatorname{argmin}}\, & f(x_{k-1}) + \langle x - x_{k-1}, \nabla f(x_{k-1}) \rangle \\ & + \frac{1}{2\rho}\|x - x_{k-1}\|^2. \end{aligned} \tag{3}$$

Adopting this same basic gradient idea to the regularized problem:

$$\min\{f(x) + \lambda J(x) : x \in \mathbb{R}^n\}, \tag{4}$$

leads to the iterative scheme:

$$\begin{aligned} x_k = \underset{x}{\operatorname{argmin}}\, & f(x_{k-1}) + \langle x - x_{k-1}, , \nabla f(x_{k-1}) \rangle \\ & + \frac{1}{2\rho}\|x - x_{k-1}\|^2 + \lambda J(x). \end{aligned} \tag{5}$$

After ignoring constant terms, this can be rewritten as

$$x_k = \underset{x}{\operatorname{argmin}} \frac{1}{2\rho}\|x - (x_{k-1} - \rho \nabla f(x_{k-1}))\|^2 + \lambda J(x).$$

Mathematically, the red part of the above function is a gradient descent operation and the blue part can be solved by the proximal operator $\text{prox}_{\lambda,J}$.

2. RGB images of the testing scenes and the visualization of spatial interaction

Fig. 1 shows the RGB images of the 10 scenes and the visualization of its corresponding spatial interaction. From Fig. 1, we can see that the feature map of the spatial interaction highlights the image foreground. Aided by the spatial interaction, the calculation of $query$(Q), key(K) and $value$(V) in spectral self-attention branch pays more attention to the informative regions and suppresses the uninformative areas.

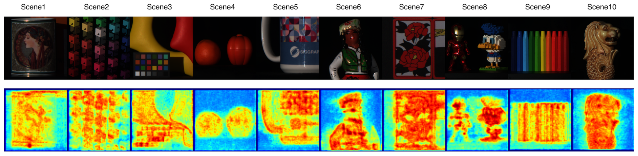

Figure 1. The RGB images of the 10 scenes and the visualization of the spatial interaction of the bi-directional interaction.

.

3. Stage Interaction and Block Interaction

Stage Interaction. The proposed stage interaction module normalizes current stage features in a **SP**atial **A**daptive **N**ormalization(SPAN) manner [11, 12]. SPAN uses previous stage features to generate modulation parameters. The calculation of modulation parameters is formulated as eq.(6a, 6b):

$$\psi_n^{k-1} = DConv(\sigma(Conv(F_n^{k-1}) + Conv(F_{N-n}^{k-1}))), \tag{6a}$$

$$\gamma_n^{k-1} = DConv(\sigma(Conv(F_n^{k-1}) + Conv(F_{N-n}^{k-1}))), \tag{6b}$$

where σ represents the activation function, F_n^{k-1} and $\hat{F}_n^k$ denote previous stage features and current stage features before normalization of n_{th} block, respectively. $\psi_n^{k-1}, \gamma_n^{k-1}$ represent the generated modulation parameters of current stage of n_{th} block respectively. k is the number of stage and N is the block number of MixS^2 Transformer. Then the current stage features are modulated as eq.(7):

$$F_n^k = \psi_n^{k-1} \odot \hat{F}_n^k + \gamma_n^{k-1}. \tag{7}$$

The proposed stage interaction module has several merits. First, information loss makes the network more vulnerable due to repeated use of up- and down-sampling operations in the encoder-decoder. Second, the multi-scale features of one stage help enrich the features of the next stage. Third, the network optimization procedure becomes more stable as it eases the flow of information.

Block Interaction. The encoder-decoder networks [2, 4, 8, 13] first gradually map the input to low-resolution representations, and then progressively apply reverse mapping to recover the original resolution. While these models effectively encode multi-scale information, they are prone to sacrificing spatial details due to the repeated use of downsampling operations. To address this issue, inspired by [5], we introduce the block interaction to allow information flow from different scales within a single MixS^2 Transformer. Each block interaction takes the output of all encoder blocks as an input and combines multiscale features using convolutional layers. The output of the block interaction is delivered to its corresponding decoder block.

4. More visual comparison results on simulation data

Fig.2-11 show more visual comparison results of the best five competing methods with 28 spectral channels for 10 testing scenes. Ground truth, measurements, and RGB images are shown for reference. We compare our RDLUF-MixS^2 9stage with CST-L-Plus [9], MST-L [3], HDNet [6] and DGSMP [7]. Thanks to the spectral self-attention branch that implicitly models long-range dependencies, and the multiscale convolution branch that improves texture and detail modeling capabilities, and the spectral-spatial interactions across them, the proposed method yields more detailed content, cleaner textures, and fewer artifacts.

5. More visual comparison results on real data

Fig.12-16 show more visual comparison results with 28 spectral channels for the 5 real scenes. We compare the our RDLUF-MixS^2 3stage with MST [3], DeSCI [10], GAP-TV [15] and TwIST [1]. From Fig.12-16, we can see that using the ability to model long-range dependencies of the spectral self-attention branch, the proposed method reconstructs the visually pleasant image. By the help of the multiscale convolution branch and the bi-directional interaction, the proposed method yields more detailed content, cleaner textures.

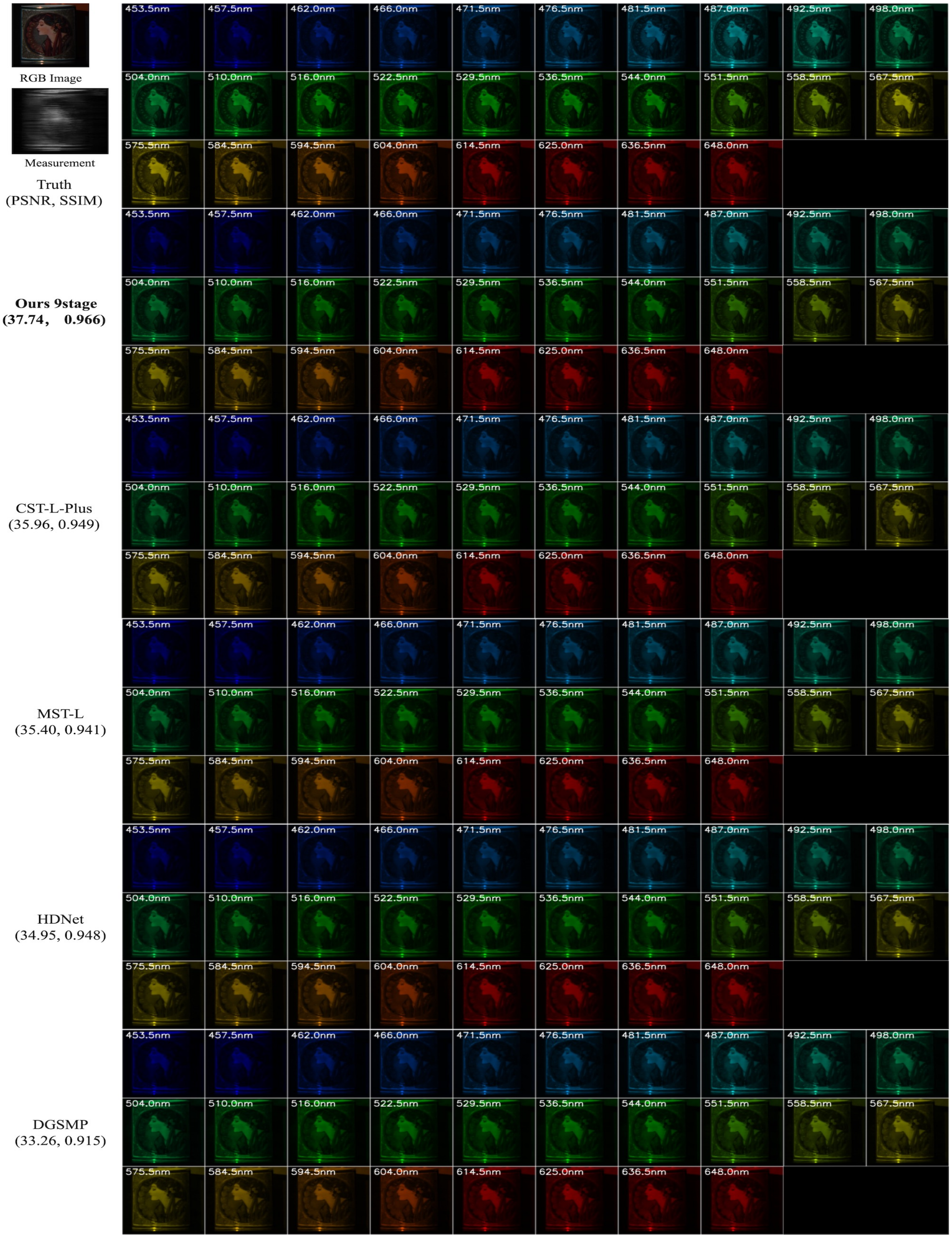

Figure 2. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 37.74dB, SSIM = 0.966), CST-L-Plus [9] (PSNR = 35.96dB, SSIM = 0.949), MST-L [3] (PSNR = 35.40dB, SSIM = 0.941), HDNet [6] (PSNR = 34.95dB, SSIM = 0.948) and DGSMP [7] (PSNR = 33.26dB, SSIM = 0.915) for *Scene1*. Zoom in for better view.

.

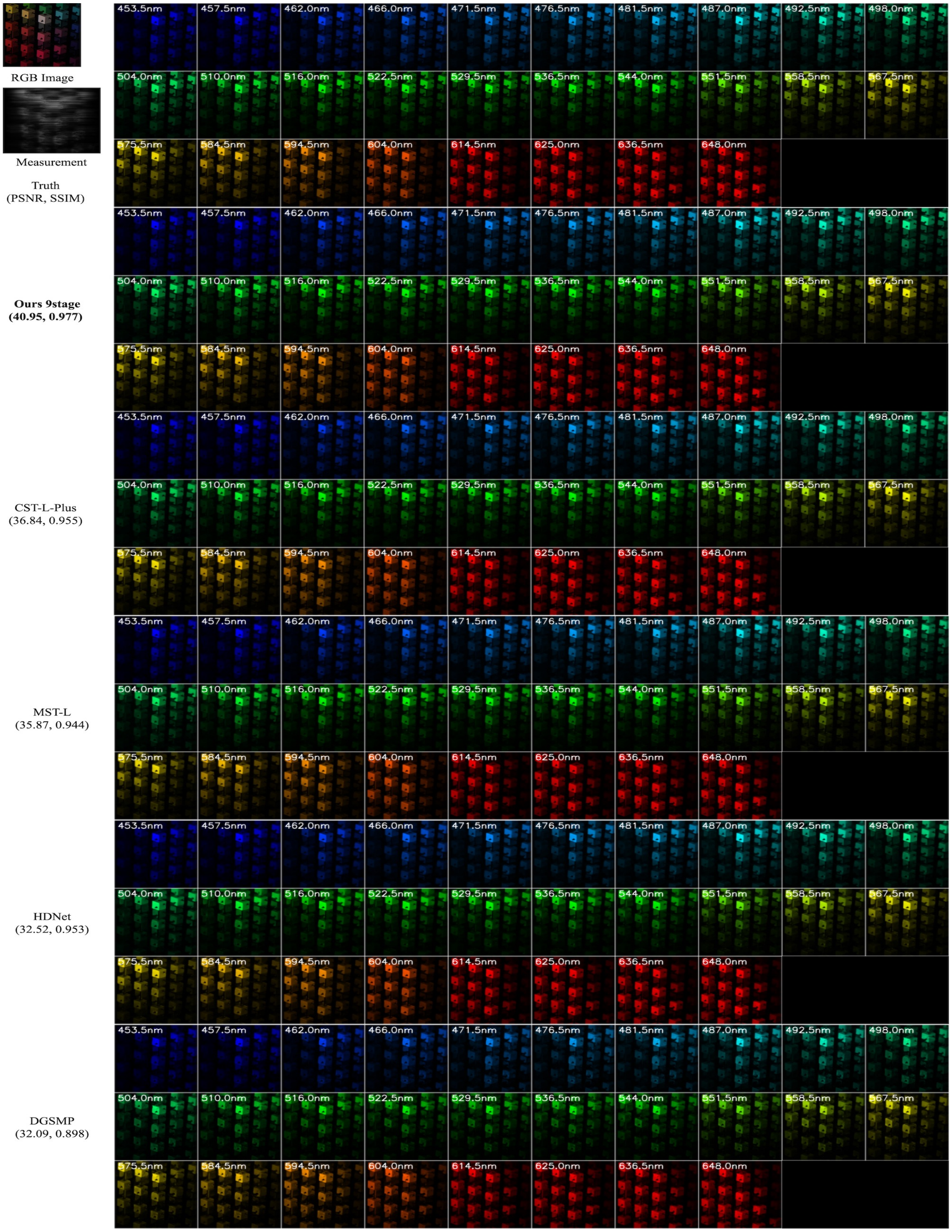

Figure 3. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 40.95dB, SSIM = 0.977), CST-L-Plus [9] (PSNR = 36.84dB, SSIM = 0.955), MST-L [3] (PSNR = 35.87dB, SSIM = 0.944), HDNet [6] (PSNR = 32.52dB, SSIM = 0.953) and DGSMP [7] (PSNR = 32.09dB, SSIM = 0.898) for *Scene2*. Zoom in for better view.

.

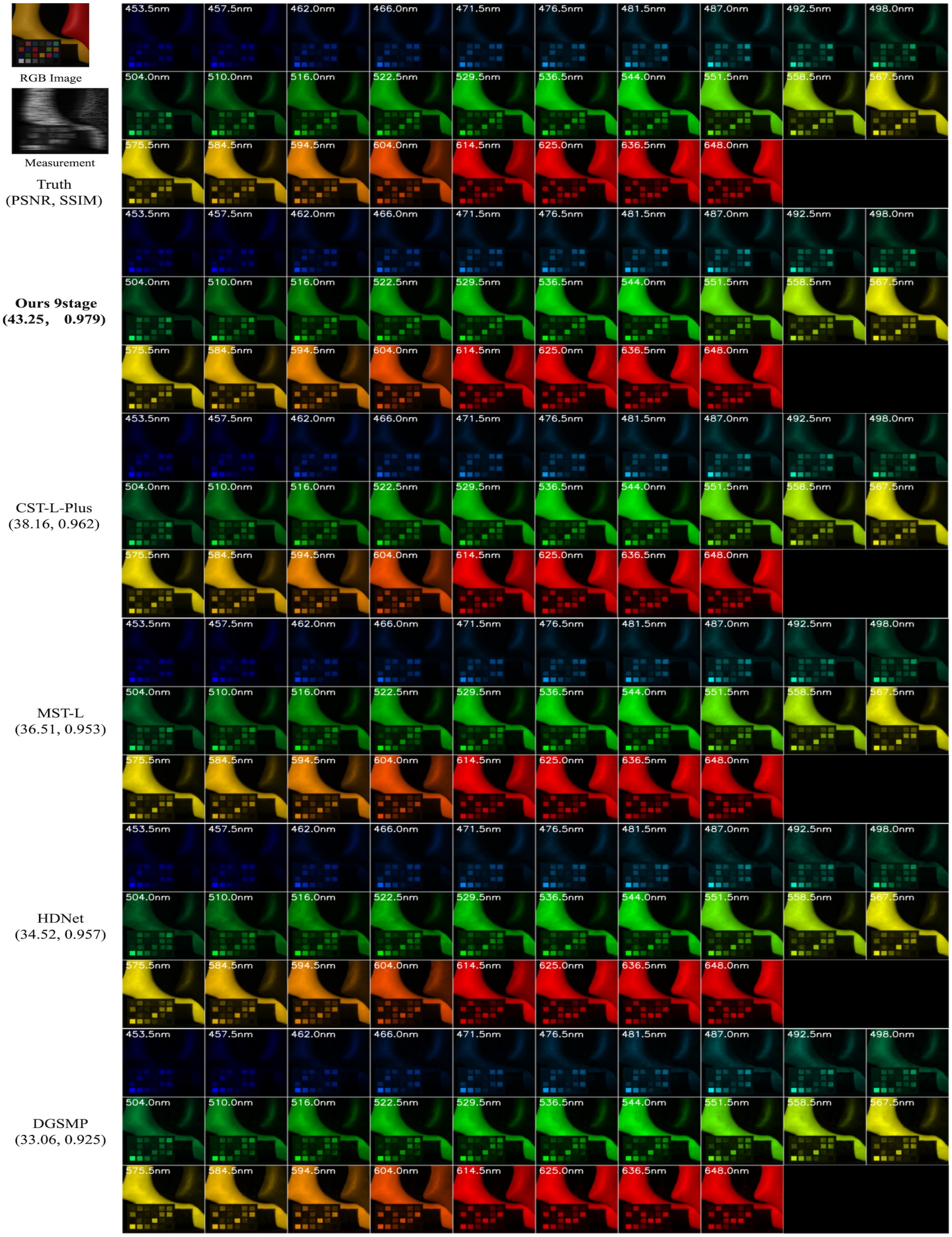

Figure 4. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 43.25dB, SSIM = 0.979), CST-L-Plus [9] (PSNR = 38.16dB, SSIM = 0.962), MST-L [3] (PSNR = 36.51dB, SSIM = 0.953), HDNet [6] (PSNR = 34.52dB, SSIM = 0.957) and DGSMP [7] (PSNR = 33.06dB, SSIM = 0.925) for *Scene3*. Zoom in for better view.

.

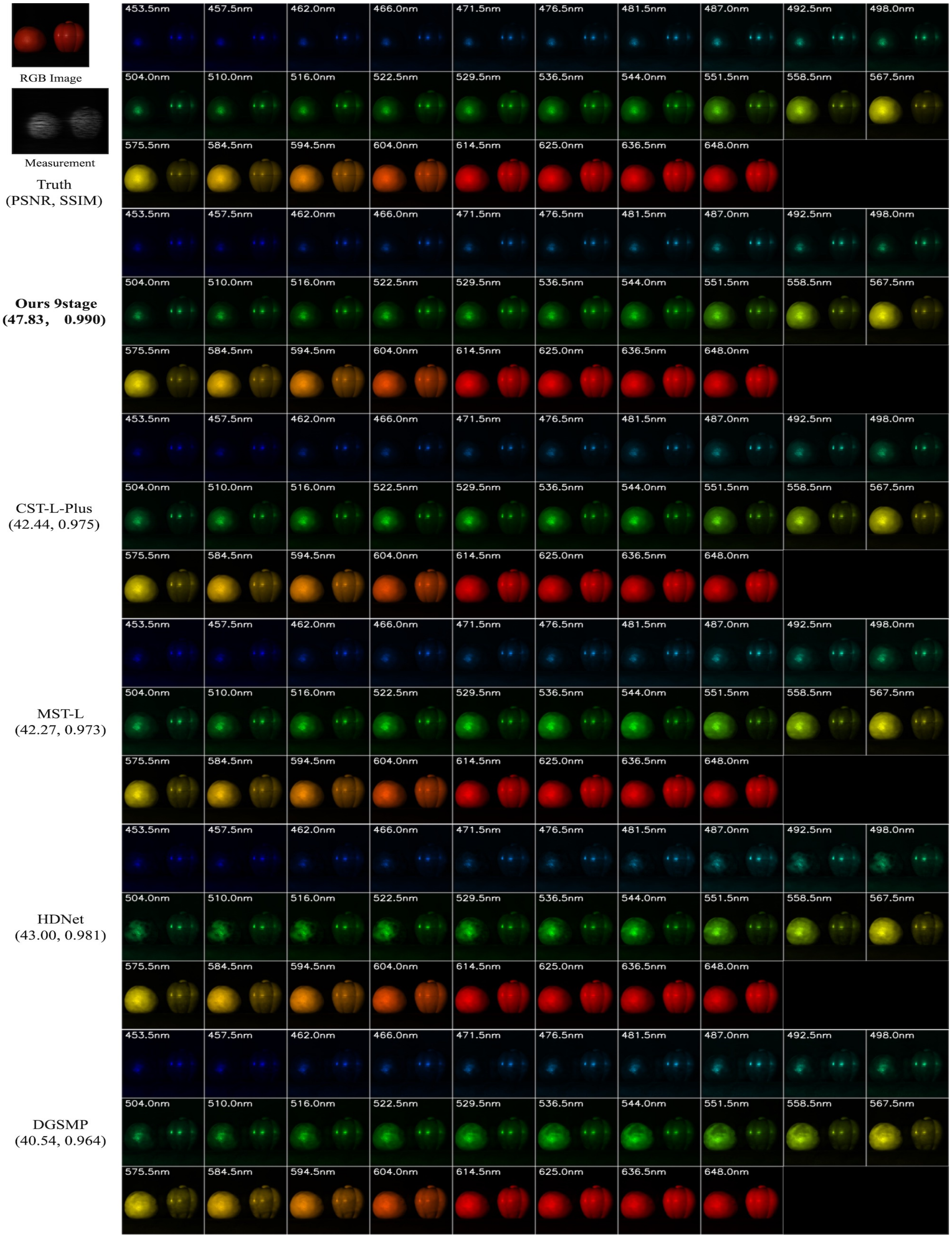

Figure 5. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 47.83dB, SSIM = 0.990), CST-L-Plus [9] (PSNR = 42.44dB, SSIM = 0.975), MST-L [3] (PSNR = 42.27dB, SSIM = 0.973), HDNet [6] (PSNR = 43.00dB, SSIM = 0.981) and DGSMP [7] (PSNR = 40.54dB, SSIM = 0.964) for *Scene4*. Zoom in for better view.

.

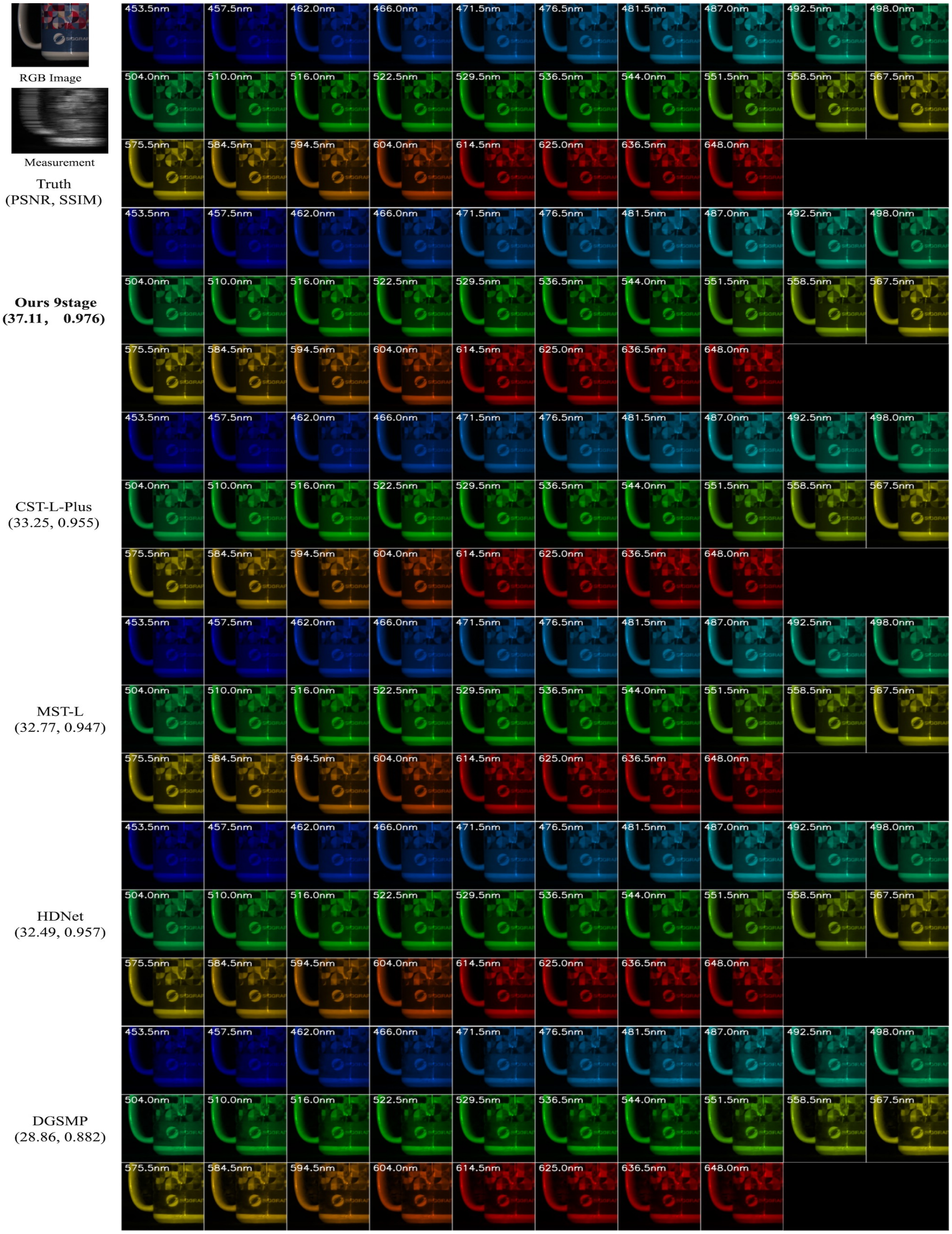

Figure 6. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 37.11dB, SSIM = 0.976), CST-L-Plus [9] (PSNR = 33.25dB, SSIM = 0.955), MST-L [3] (PSNR = 32.77dB, SSIM = 0.947), HDNet [6] (PSNR = 32.49dB, SSIM = 0.957) and DGSMP [7] (PSNR = 28.86dB, SSIM = 0.882) for *Scene5*. Zoom in for better view.

.

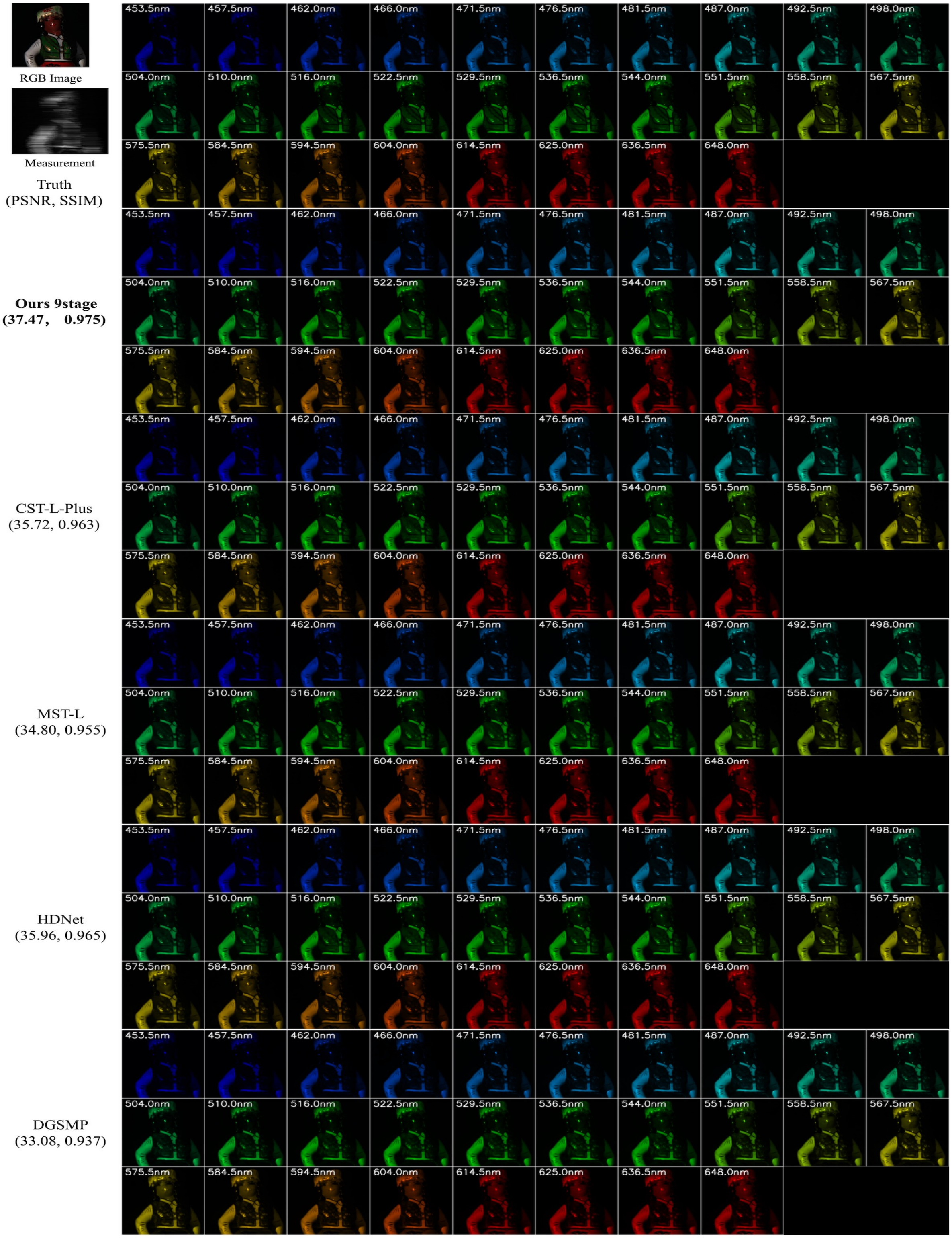

Figure 7. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 37.47dB, SSIM = 0.975), CST-L-Plus [9] (PSNR = 35.72dB, SSIM = 0.963), MST-L [3] (PSNR = 34.80dB, SSIM = 0.955), HDNet [6] (PSNR = 35.96dB, SSIM = 0.965) and DGSMP [7] (PSNR = 33.08dB, SSIM = 0.937) for *Scene6*. Zoom in for better view.

.

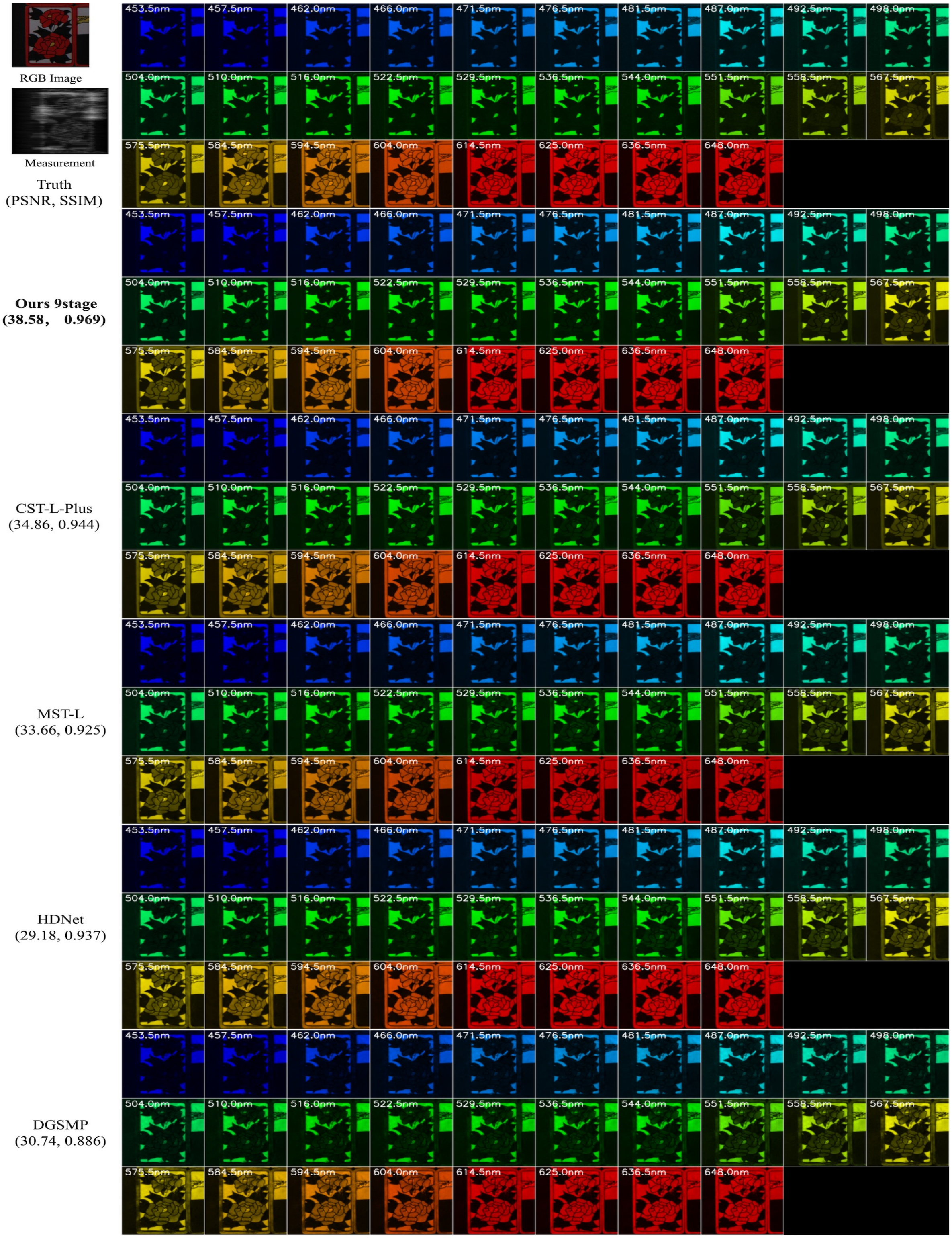

Figure 8. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 38.58dB, SSIM = 0.969), CST-L-Plus [9] (PSNR = 34.86dB, SSIM = 0.944), MST-L [3] (PSNR = 33.66dB, SSIM = 0.925), HDNet [6] (PSNR = 29.18dB, SSIM = 0.937) and DGSMP [7] (PSNR = 30.74dB, SSIM = 0.886) for *Scene7*. Zoom in for better view.

.

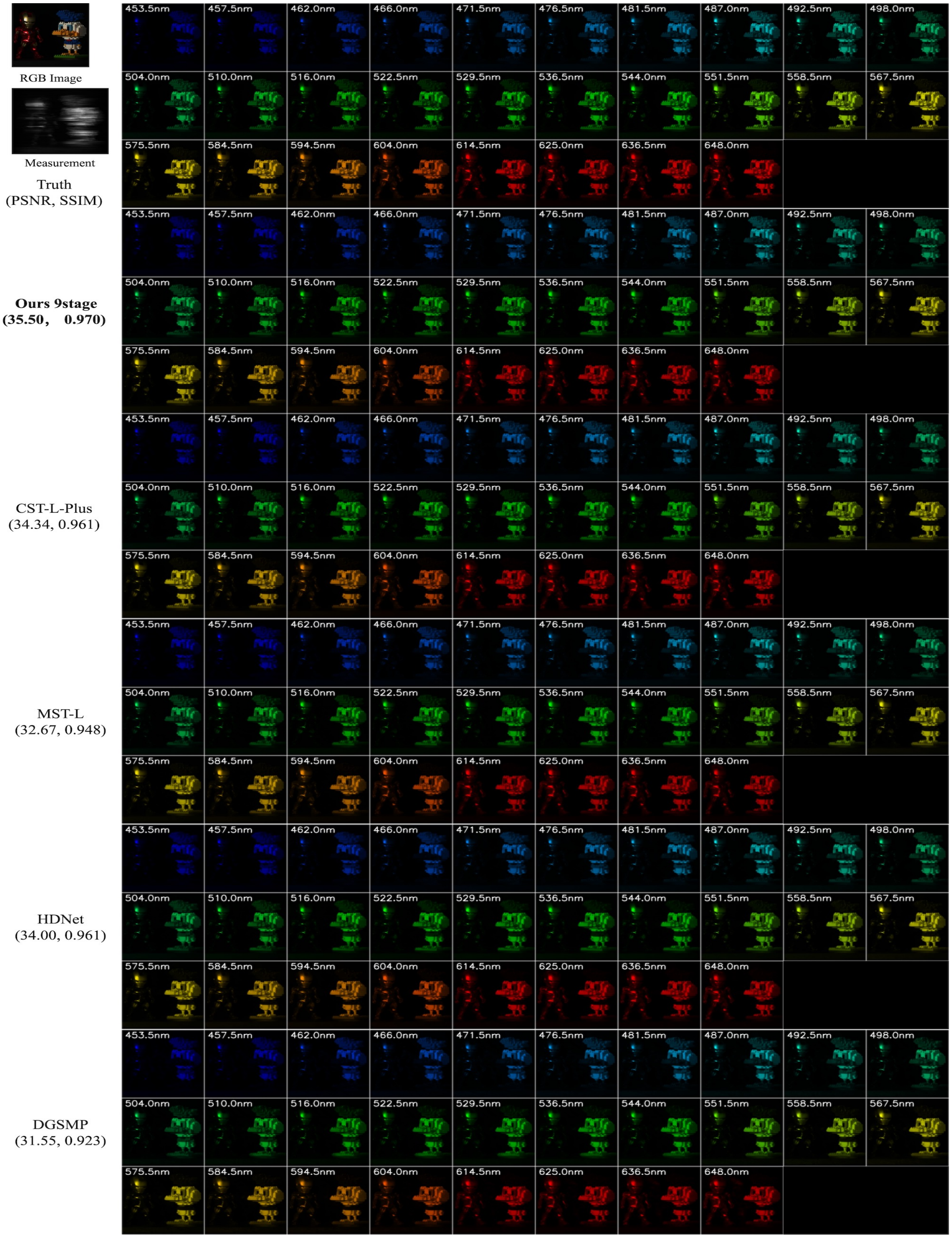

Figure 9. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 35.50dB, SSIM = 0.970), CST-L-Plus [9] (PSNR = 34.34dB, SSIM = 0.961), MST-L [3] (PSNR = 32.67dB, SSIM = 0.948), HDNet [6] (PSNR = 34.00dB, SSIM = 0.961) and DGSMP [7] (PSNR = 31.55dB, SSIM = 0.923) for *Scene8*. Zoom in for better view.

.

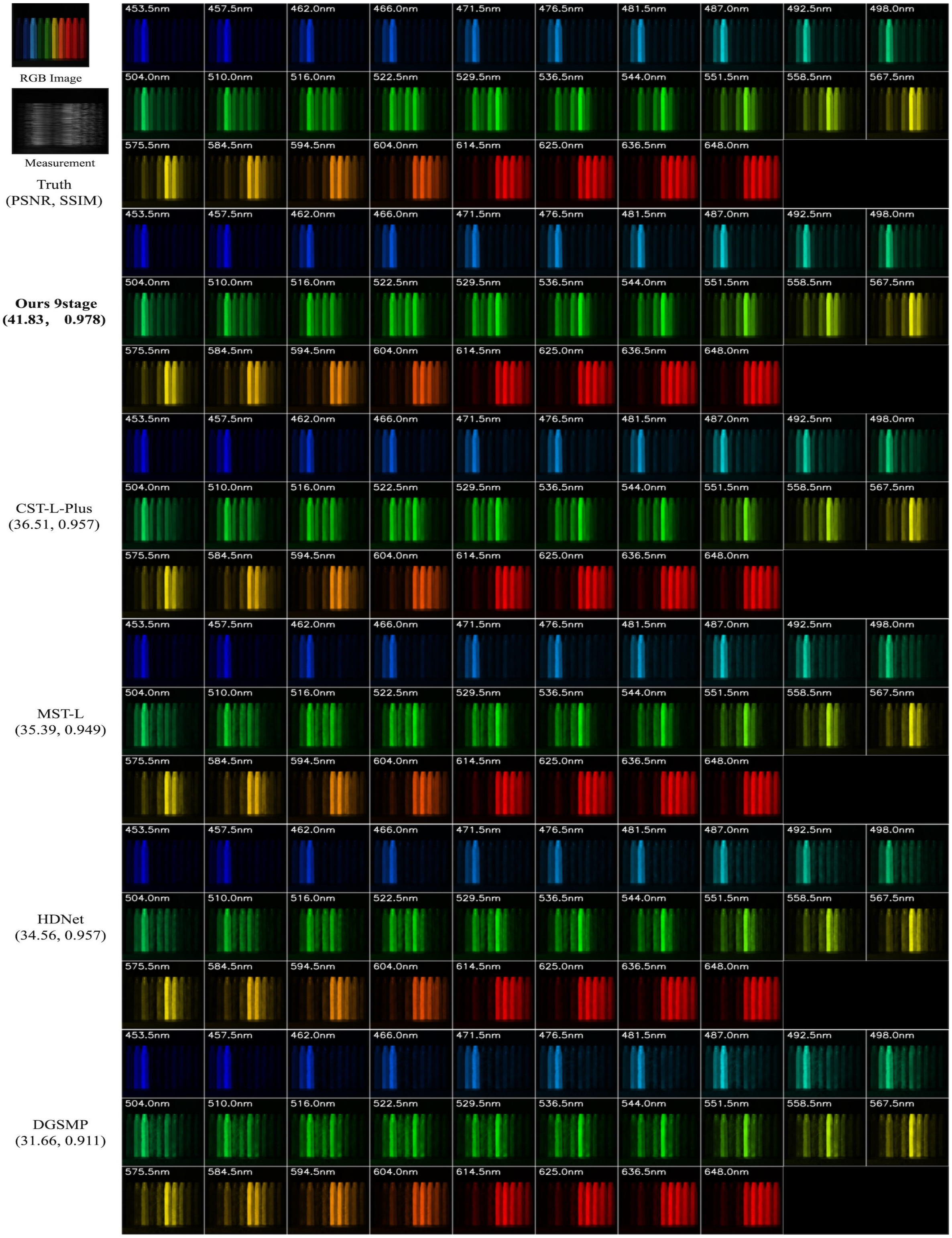
RGB Image
Measurement
Truth
(PSNR, SSIM)
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
Ours 9stage
(41.83， 0.978)
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
CST-L-Plus
(36.51, 0.957)
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
MST-L
(35.39, 0.949)
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
HDNet
(34.56, 0.957)
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
DGSMP
(31.66, 0.911)
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm

Figure 10. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 41.83dB, SSIM = 0.978), CST-L-Plus [9] (PSNR = 36.51dB, SSIM = 0.957), MST-L [3] (PSNR = 35.39dB, SSIM = 0.949), HDNet [6] (PSNR = 34.56dB, SSIM = 0.957) and DGSMP [7] (PSNR = 31.66dB, SSIM = 0.911) for *Scene9*. Zoom in for better view.

.

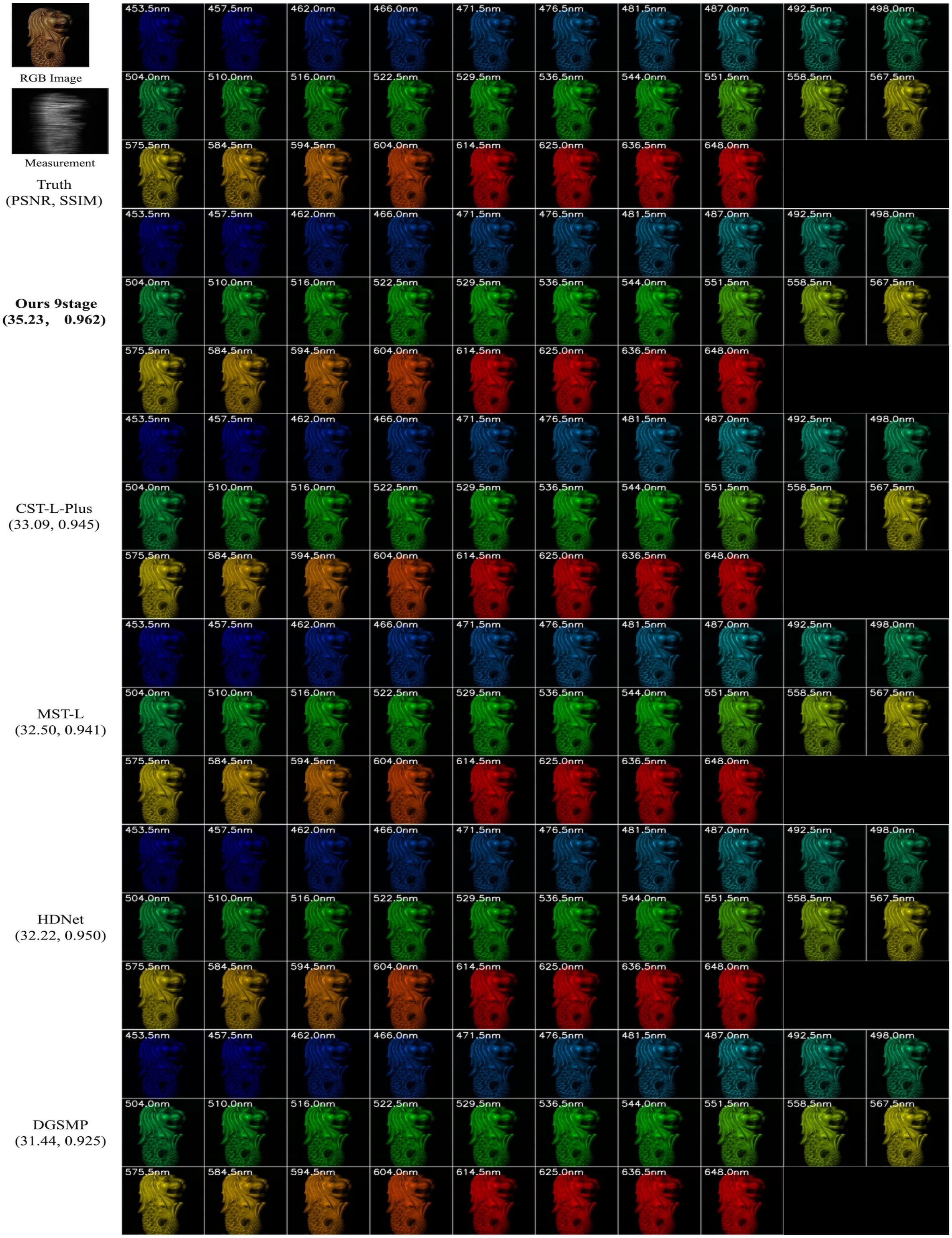

Figure 11. Simulation: RGB image, measurement, ground truth and reconstructed results by the proposed method with 9stage (PSNR = 35.23dB, SSIM = 0.962), CST-L-Plus [9] (PSNR = 33.09dB, SSIM = 0.945), MST-L [3] (PSNR = 32.50dB, SSIM = 0.941), HDNet [6] (PSNR = 32.22dB, SSIM = 0.950) and DGSMP [7] (PSNR = 31.44dB, SSIM = 0.925) for *Scene10*. Zoom in for better view.

.

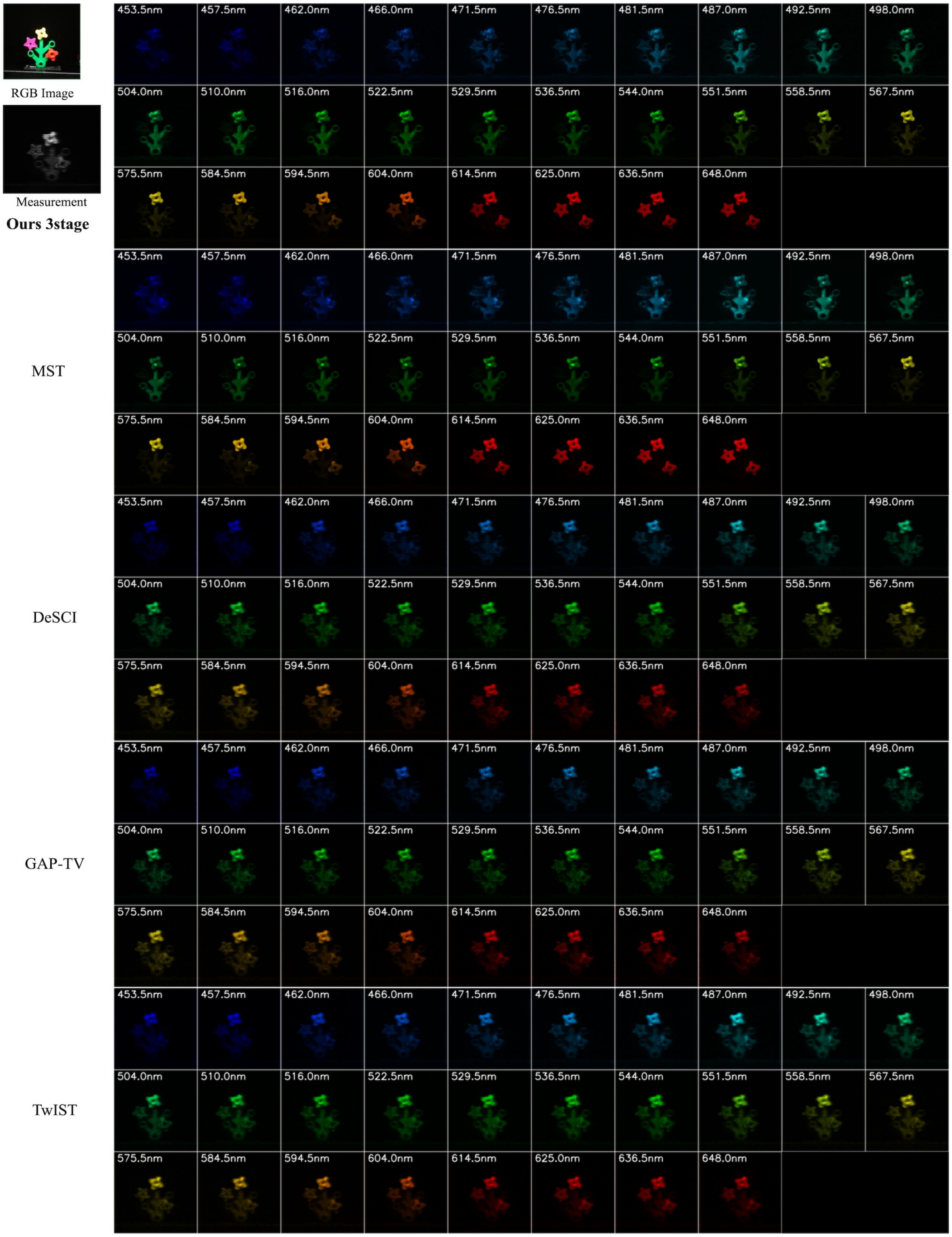

Figure 12. Real data: RGB image, measurement and reconstructed results by the proposed method with 3stage, MST [3], DeSCI [10], GAP-TV [15] and TwIST [1] for *Scene1*. Zoom in for better view.

.

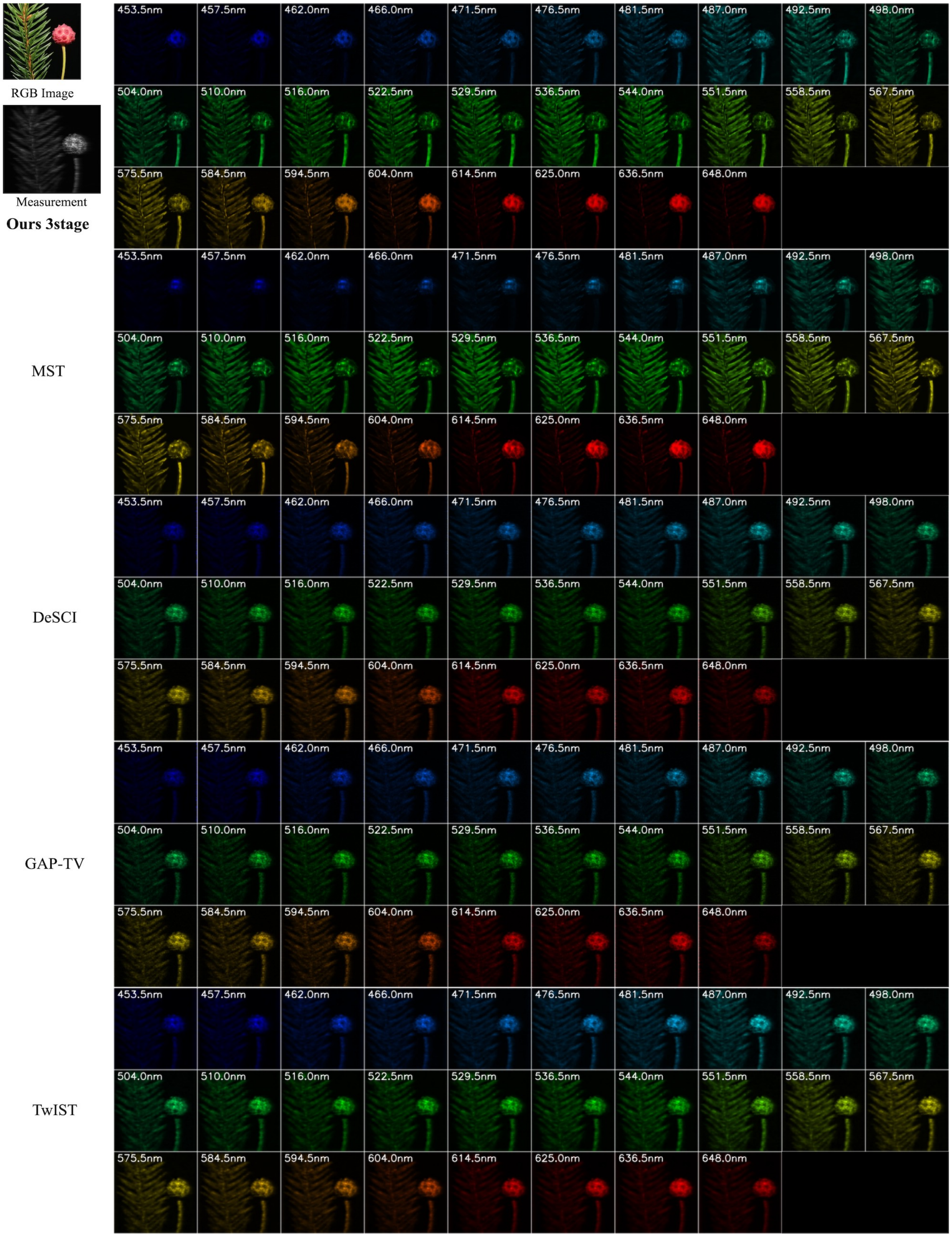
RGB Image
Measurement
Ours 3stage
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
MST
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
DeSCI
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
GAP-TV
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm
TwIST
453.5nm 457.5nm 462.0nm 466.0nm 471.5nm 476.5nm 481.5nm 487.0nm 492.5nm 498.0nm
504.0nm 510.0nm 516.0nm 522.5nm 529.5nm 536.5nm 544.0nm 551.5nm 558.5nm 567.5nm
575.5nm 584.5nm 594.5nm 604.0nm 614.5nm 625.0nm 636.5nm 648.0nm

Figure 13. Real data: RGB image, measurement and reconstructed results by the proposed method with 3stage, MST [3], DeSCI [10], GAP-TV [15] and TwIST [1] for *Scene2*. Zoom in for better view.

.

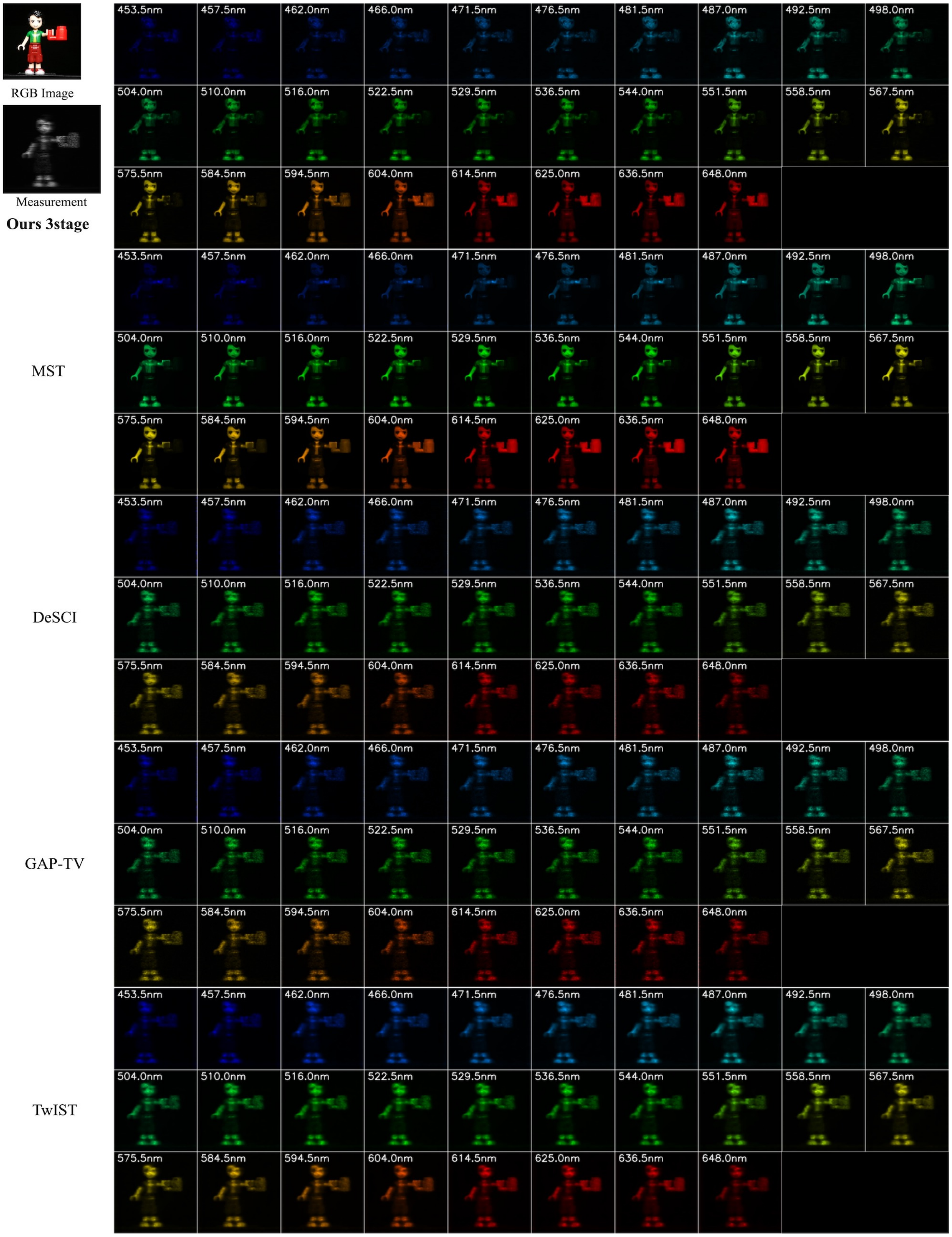

Figure 14. Real data: RGB image, measurement and reconstructed results by the proposed method with 3stage, MST [3], DeSCI [10], GAP-TV [15] and TwIST [1] for *Scene3*. Zoom in for better view.

.

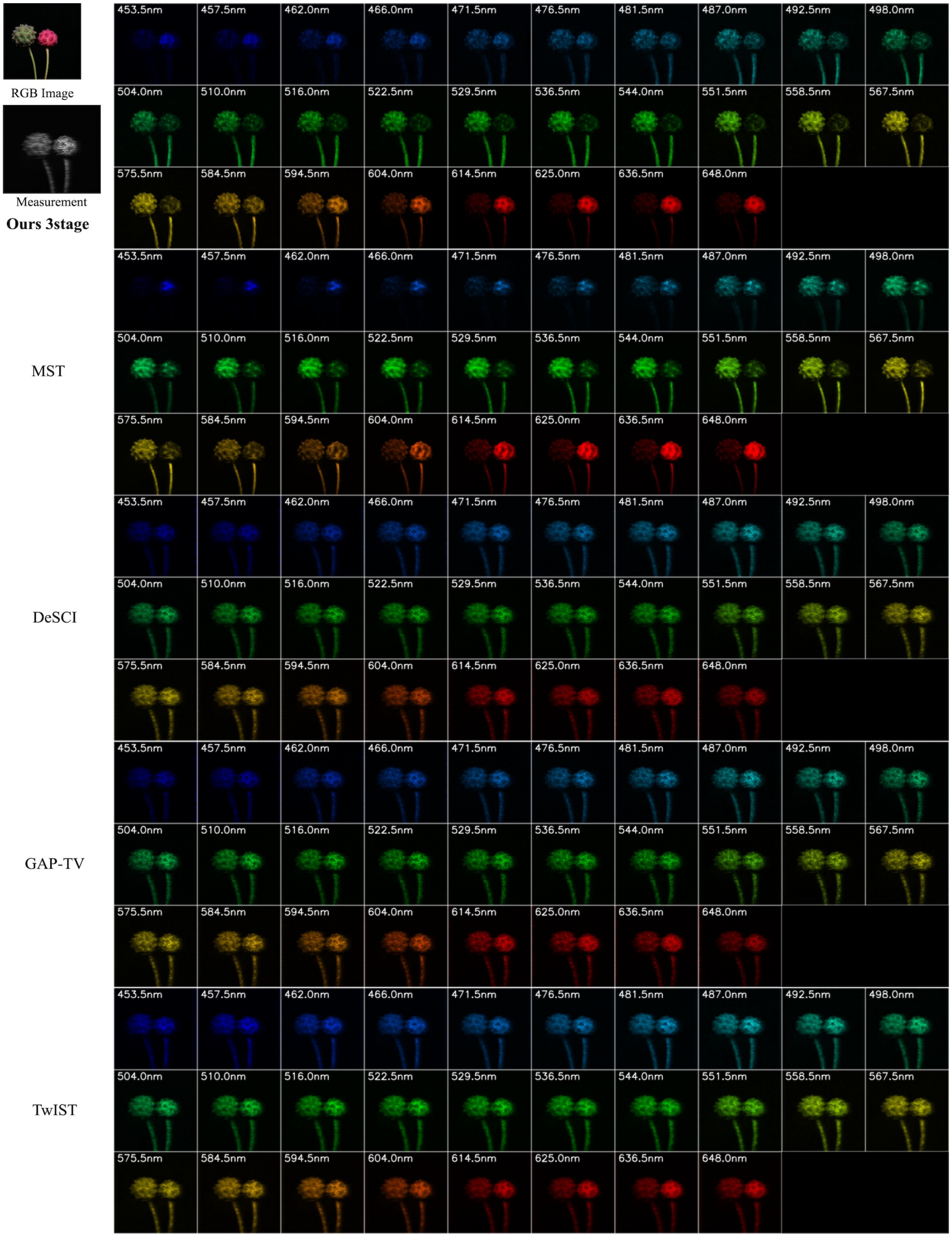

Figure 15. Real data: RGB image, measurement and reconstructed results by the proposed method with 3stage, MST [3], DeSCI [10], GAP-TV [15] and TwIST [1] for *Scene4*. Zoom in for better view.

.

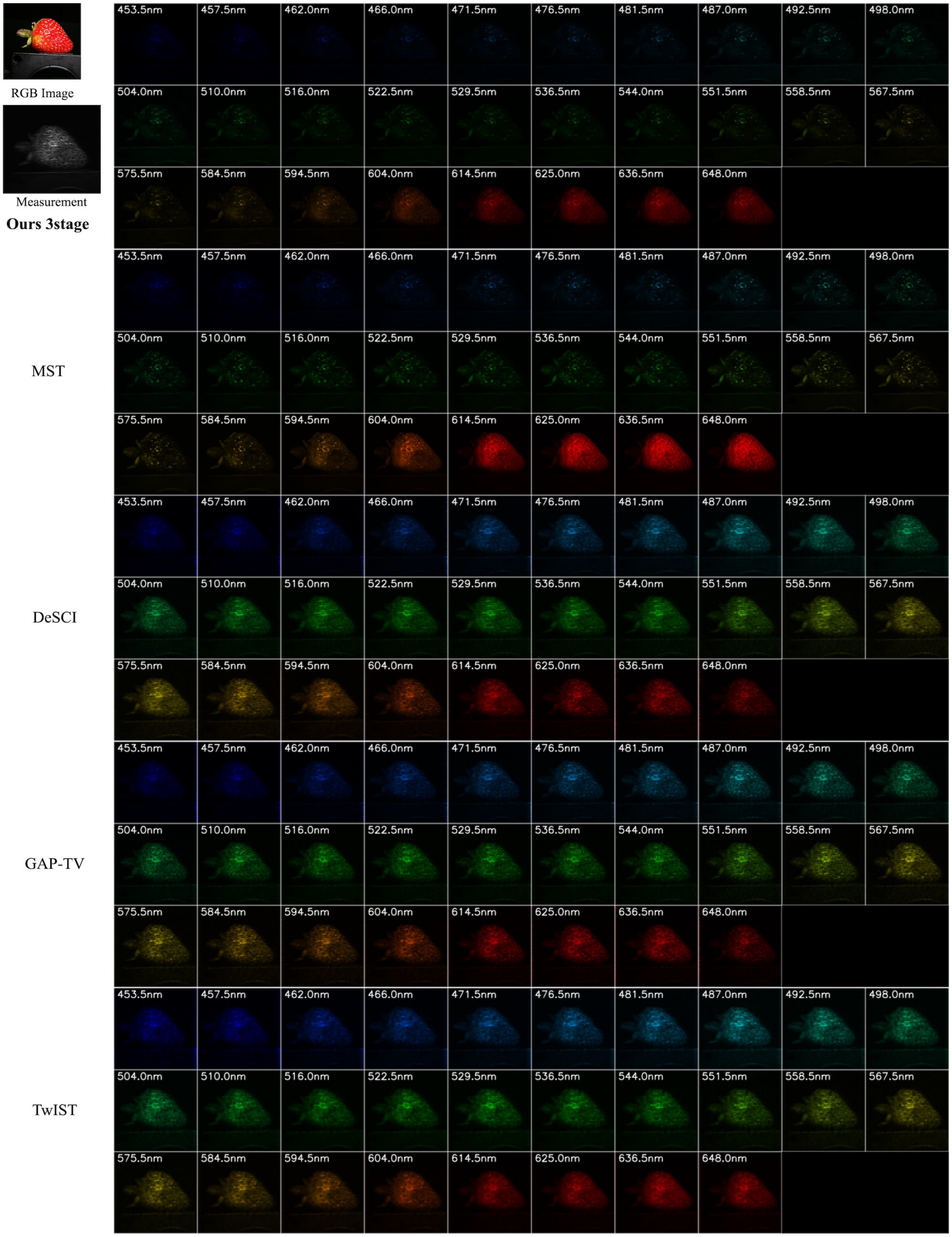

Figure 16. Real data: RGB image, measurement and reconstructed results by the proposed method with 3stage, MST [3], DeSCI [10], GAP-TV [15] and TwIST [1] for *Scene5*. Zoom in for better view.

.

References

[1] José M Bioucas-Dias and Mário AT Figueiredo. A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Transactions on Image processing*, 16(12):2992–3004, 2007. 1, 2, 13, 14, 15, 16, 17

[2] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11036–11045, 2019. 2

[3] Yuanhao Cai, Jing Lin, Xiaowan Hu, Haoqian Wang, Xin Yuan, Yulun Zhang, Radu Timofte, and Luc Van Gool. Mask-guided spectral-wise transformer for efficient hyperspectral image reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17502–17511, 2022. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

[4] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3291–3300, 2018. 2

[5] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4641–4650, 2021. 2

[6] Xiaowan Hu, Yuanhao Cai, Jing Lin, Haoqian Wang, Xin Yuan, Yulun Zhang, Radu Timofte, and Luc Van Gool. Hdnet: High-resolution dual-domain learning for spectral compressive imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17542–17551, 2022. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[7] Tao Huang, Weisheng Dong, Xin Yuan, Jinjian Wu, and Guangming Shi. Deep gaussian scale mixture prior for spectral compressive imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16216–16225, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[8] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8878–8887, 2019. 2

[9] Jing Lin, Yuanhao Cai, Xiaowan Hu, Haoqian Wang, Xin Yuan, Yulun Zhang, Radu Timofte, and Luc Van Gool. Coarse-to-fine sparse transformer for hyperspectral image reconstruction. *arXiv preprint arXiv:2203.04845*, 2022. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

[10] Yang Liu, Xin Yuan, Jinli Suo, David J Brady, and Qionghai Dai. Rank minimization for snapshot compressive imaging. *IEEE transactions on pattern analysis and machine intelligence*, 41(12):2990–3006, 2018. 1, 2, 13, 14, 15, 16, 17

[11] Chong Mou, Qian Wang, and Jian Zhang. Deep generalized unfolding networks for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17399–17410, 2022. 2

[12] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019. 2

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[14] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 1

[15] Xin Yuan. Generalized alternating projection based total variation minimization for compressive sensing. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2539–2543. IEEE, 2016. 1, 2, 13, 14, 15, 16, 17