Outlier-Robust Diffusion Solvers for Inverse Problems

Supplementary Material

A. Comparison with IRLS-PnPDP

Both our methods and the recent work, IRLS-PnPDP [32], employ the well-established iteratively reweighted least squares (IRLS) strategy to address outlier problems in IPs. Our methods leverage this strategy to mitigate outliers within the Huber loss framework, which differentially penalizes small and large residuals to limit the influence of outliers while preserving all measurement information. IRLS-PnPDP uses the strategy to solve an ℓ_q-norm minimization problem derived from a generalized Gaussian scale mixture model. While both approaches achieve robustness through iterative reweighting, the Huber loss offers a smooth transition between quadratic and linear penalties controlled by a single threshold parameter, whereas the ℓ_q-norm with $q < 1$ encourages sparser solutions at the cost of introducing non-convexity. Additionally, we provide a conjugate gradient method as an alternative to gradient descent, avoiding the need for delicate learning rate tuning via an efficient line search strategy.

B. More experimental results

In this section, we first present the detailed setup of the hyperparameter set in our proposed Robust-GD and Robust-CG methods for different inverse problems in Section B.1. We then present the performance of our methods compared to other approaches on the FFHQ dataset in Section B.2. We also validate the effect of the Huber loss threshold parameter δ and the learning rate η_x in Robust-GD, as well as the use of $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$ versus $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{d}$ for calculating α in Robust-CG, in Section B.3.

B.1. Parameter Setup

We present the detailed setup of the hyperparameters used for Robust-GD and Robust-CG in Table 7. Here, N denotes the total number of sample steps and J denotes the total number of iterations. δ is the Huber loss threshold parameter, η_x is the learning rate used in Robust-GD, and η is the finite-difference approximation parameter used in Robust-CG.

	Super-resolution ($4\times$)		**Inpainting (random** 70%**)**		**Gaussian Deblurring**		**Motion Deblurring**		**Nonlinear Deblurring**	
Hyperparameter	**Robust-GD**	**Robust-CG**	**Robust-GD**	**Robust-CG**	**Robust-GD**	**Robust-CG**	**Robust-GD**	**Robust-CG**	**Robust-GD**	**Robust-CG**
N	200	200	200	200	200	200	200	200	200	200
J	100	20	100	100	100	20	100	20	100	50
δ	0.020	0.005	0.010	0.020	0.020	0.020	0.020	0.020	0.010	0.010
η_x	0.0001	/	0.0001	/	0.0001	/	0.00005	/	0.00005	/
η	/	0.0001	/	0.0001	/	0.0001	/	0.0001	/	0.0001

Table 7. Detailed setup of the hyperparameter set in our proposed Robust-GD and Robust-CG methods.

B.2. Experimental results on FFHQ

We validate the performance of our proposed Robust-GD and Robust-CG methods on the FFHQ dataset for five tasks, namely super-resolution ($4\times$), inpainting (random 70%), Gaussian deblurring, motion deblurring, and nonlinear deblurring. The experimental settings for all tasks are the same as those in Section 4. The results are presented in Table 8 for linear IPs and Table 9 for the nonlinear deblurring task. The results indicate that our methods perform the best across nearly all metrics for all tasks compared to recent DM-based methods.

B.3. Ablation study

We validate the effect of the Huber loss threshold parameter δ and the learning rate η_x on the Robust-GD method. Additionally, we compare the use of the $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$ term versus the $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{d}$ term for calculating α in Robust-CG (refer to Eq. (21)).

B.3.1. Huber loss threshold parameter

To evaluate the effect of the Huber loss threshold parameter δ, we use the same experimental setting as described in Section 4.5.1. We tested the Robust-GD method on 100 validation images from the CelebA dataset for the Gaussian and motion deblurring tasks, using a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$. We chose the value of δ from the set $\{0.005, 0.01, 0.02, 0.04\}$. The results presented in Table 10 indicate that Robust-GD performs similarly

ρ	Methods	**Super-resolution** ($4\times$)				**Inpainting (random** 70%**)**				**Gaussian Deblurring**				**Motion Deblurring**			
		PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
0.02	DPS	22.53 ± 1.79	0.612 ± 0.078	0.228 ± 0.062	95.41	25.26 ± 1.71	0.719 ± 0.060	0.182 ± 0.049	86.95	23.76 ± 1.91	0.643 ± 0.076	**0.183** ± 0.052	85.60	22.00 ± 1.79	0.587 ± 0.081	0.220 ± 0.058	93.82
	DiffPIR	19.98 ± 1.50	0.525 ± 0.043	0.509 ± 0.068	263.99	25.03 ± 1.39	0.691 ± 0.032	0.310 ± 0.074	146.31	21.49 ± 0.97	0.443 ± 0.062	0.573 ± 0.084	229.75	19.13 ± 0.82	0.322 ± 0.062	0.642 ± 0.082	284.05
	DCPS	20.39 ± 2.21	0.612 ± 0.084	0.348 ± 0.097	173.42	29.31 ± 2.03	**0.849** ± 0.039	0.145 ± 0.048	81.91	21.45 ± 2.05	0.548 ± 0.086	0.297 ± 0.080	158.54	20.82 ± 3.68	0.596 ± 0.150	0.315 ± 0.158	131.96
	RED-diff	21.73 ± 1.52	0.666 ± 0.031	0.480 ± 0.066	215.67	23.86 ± 1.29	0.646 ± 0.027	0.392 ± 0.072	160.64	26.50 ± 2.02	0.727 ± 0.053	0.318 ± 0.071	103.89	23.42 ± 1.75	0.546 ± 0.060	0.410 ± 0.076	157.78
	DAPS	21.92 ± 1.69	0.689 ± 0.031	0.440 ± 0.075	264.15	24.46 ± 1.67	0.694 ± 0.036	0.308 ± 0.087	203.10	22.83 ± 2.27	0.565 ± 0.086	0.467 ± 0.080	235.16	22.35 ± 2.47	0.585 ± 0.104	0.374 ± 0.118	230.03
	Robust-GD	27.83 ± 1.93	0.780 ± 0.039	0.226 ± 0.060	91.25	28.95 ± 1.77	0.808 ± 0.022	0.087 ± 0.023	67.89	**28.63** ± 2.24	**0.812** ± 0.045	0.200 ± 0.056	**78.54**	**28.57** ± 2.21	**0.795** ± 0.046	**0.131** ± 0.051	**68.53**
	Robust-CG	**28.16** ± 2.21	**0.814** ± 0.045	**0.186** ± 0.050	**75.58**	**29.51** ± 2.22	0.847 ± 0.023	**0.070** ± 0.017	**53.15**	28.04 ± 2.35	0.804 ± 0.054	0.232 ± 0.065	87.66	27.79 ± 2.61	0.783 ± 0.062	0.162 ± 0.062	77.63
0.10	DPS	19.89 ± 1.78	0.560 ± 0.086	0.262 ± 0.070	147.83	21.36 ± 1.66	0.641 ± 0.087	0.226 ± 0.076	152.70	21.07 ± 1.55	0.602 ± 0.080	0.214 ± 0.056	91.95	19.79 ± 1.52	0.542 ± 0.085	0.250 ± 0.059	100.22
	DiffPIR	13.79 ± 1.43	0.244 ± 0.065	0.705 ± 0.067	335.80	19.25 ± 1.50	0.432 ± 0.076	0.599 ± 0.113	191.14	17.30 ± 0.86	0.310 ± 0.064	0.685 ± 0.070	332.72	15.98 ± 0.73	0.221 ± 0.057	0.744 ± 0.076	329.52
	DCPS	14.96 ± 1.31	0.364 ± 0.082	0.590 ± 0.076	271.28	23.58 ± 1.59	0.787 ± 0.048	0.186 ± 0.053	95.24	13.94 ± 1.51	0.246 ± 0.100	0.621 ± 0.093	313.75	11.65 ± 2.81	0.155 ± 0.157	0.920 ± 0.250	332.83
	RED-diff	15.83 ± 1.61	0.436 ± 0.055	0.659 ± 0.073	281.44	19.36 ± 1.42	0.426 ± 0.066	0.631 ± 0.107	186.58	22.17 ± 1.57	0.600 ± 0.067	0.440 ± 0.082	158.14	17.02 ± 2.06	0.296 ± 0.088	0.828 ± 0.143	335.15
	DAPS	15.60 ± 1.94	0.395 ± 0.075	0.650 ± 0.082	319.75	18.20 ± 1.67	0.427 ± 0.082	0.615 ± 0.115	282.18	14.52 ± 2.17	0.287 ± 0.103	0.654 ± 0.081	348.48	13.92 ± 2.21	0.263 ± 0.116	0.766 ± 0.140	375.09
	Robust-GD	24.15 ± 1.36	0.648 ± 0.040	0.417 ± 0.080	141.06	20.27 ± 1.08	0.494 ± 0.060	0.500 ± 0.088	197.86	**28.37** ± 2.17	**0.804** ± 0.045	**0.209** ± 0.056	**82.02**	26.36 ± 2.11	0.710 ± 0.045	**0.153** ± 0.050	**70.98**
	Robust-CG	**27.47** ± 2.17	**0.800** ± 0.046	**0.194** ± 0.052	**76.75**	**28.24** ± 1.95	**0.799** ± 0.022	**0.087** ± 0.022	**59.16**	27.96 ± 2.32	0.802 ± 0.054	0.234 ± 0.065	86.58	**27.51** ± 2.42	**0.767** ± 0.060	0.168 ± 0.059	81.94

Table 8. (**Linear IPs**) **Super-resolution** ($4\times$), **inpainting (random** 70%), **Gaussian deblurring** and **motion deblurring** with additive Gaussian noise ($\sigma = 0.05$) and contamination fraction $\rho = 0.02$ or 0.10.

	$\rho = 0.02$				$\rho = 0.10$			
Methods	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
DPS DPS	22.64 ± 1.96	0.607 ± 0.089	0.227 ± 0.066	94.15	20.50 ± 1.58	0.575 ± 0.085	0.242 ± 0.064	94.60
RED-diff	21.93 ± 1.29	0.487 ± 0.052	0.563 ± 0.101	155.26	15.98 ± 1.56	0.263 ± 0.071	0.923 ± 0.136	274.57
DAPS	20.63 ± 1.56	0.447 ± 0.077	0.635 ± 0.161	300.53	15.40 ± 1.62	0.217 ± 0.068	1.044 ± 0.123	418.32
Robust-GD	**27.57** ± 1.66	**0.768** ± 0.038	**0.137** ± 0.040	**71.36**	**26.36** ± 1.44	0.710 ± 0.036	**0.153** ± 0.043	**77.31**
Robust-CG	25.90 ± 1.86	0.724 ± 0.056	0.210 ± 0.062	95.05	25.83 ± 1.80	**0.719** ± 0.052	0.207 ± 0.051	95.47

Table 9. **Nonlinear deblurring** with additive Gaussian noise ($\sigma = 0.05$) and contamination fraction $\rho = 0.02$ or 0.10.

across these different threshold values, thereby demonstrating the robustness of our methods to the selection of the threshold parameter.

	Gaussian Deblurring			**Motion Deblurring**		
Threshold δ	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
0.005	29.62	0.826	0.141	28.42	0.796	0.152
0.010	29.80	0.827	0.136	29.33	0.806	0.128
0.020	29.27	0.817	0.138	29.44	0.789	0.123
0.040	29.21	0.790	0.160	28.52	0.734	0.156

Table 10. Performance of Robust-CG with different value of the Huber loss threshold δ on **Gaussian deblurring** and **motion deblurring** with additive Gaussian noise ($\sigma = 0.05$) and contamination fraction 0.10.

B.3.2. Learning rate in Robust-GD

We validate the effect of the learning rate η_x in the Robust-GD method on five tasks, namely super-resolution ($4\times$), inpainting (random 70%), Gaussian deblurring, motion deblurring and nonlinear deblurring tasks. We use 100 validation images from the CelebA dataset with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$. We choose five values of η_x from the set $\{0.0010, 0.0005, 0.0001, 0.00005, 0.00001\}$. The results presented in Table 11 indicate that the performance of Robust-GD on five tasks is sensitive to the choice of η_x, thereby motivating the proposition of Robust-CG to avoid the delicate fine-tuning of the learning rate required by Robust-GD.

	Super-resolution ($4\times$)			**Inpainting (random** 70%**)**			**Gaussian Deblurring**			**Motion Deblurring**			**Nonlinear Deblurring**		
Learning Rate η_x	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
0.0010	16.20	0.413	0.625	19.16	0.438	0.585	25.32	0.697	0.215	19.75	0.357	0.646	14.44	0.173	1.160
0.0005	16.60	0.429	0.613	19.18	0.445	0.583	28.49	0.756	0.182	22.11	0.431	0.541	14.36	0.174	1.181
0.0001	26.52	0.716	0.264	23.28	0.575	0.374	29.27	0.817	0.138	28.53	0.734	0.154	24.52	0.582	0.269
0.00005	28.29	0.799	0.145	28.29	0.750	0.132	29.82	0.826	0.136	29.44	0.789	0.123	27.06	0.711	0.156
0.00001	15.84	0.546	0.405	23.62	0.776	0.176	29.21	0.814	0.154	28.08	0.789	0.160	22.84	0.674	0.225

Table 11. Performance of Robust-GD with different value of learning rate η_x on **super-resolution**, **inpainting**, **Gaussian deblurring**, **motion deblurring** and **nonlinear deblurring** with Gaussian noise ($\sigma = 0.05$) and contamination fraction $\rho = 0.10$.

B.3.3. Conjugate gradient term in Robust-CG

We validate the performance of Robust-CG using the $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$ term versus the $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{d}$ term for calculating α (refer to Eq. (21)) on 100 validation images from the CelebA dataset. The evaluation covers five tasks, namely super-resolution ($4\times$), inpainting

(random 70%), Gaussian deblurring, motion deblurring, and nonlinear deblurring, each with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$. The results presented in Table 12 indicate that Robust-CG using $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$ exhibits more stable and superior performance across all tasks compared to using $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{d}$.

	Super-resolution (4×)			**Inpainting (random** 70%)			**Gaussian Deblurring**			**Motion Deblurring**			**Nonlinear Deblurring**		
Methods	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
$\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$	28.96	0.819	0.129	29.74	0.809	0.093	29.38	0.819	0.151	28.91	0.777	0.144	26.80	0.728	0.188
$\boldsymbol{g}^{\mathrm{T}}\boldsymbol{d}$	28.07	0.792	0.164	19.15	0.446	0.582	29.51	0.803	0.152	24.50	0.530	0.403	23.75	0.526	0.369

Table 12. Performance of Robust-CG using the $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{g}$ term versus the $\boldsymbol{g}^{\mathrm{T}}\boldsymbol{d}$ term for calculating α on **super-resolution** (4×), **inpainting (random** 70%), **Gaussian deblurring**, **motion deblurring** and **nonlinear deblurring** with additive Gaussian noise ($\sigma = 0.05$) and contamination fraction $\rho = 0.10$.

B.4. Ablation of core components

To validate the effectiveness of measurement refinement and the Huber loss, we incorporate these components into two baseline methods, namely DPS and DiffPIR, and denote the resulting variants as Robust-DPS and Robust-DiffPIR, respectively. The results in Table 13 indicate that, with the incorporation of measurement refinement and the Huber loss, both Robust-DPS and Robust-DiffPIR outperform their original counterparts. Additionally, the proposed methods Robust-GD and Robust-CG achieve the best performance across most evaluation metrics, which suggests that their design is more compatible with these components. Furthermore, we visualize the relationship between the distortion metric PSNR and computational cost, including the average inference time over 100 images for the CelebA Gaussian deblurring task, as well as the number of function evaluations and forward operator evaluations for each algorithm. The results in Figure 6 show that Robust-GD and Robust-CG achieve a favorable trade-off between efficiency and reconstruction performance.

	CelebA (256 × 256)				**FFHQ** (256 × 256)				**CelebA** (256 × 256)				**FFHQ** (256 × 256)			
	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓
Methods	**Gaussian deblurring**								**Motion deblurring**							
DPS	22.06	0.646	0.178	63.49	21.07	0.602	0.214	91.95	20.90	0.596	0.208	68.96	19.19	0.542	0.250	100.22
Robust-DPS	27.70	0.765	**0.099**	**53.82**	26.23	0.742	**0.115**	**62.79**	27.58	0.770	**0.108**	**56.01**	26.00	0.744	**0.127**	**69.63**
DiffPIR	18.07	0.355	0.622	308.93	17.30	0.310	0.685	332.72	17.11	0.259	0.712	340.27	15.98	0.221	0.744	329.52
Robust-DiffPIR	26.53	0.723	0.166	78.30	24.80	0.675	0.227	102.18	23.46	0.621	0.229	106.00	21.91	0.565	0.285	132.55
DCPS	15.46	0.302	0.575	331.91	13.94	0.246	0.621	313.75	12.60	0.174	0.927	338.22	11.65	0.155	0.920	332.83
RED-diff	22.96	0.626	0.391	139.05	22.17	0.600	0.440	158.14	17.36	0.311	0.836	303.61	17.02	0.296	0.828	335.15
DAPS	16.18	0.356	0.561	297.13	14.52	0.287	0.654	348.48	14.99	0.331	0.718	292.81	13.92	0.263	0.766	375.09
Robust-GD	29.27	0.817	0.138	65.59	**28.37**	**0.804**	0.209	82.02	**29.44**	**0.789**	0.123	84.79	26.36	0.710	0.153	70.98
Robust-CG	**29.38**	**0.819**	0.151	62.52	27.96	0.802	0.234	86.58	28.91	0.777	0.144	91.22	**27.51**	**0.767**	0.168	81.94

Table 13. (**Linear IPs**) **Gaussian deblurring** and **motion deblurring** with additive Gaussian noise ($\sigma = 0.05$) and contamination fraction 0.10 on the CelebA and FFHQ datasets.

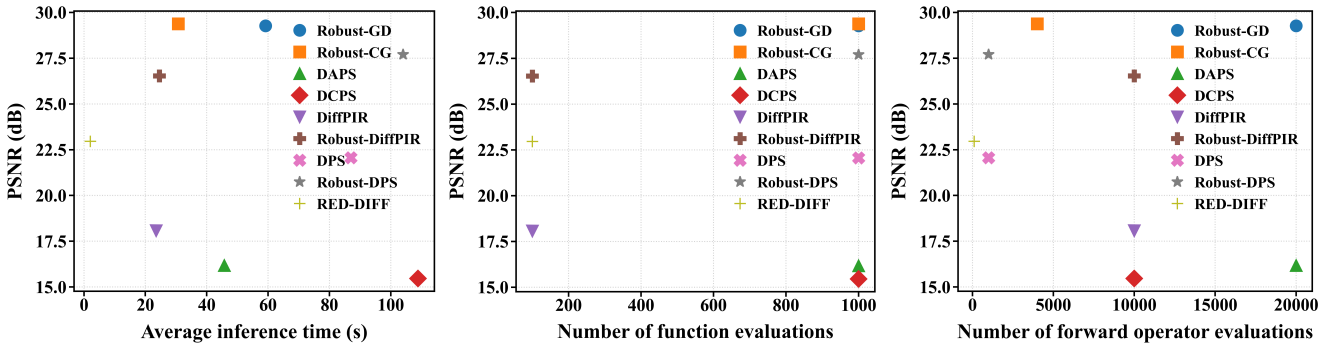

Figure 6. Visualization of the relationship between the distortion metric PSNR and computational cost for the CelebA Gaussian deblurring task. The computational cost is measured in terms of (left) average inference time over 100 images, (middle) number of function evaluations, and (right) number of forward operator evaluations.

C. Visualization of experimental results

We visualize our experimental results on five tasks, namely super-resolution ($4\times$), inpainting (random 70%), Gaussian deblurring, motion deblurring and nonlinear deblurring tasks across three datasets, namely CelebA, FFHQ and ImageNet. The experiments are conducted with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$.

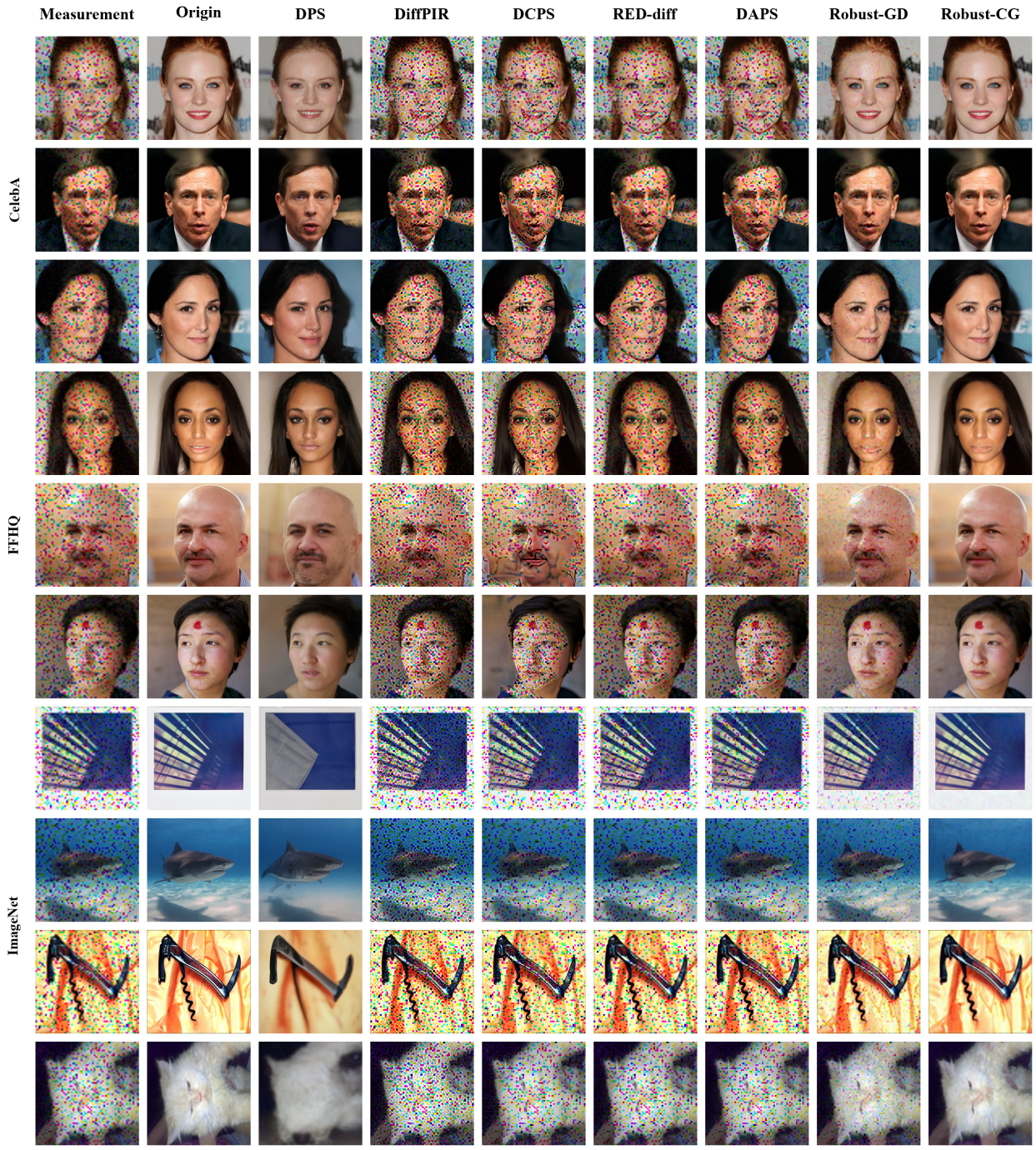

Figure 7. Visualization of the experimental results for the super-resolution ($4\times$) task with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$.

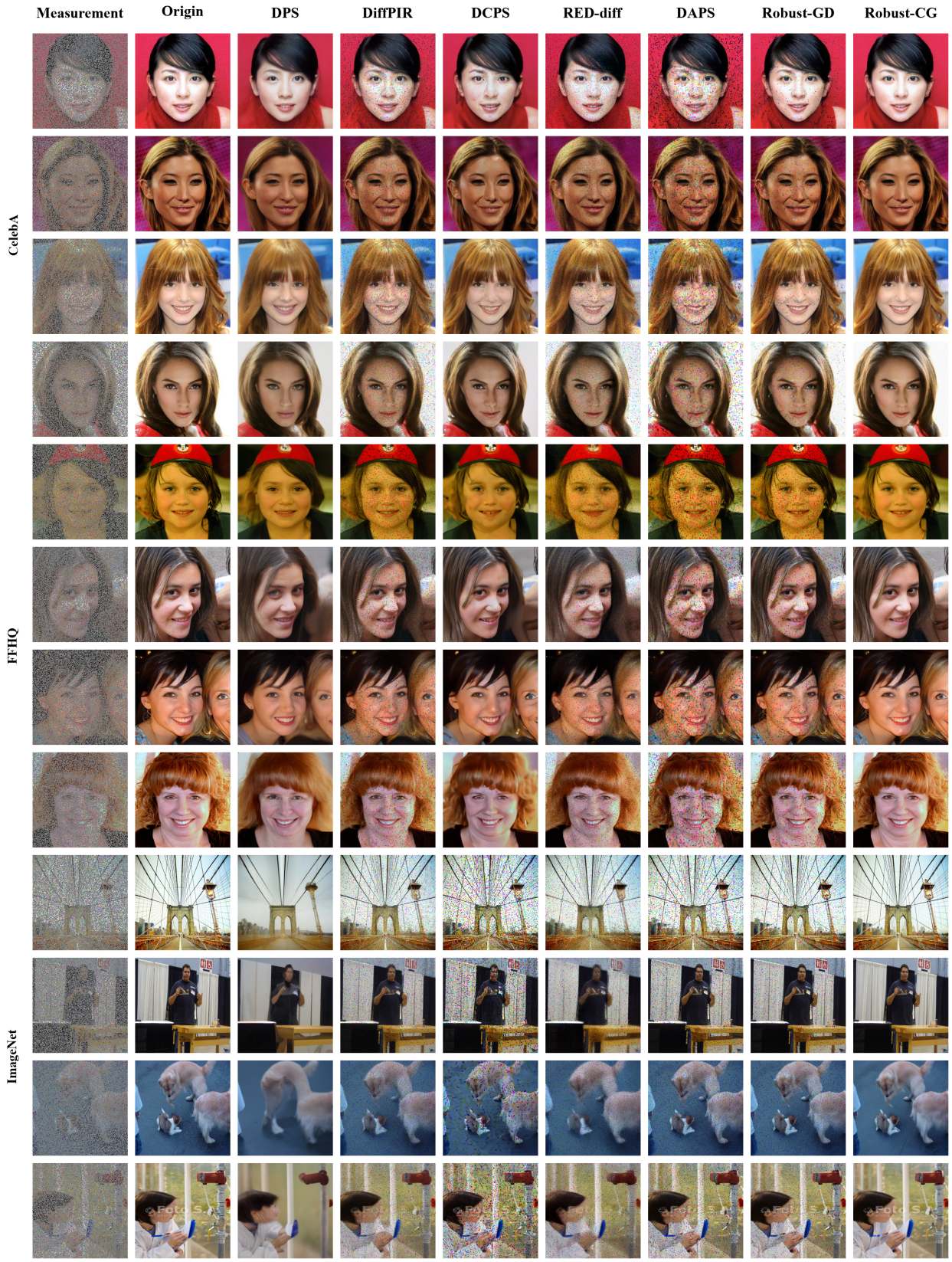

Figure 8. Visualization of the experimental results for the inpainting (random 70%) task with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$.

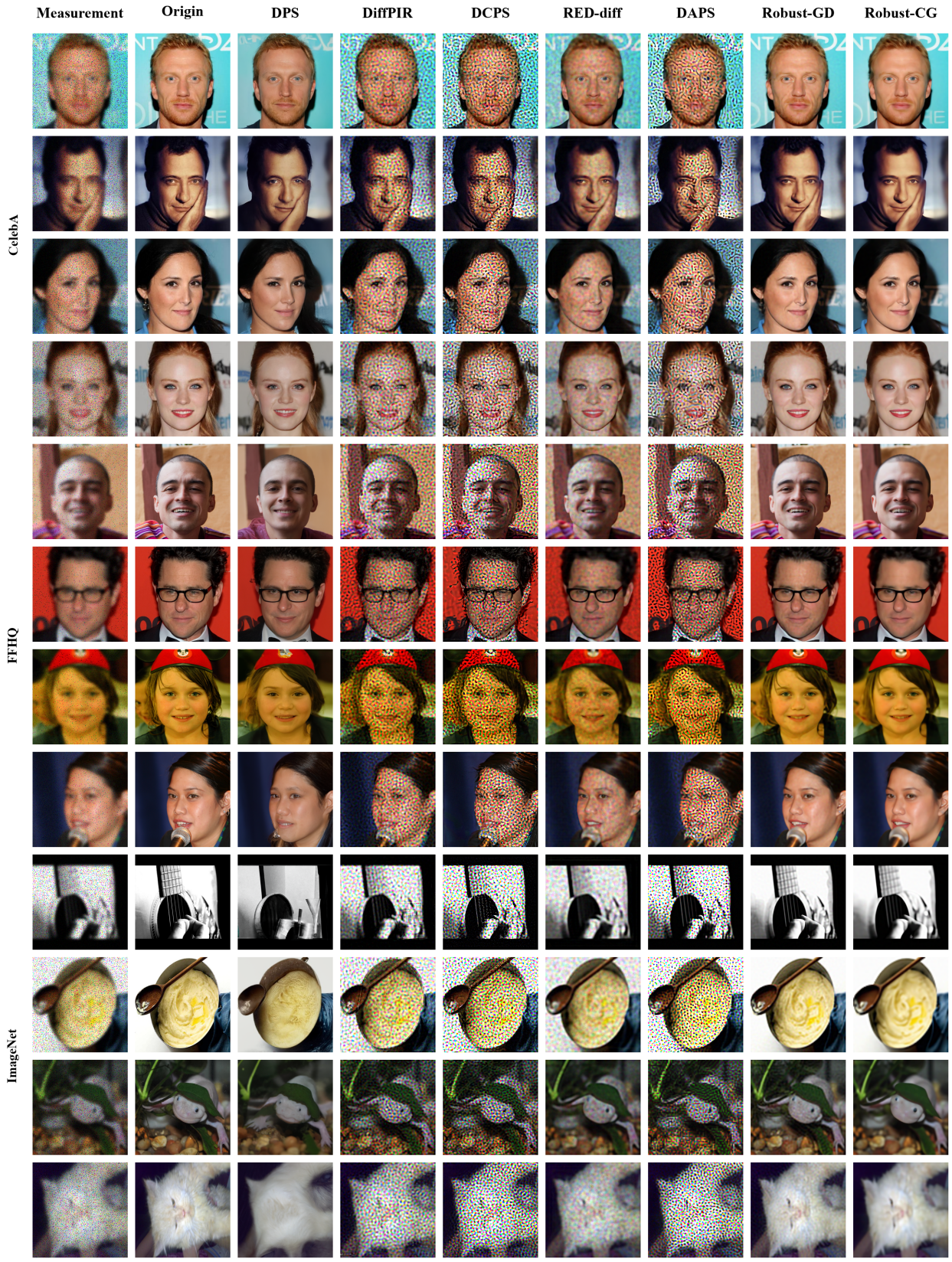

Figure 9. Visualization of the experimental results for the Gaussian deblurring task with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$.

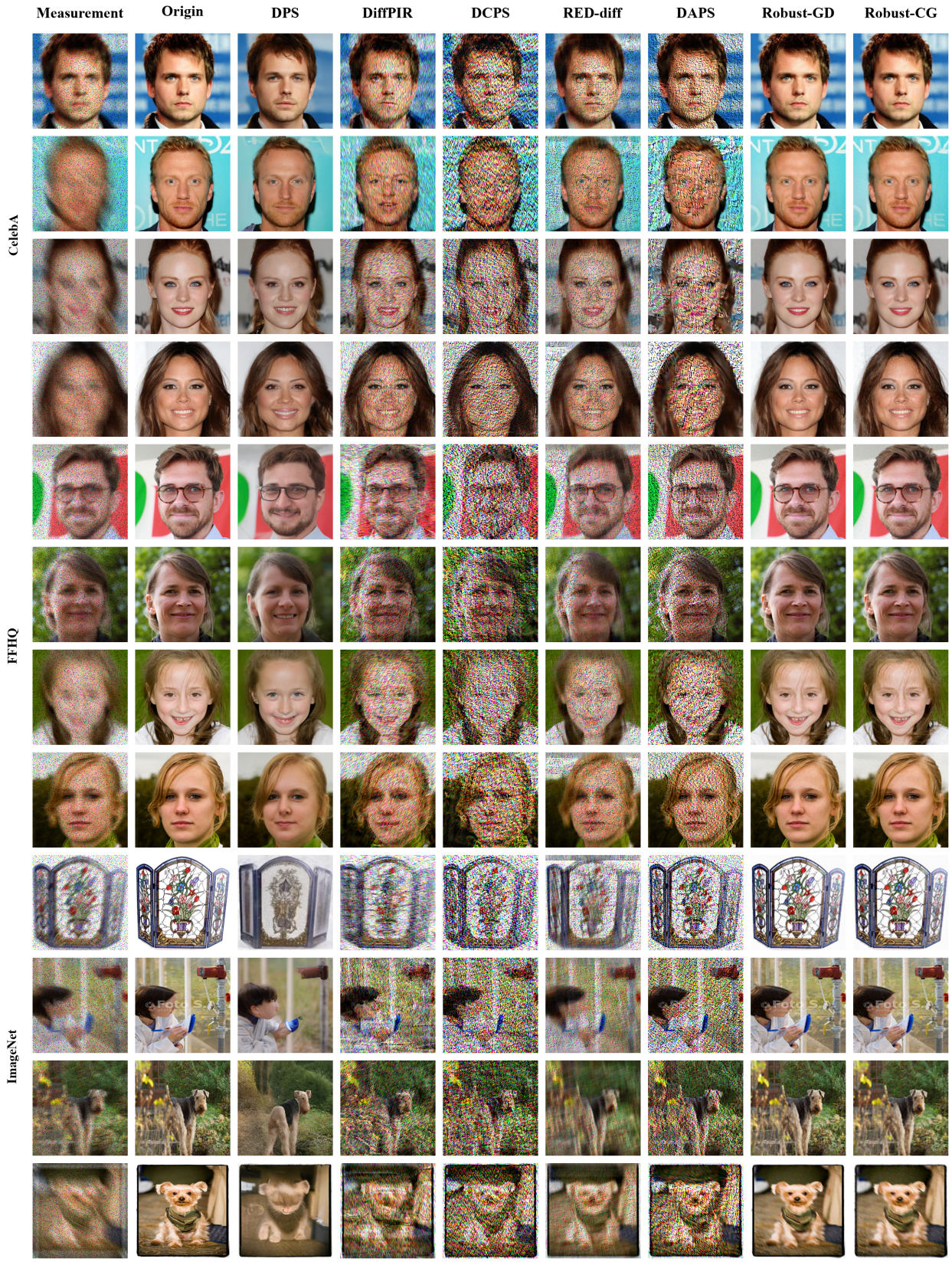

Figure 10. Visualization of the experimental results for the motion deblurring task with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$.

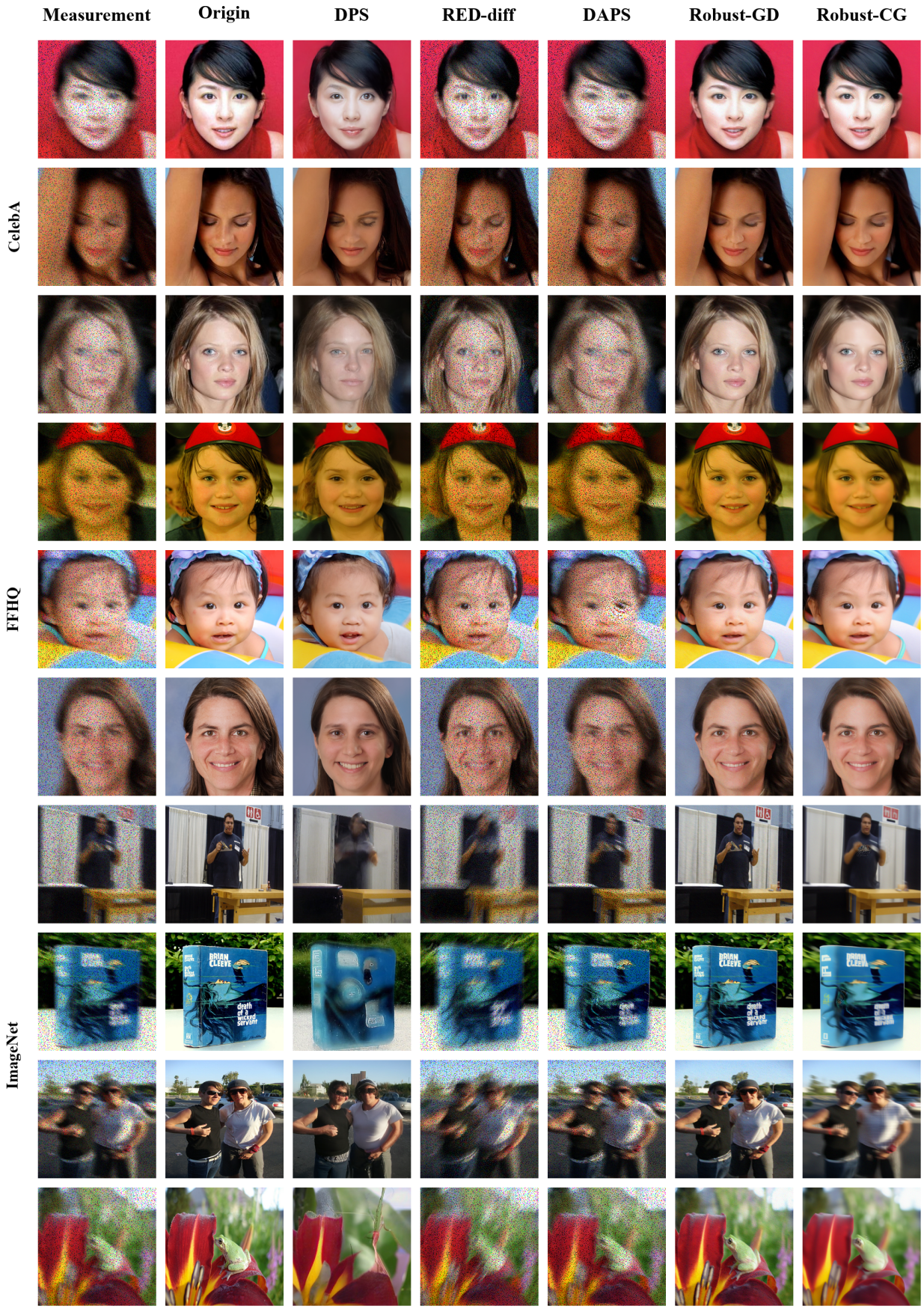

Figure 11. Visualization of the experimental results for the nonlinear deblurring task with a contamination factor of $\rho = 0.10$ and a Gaussian noise level of $\sigma = 0.05$.