CARD: Correlation Aware Restoration with Diffusion Supplementary Material

Niki Nezakati, Arnab Ghosh, Amit Roy-Chowdhury, Vishwanath Saragadam
University of California, Riverside
{nneza001, aghos034, amitrc, vishwans}@ucr.edu

1. Implementation Details

All experiments utilize frozen diffusion backbones identical to those in DDRM [1]. We employ publicly available UNet architectures at 256×256 resolution, the standard checkpoints for LSUN-Bedroom and LSUN-Cat, and the OpenAI/Guided-Diffusion models for ImageNet.

1.1. Sampling configuration

Following DDRM, we sample from a uniformly spaced subset of the $T = 1000$ pretraining timesteps. We perform $K = 20$ neural function evaluations (NFEs) per image through linear subsampling of the original schedule. Through hyperparameter grid search, we identify optimal values of $\eta = 0.80$ for the stochasticity parameter and $\eta_b = 1.0$ for the measurement-blending parameter. Compared to DDRM's default $\eta = 0.85$, our reduced stochasticity consistently improves performance with correlated noise while preserving perceptual quality.

1.2. Degradation models

We adopt the standard linear measurement model

$$\mathbf{y} = H\mathbf{x} + \mathbf{n}, \tag{1}$$

with pixel values normalized to $[0, 1]$. The forward operators H are defined as follows. For denoising, we set $H = I$ (identity operator). For deblurring, we apply separable 1D kernels along horizontal and vertical axes with three blur types: (i) uniform box filter with length 9 (effective 9×9 PSF), (ii) isotropic Gaussian blur with 5-tap kernel ($\sigma = 10$ in code units), and (iii) anisotropic Gaussian blur with orthogonal 9-tap kernels of differing spreads ($\sigma_x = 20$, $\sigma_y = 1$). To address ill-posedness, singular values below 3×10^{-2} are set to zero following standard deblurring implementations. For super-resolution, we employ block-averaging downsamplers with scale factors $2\times$ and $4\times$, applied independently along each spatial dimension. All operators act on $[0, 1]$-normalized images and remain fixed across all compared methods.

1.3. Correlated noise generation and whitening

We model synthetic correlated noise using the covariance structure

$$\Sigma_{\text{synth}} = \sigma^2(I + \alpha B) + \varepsilon I, \tag{2}$$

where B is a symmetric banded adjacency matrix with nonzeros on selected off-diagonals. To maintain computational efficiency, correlation and whitening are applied to nonoverlapping 8×8 patches per channel. For patch size $p = 8$ and dimension $d = p^2 = 64$, we instantiate a $d \times d$ covariance matrix Σ and compute its Cholesky factor L and symmetric inverse square-root $W = \Sigma^{-1/2}$. Correlated noise samples are generated as $\mathbf{n} = L\mathbf{z}$ where $\mathbf{z} \sim \mathcal{N}(0, I)$, and whitening is performed via W such that $W\Sigma W^\top = I$.

1.4. Blind covariance estimation from noisy images

When dark frames are unavailable, we estimate the correlated noise covariance Σ directly from noisy images by leveraging low-content regions.

Patch extraction and low-content selection. We use $p \times p$ patches and estimate the noise covariance independently per color channel, matching our patchwise whitening implementation. Given a noisy image $\mathbf{y}$, we extract patches and rank them using a lightly smoothed image

$$\mathbf{y}_{\text{score}} = G(\mathbf{y}), \tag{3}$$

where G is a small Gaussian blur. We then compute a flatness score on each patch of $\mathbf{y}_{\text{score}}$ using gradient energy, which favors spatially uniform regions. We select the lowest-scoring fraction of patches (top-20% flattest) and take the corresponding patch values from the original noisy image $\mathbf{y}$.

Covariance estimation and stabilization. Let $X \in \mathbb{R}^{M \times d}$ be the matrix formed by stacking the selected patch vectors, after subtracting the empirical mean across samples. We estimate the patch-noise covariance as

$$\widehat{\Sigma}_{\text{noise}} = \frac{1}{M} X^\top X. \tag{4}$$

Table 1. **ImageNet denoising results.** We report PSNR/LPIPS as P/L. Best and second-best results are marked in bold and underlined, respectively. CARD with estimated (blind) Σ shows small performance drop vs. oracle Σ, but still outperforms all other baselines.

Model	Noise Level σ_0		
	0.1 (P↑/L↓)	0.5 (P↑/L↓)	0.9 (P↑/L↓)
Restormer [2]	31.3/0.10	23.9/0.28	18.5/0.59
DDRM [1]	31.0/0.14	24.8/0.33	22.7/0.40
CARD (Blind Σ)	<u>31.8</u>/<u>0.08</u>	<u>26.2</u>/<u>0.17</u>	<u>24.7</u>/<u>0.24</u>
CARD (Oracle Σ)	**34.0/0.07**	**29.1/0.15**	**26.7/0.22**

To further reduce the influence of occasional textured outliers that pass the selection stage, we apply a robust refinement step based on iterative reweighting (Huber) using Mahalanobis distances under the current covariance estimate.

Recovering $\widehat{\Sigma}$ and computing the whitening transform. Under our noise model, the selected patch vectors are dominated by the measurement noise $\mathbf{n}$ from the observation model $\mathbf{y} = H\mathbf{x} + \mathbf{n}$, so $\widehat{\Sigma}_{\text{noise}} \approx \text{Cov}(\mathbf{n}_{\text{patch}})$. In experiments where the noise magnitude σ_0 is fixed and known, we first normalize the covariance via

$$\widehat{\Sigma} \;=\; \widehat{\Sigma}_{\text{noise}}/\sigma_0^2. \tag{5}$$

To apply the whitening transform in CARD, we compute a Cholesky factorization $\widehat{\Sigma} = LL^\top$, yielding the whitening operator $W = L^{-1}$ that satisfies $W\widehat{\Sigma}W^\top = I$. This whitening transform is used in CARD to form whitened measurements and operators.

Evaluation of estimation quality. Table 1 reports ImageNet denoising performance when CARD uses a covariance Σ estimated from noisy images (blind) compared to using the ground-truth covariance (oracle). Although the blind estimate introduces a small performance drop relative to the oracle setting, it still consistently outperforms all other baselines.

1.5. Noise level parameter

The parameter σ_0 controls the measurement-noise magnitude in DDRM's likelihood term. For simulated experiments, we use a consistent noise level σ_0 across all methods. For real-world experiments, we tune σ_0 separately for each baseline to ensure fair comparisons.

1.6. Evaluation metrics

We report three standard metrics, computed per image and then averaged: PSNR (dB) on $[0, 1]$-normalized RGB images, SSIM on $[0, 1]$ scale, and LPIPS using the VGG backbone. All metrics are computed on outputs mapped from the model's native $[-1, 1]$ range back to $[0, 1]$.

1.7. Computational resources

All experiments are executed on a single NVIDIA GeForce RTX 4090 GPU (24 GB) running on an Exxact workstation and CUDA 13.0.

1.8. Latency analysis

We evaluate the runtime and memory overhead introduced by the whitening operation and additional matrix computations. End-to-end inference time and total GPU memory were measured on a single RTX 4090 GPU over 1000 ImageNet images using identical sampling schedules for both DDRM and CARD. DDRM requires 863.66 s total runtime with 1281.4 MB memory allocation, while CARD requires 866.36 s total runtime with 1287.4 MB memory allocation. This corresponds to an overhead of only +2.70 s per 1000 images (≈ 2.7 ms per image, $\approx 0.3\%$ runtime increase) and +6.0 MB of memory.

In practice, whitening introduces negligible computational overhead because the covariance matrix Σ is defined at the patch level, the whitening transform and the corresponding whitened operator decomposition are computed once and reused, and per-image whitening reduces to a lightweight linear transformation. Consequently, the dominant computational cost remains the UNet evaluations in the diffusion model.

2. DDRM Under Whitening

This section derives the DDRM updates when measurements are corrupted by correlated Gaussian noise and whitened prior to conditioning. Whitening modifies only the forward operator and measurements, and the diffusion schedule $\{\sigma_t\}_{t=0}^T$ and denoiser outputs remain unchanged. The resulting equations are DDRM's spectral one-dimensional Gaussians applied to the whitened problem (see Eqs. (4)–(8) in [1]).

2.1. Whitened measurement model

We consider the linear observation model with correlated noise

$$\mathbf{y} = H\mathbf{x}_0 + \mathbf{n}, \qquad \mathbf{n} \sim \mathcal{N}(0, \sigma_y^2\Sigma). \tag{6}$$

To restore independence, we left-precondition by the symmetric inverse square-root $W = \Sigma^{-1/2}$, obtaining

$$\tilde{\mathbf{y}} = W\mathbf{y}, \qquad \tilde{H} = WH, \qquad \tilde{\mathbf{n}} = W\mathbf{n} \sim \mathcal{N}(0, \sigma_y^2 I). \tag{7}$$

All conditioning steps are then carried out on $(\tilde{H}, \tilde{\mathbf{y}})$ with i.i.d. noise. Let $\tilde{H} = \tilde{U}\tilde{S}\tilde{V}^\top$ denote the SVD. We work in spectral coordinates defined by

$$\bar{\tilde{\mathbf{x}}}_t = \tilde{V}^\top \mathbf{x}_t, \qquad \bar{\tilde{\mathbf{y}}} = \tilde{S}^\dagger \tilde{U}^\top \tilde{\mathbf{y}}. \tag{8}$$

The diffusion prior retains the usual Gaussian marginals

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0, \sigma_t^2 I), \qquad 0 = \sigma_0 < \cdots < \sigma_T, \tag{9}$$

so whitening modifies only the measurement-consistency component of the update (replacing $s_i, \bar{y}^{(i)}$ by $\tilde{s}_i, \bar{\tilde{y}}^{(i)}$), while the prior side remains unchanged. For compact notation, we define

$$\delta_i \equiv \sigma_y/\tilde{s}_i, \qquad \alpha_t \equiv \sqrt{1-\eta^2}\,\sigma_t, \qquad \beta_t \equiv \alpha_t/\sigma_{t+1}, \tag{10}$$

and use shorthands $\xi_t^{(i)} := \bar{\tilde{x}}_t^{(i)}$ and $v^{(i)} := \bar{\tilde{y}}^{(i)}$ for spectral coordinates. Following DDRM, we assume $\sigma_T \geq \delta_i$ for all $\tilde{s}_i > 0$ to ensure non-negative initial variance.

2.2. Whitened DDRM updates

Initialization ($t = T$). At the largest noise level, the posterior per spectral coordinate is Gaussian centered at the measurement component for observable coordinates ($\tilde{s}_i > 0$), while unobservable coordinates ($\tilde{s}_i = 0$) follow the unconditional prior:

$$q^{(T)}\big(\xi_T^{(i)} \mid \mathbf{x}_0, \tilde{\mathbf{y}}\big) = \begin{cases} \mathcal{N}\big(v^{(i)},\, \sigma_T^2 - \delta_i^2\big), & \tilde{s}_i > 0, \\ \mathcal{N}\big(\xi_0^{(i)},\, \sigma_T^2\big), & \tilde{s}_i = 0, \end{cases} \tag{11}$$

$$p_\theta^{(T)}\big(\xi_T^{(i)} \mid \tilde{\mathbf{y}}\big) = \begin{cases} \mathcal{N}\big(v^{(i)},\, \sigma_T^2 - \delta_i^2\big), & \tilde{s}_i > 0, \\ \mathcal{N}\big(0,\, \sigma_T^2\big), & \tilde{s}_i = 0. \end{cases} \tag{12}$$

These match DDRM's initializers with the substitutions $(s_i, \bar{y}^{(i)}) \mapsto (\tilde{s}_i, v^{(i)})$.

Transitions ($t < T$): ground truth posterior q. Each step interpolates between the current clean estimate and the measurement across three scenarios. We abbreviate $q^{(t)}(\cdot|\mathbf{x}_{t+1}, \mathbf{x}_0, \tilde{\mathbf{y}})$ as $q^{(t)}(\cdot|\cdot)$:

$$q^{(t)}\big(\xi_t^{(i)} \big|\, \cdot\,\big) = \mathcal{N}\big(\,\xi_0^{(i)} + \beta_t(\xi_{t+1}^{(i)} - \xi_0^{(i)}),\, \eta^2\sigma_t^2\big), \quad \tilde{s}_i = 0, \tag{13}$$

$$q^{(t)}\big(\xi_t^{(i)} \big|\, \cdot\,\big) = \mathcal{N}\big(\,\xi_0^{(i)} + \tfrac{\alpha_t}{\delta_i}(\,v^{(i)} - \xi_0^{(i)}\,),\, \eta^2\sigma_t^2\big), \quad 0 < \sigma_t < \delta_i, \tag{14}$$

$$q^{(t)}\big(\xi_t^{(i)} \big|\, \cdot\,\big) = \mathcal{N}\big(\,(1-\eta_b)\xi_0^{(i)} + \eta_b v^{(i)},\, \sigma_t^2 - \eta_b^2\delta_i^2\big), \quad \sigma_t \geq \delta_i. \tag{15}$$

When a coordinate is unobserved ($\tilde{s}_i{=}0$), the step reduces to unconditional generation. When observed and diffusion noise dominates ($\sigma_t < \delta_i$), the update pulls toward the measurement with gain α_t/δ_i. Once measurement noise dominates ($\sigma_t \geq \delta_i$), the mean becomes a convex blend with weight η_b and the variance reduces by $\eta_b^2\delta_i^2$ to preserve Gaussian marginals.

Transitions ($t < T$): sampling distribution p_θ. Let $\mathbf{x}_{\theta,t} = f_\theta(\mathbf{x}_{t+1}, t{+}1)$ and $\xi_{\theta,t}^{(i)} = (\tilde{V}^\top \mathbf{x}_{\theta,t})^{(i)}$. The sampling updates mirror the variational ones with $\xi_0^{(i)}$ replaced by $\xi_{\theta,t}^{(i)}$:

$$p_\theta^{(t)}\big(\xi_t^{(i)} \big|\, \cdot\,\big) = \mathcal{N}\big(\,\xi_{\theta,t}^{(i)} + \beta_t(\xi_{t+1}^{(i)} - \xi_{\theta,t}^{(i)}),\, \eta^2\sigma_t^2\big), \quad \tilde{s}_i = 0, \tag{16}$$

$$p_\theta^{(t)}\big(\xi_t^{(i)} \big|\, \cdot\,\big) = \mathcal{N}\big(\,\xi_{\theta,t}^{(i)} + \tfrac{\alpha_t}{\delta_i}(\,v^{(i)} - \xi_{\theta,t}^{(i)}\,),\, \eta^2\sigma_t^2\big), \quad 0 < \sigma_t < \delta_i, \tag{17}$$

$$p_\theta^{(t)}\big(\xi_t^{(i)} \big|\, \cdot\,\big) = \mathcal{N}\big(\,(1-\eta_b)\xi_{\theta,t}^{(i)} + \eta_b v^{(i)},\, \sigma_t^2 - \eta_b^2\delta_i^2\big), \quad \sigma_t \geq \delta_i. \tag{18}$$

These are DDRM's per-coordinate sampling rules in whitened coordinates, with substitutions $(s_i, \bar{y}^{(i)}) \mapsto (\tilde{s}_i, v^{(i)})$ and threshold σ_y/s_i replaced by $\delta_i = \sigma_y/\tilde{s}_i$.

Interpretation of boundary conditions. After whitening, every coordinate has unit noise variance, but the forward operator is rescaled by $\Sigma^{-1/2}$. As a result, the transition between the "diffusion-dominated" and the "measurement-dominated" conditions occurs when the diffusion noise level σ_t matches the effective measurement noise along each singular direction. This happens at $\sigma_t = \delta_i$, where δ_i is the standard deviation of the measurement noise in the whitened basis. Because whitening changes the singular values from s_i to $\tilde{s}_i$, the boundary is determined by $\tilde{s}_i$, not the original s_i.

3. CIN-D Capture Protocol

We collect the *Correlated Image Noise Dataset* (CIN-D) using a rolling-shutter machine-vision camera. The camera is configured to output 16-bit Bayer frames and, for convenience, 16-bit Linear RGB images. To reduce storage requirements and simplify subsequent noise modeling, we configure the camera's region of interest (ROI) to capture only a 256×256 center crop of the full-resolution sensor. This size matches the input resolution of the DDRM backbone, avoiding the need for resizing operations that would corrupt the natural noise structure. All gamma correction and automatic image-processing features (e.g., auto-exposure, auto-gain, auto-white-balance) are explicitly disabled to ensure a linear and stationary acquisition pipeline. We show representative examples in Fig. 1 illustrating the diversity and the degradation seen in the same scenes due to higher correlated noise levels.

For each scene, we sequentially capture four images by switching between the noise levels summarized in Table 2. The gain and exposure combinations are carefully selected through trial and error to maintain approximately constant effective exposure, and hence overall scene brightness, across all four noise levels. This design keeps all acquisition variables fixed except for the noise-level control parameters, which is essential for studying denoising

Figure 1. **CIN-D dataset examples.** CIN-D contains images of the same scenes captured at different noise levels using a FLIR Blackfly S BFS-U3-63S4C-C color camera. Noise levels decrease from top to bottom through controlled variations in sensor gain and exposure time, while scene content and viewing geometry remain fixed. This illustrates the progressive degradation in image quality that our dataset captures for benchmarking restoration algorithms under realistic correlated noise conditions.

Table 2. **Gain and exposure settings for CIN-D noise levels.** Higher gain and lower exposure produce stronger noise.

Noise level ℓ	Gain g_ℓ (dB)	Exposure t_ℓ (ms)
Zero	0.0	350
Low	25.0	20
Medium	35.0	8
High	43.0	2.5

and restoration algorithms that assume identical underlying scene radiance. In the remainder of this section, we briefly describe the hardware setup (Table 3) and the acquisition strategies for indoor and outdoor capture.

Table 3. **Hardware components used for CIN-D acquisition.** The setup includes a rolling-shutter machine-vision camera, a fixed focal-length lens, and a mounting system for stable capture.

Component	Model / Part no.	Key specifications
Camera	FLIR Blackfly S BFS-U3-63S4C-C	6.3 MP color CMOS, rolling shutter
Lens	Edmund Optics TECHSPEC HP, #86-572	25 mm, $f/1.8$–$f/22$, C-mount
Mounting (indoor)	Post Mounts	Fixed camera, repeatable alignment

Hardware setup and acquisition strategy We acquire all images in CIN-D using a FLIR Blackfly S BFS-U3-63S4C-C color rolling-shutter camera coupled with an Ed-

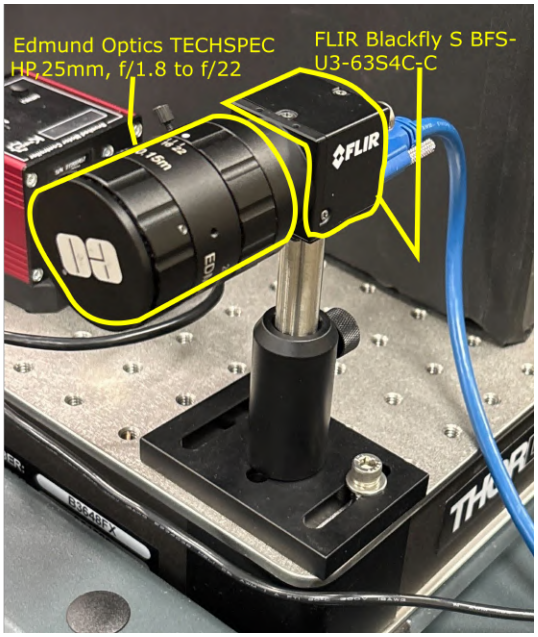

Figure 2. **Camera and lens assembly used for CIN-D acquisition.** A FLIR Blackfly S BFS-U3-63S4C-C camera is paired with a fixed-focal-length C-mount lens and rigidly mounted on an optical bench for indoor scene capture.

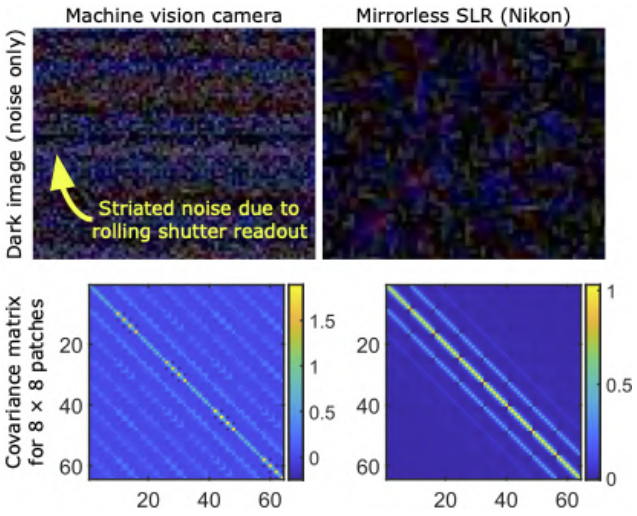

Figure 3. **Comparison of dark-frame noise in a machine-vision rolling-shutter camera and a mirrorless SLR.** Top row: Dark frames showing pronounced horizontal striations in the machine-vision camera due to rolling-shutter readout, versus more homogeneous noise in the Nikon mirrorless camera. Bottom row: Noise covariance matrices of 8×8 luminance patches, showing anisotropic correlations for the rolling-shutter sensor, compared to a weakly correlated noise profile for the mirrorless SLR.

mund Optics TECHSPEC HP 25 mm C-mount lens as shown in Fig. 2. This lens delivers well-corrected, low-distortion imaging with an adjustable aperture that allows us to control optical throughput and prevent saturation under different lighting conditions. For indoor captures, we mount the camera rigidly on an optical bench to maintain consistent viewing geometry while varying only the target and illumination across scenes. Indoor targets remain static under approximately uniform room lighting or controlled illumination, and we adjust the lens f-number as necessary to keep sensor response within a non-saturated, signal-dominated regime at fixed gain and exposure settings. For outdoor captures, we follow the same principle of imaging static content while ensuring the camera remains completely stationary throughout acquisition by securing it firmly to a rigid surface using a tripod laptop stand and duct tape. Under both indoor and outdoor conditions, we carefully avoid moving objects, rapid illumination changes, and strong specular highlights so that dataset variation is dominated by our controlled changes in exposure, gain, and noise statistics rather than by scene dynamics. We sequentially vary the noise level using a custom acquisition script built on the PySpin (Spinnaker) SDK to access and process camera data programmatically. This approach is essential for benchmarking denoising and restoration methods under identical scene radiance and viewing geometry, while introducing progressively more challenging noise conditions. The 0 dB gain frame serves as a substitute ground-truth clean image, enabling quantitative evaluation of denoising performance on the corresponding noisy images.

Dark frame acquisition We capture dark frames to characterize sensor noise and estimate the per-noise-level spatial covariance matrix. The process mirrors the actual acquisition strategy but with the lens cap in place to block all light from reaching the sensor. This procedure is performed once per device (optionally per gain/exposure) and the resulting whitening transform is reused for all subsequent restorations. Dark frames use exactly the same (t_ℓ, g_ℓ) settings as specified in Table 2 and are stored in parallel directory trees using the same logging format. As shown in Fig. 3, computing the noise covariance matrix allows us to quantify the strength, spatial extent, and anisotropy of correlated noise at each level, which informs the construction of our whitening transform.

Estimation of noise covariance matrix. We estimate the covariance of the spatially correlated sensor noise using dark frames captured with the lens cap on at the same exposure and gain settings as the scene images. We model the dark-frame signal as a zero-mean random process that is approximately locally stationary over small patches. We first normalize intensities to $[0, 1]$ and convert the image to a single-channel linear luminance representation L to obtain a scalar noise representation, where $L = 0.2126R +$

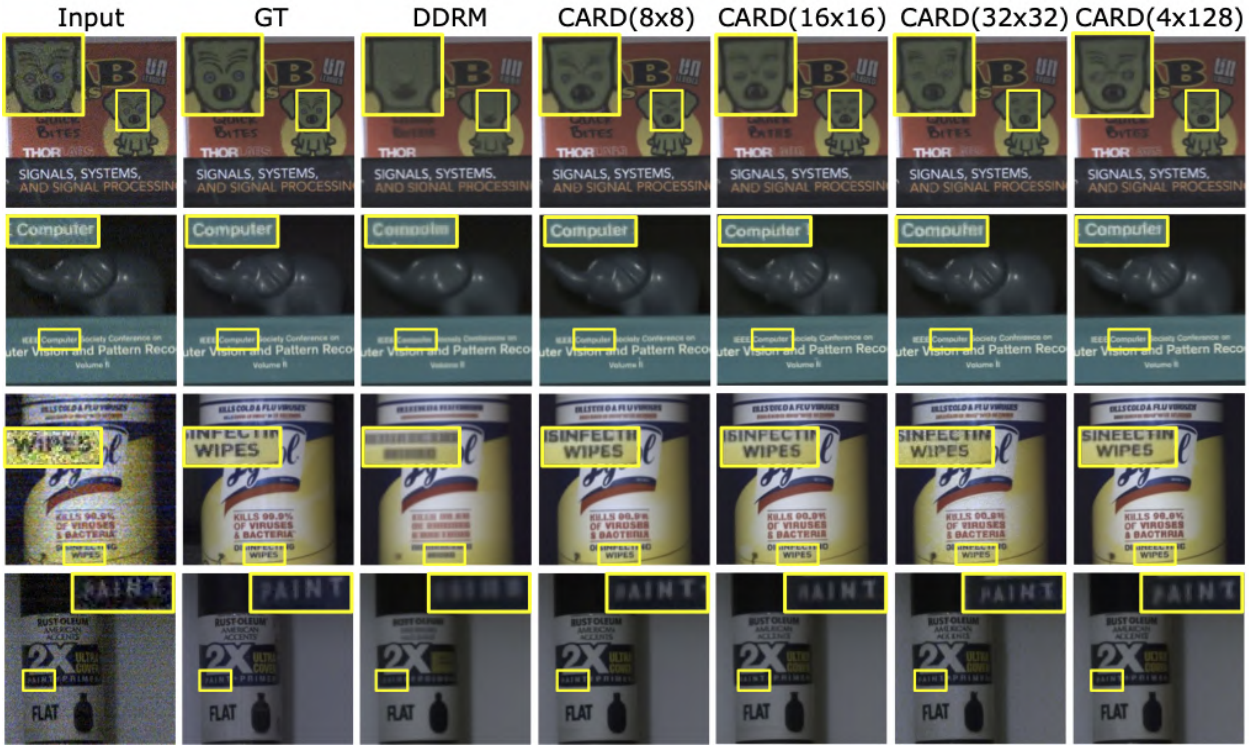

Figure 4. **Effect of patch size on CARD denoising quality.** Top two rows show low-noise images where smaller patches (8×8, 16×16) produce slightly clearer outputs. Bottom two rows show high-noise images where larger and anisotropic patches (32×32, 4×128) yield improved visual quality by better capturing spatial noise correlations.

Table 4. **Patch size effect on denoising CIN-D indoor images using CARD.** Values are PSNR/SSIM/LPIPS (P/S/L). Each patch size corresponds to the local neighborhood used for covariance estimation and whitening.

Patch Size	Low Noise (P↑/S↑/L↓)	High Noise (P↑/S↑/L↓)
8×8	39.8 / 0.96 / 0.17	33.1 / 0.87 / 0.29
16×16	39.8 / 0.96 / 0.17	33.2 / 0.87 / 0.27
32×32	39.2 / 0.94 / 0.15	33.3 / 0.88 / 0.27
4×128	39.8 / 0.96 / 0.16	33.4 / 0.88 / 0.25

$0.7152G + 0.0722B$. We partition the luminance image into non-overlapping 8×8 patches and vectorize each patch as $\mathbf{z} \in \mathbb{R}^{64}$. We aggregate all patches from all dark-frames to yield N samples, which we stack to form the data matrix $Z = [\mathbf{z}_1, \ldots, \mathbf{z}_N] \in \mathbb{R}^{64 \times N}$. After we subtract the per-pixel mean from each row of Z, we estimate the sample covariance $\Sigma \in \mathbb{R}^{64 \times 64}$. The noise covariance matrix Σ captures spatial correlations between pixels within local neighborhoods. We use this estimated covariance for the whitening steps in CARD when evaluating on real data.

4. Patch Size for Covariance Estimation

The patch size used for covariance estimation has a modest yet visible effect on denoising quality, as shown in Table 4 and Figure 4. At low noise levels, all patch sizes achieve comparable PSNR (∼39.8 dB) and SSIM (∼0.96), indicating that CARD remains stable across spatial scales. However, perceptual quality varies slightly, with the 32×32 patch achieving the lowest LPIPS (0.15). Qualitatively, as seen in the top two rows of Figure 4, smaller patch sizes such as 8×8 and 16×16 produce slightly clearer outputs at low noise, likely because local noise statistics are well-captured within compact neighborhoods when noise is mild.

At high noise levels, the elongated 4×128 patch attains the highest PSNR (33.4 dB) and lowest LPIPS (0.25) while maintaining high SSIM (0.88), suggesting it better captures the horizontal correlations characteristic of rolling-shutter readout. As illustrated in the bottom two rows of Figure 4, larger patches yield improved visual quality under severe noise, presumably because they aggregate sufficient samples to reliably estimate the stronger spatial correlations present at high gain settings. CARD proves robust to patch size choice, with 8×8 patches offering a good balance between computational efficiency and performance.

Figure 5. **Effect of patch whitening.** Patch whitening only affects the intermediate whitened input, not the final reconstructed output.

4.1. Patch boundary artifacts

Patch processing in CARD is used only for whitening and local data-consistency preconditioning, while the diffusion denoiser operates on the full image at every step. This design ensures that global image priors enforced by the diffusion model maintain spatial coherence across patch boundaries, preventing visible seams or blocking artifacts. In practice, patchwise operations affect only the intermediate whitened representation and not the final reconstruction. Figure 5 illustrates this behavior by comparing the whitened image, where patch boundaries are visible, with the final reconstructed result, where no boundary artifacts remain.

5. Additional Experiments

5.1. Experiments with simulated correlated noise

We evaluate CARD on denoising, deblurring, and super-resolution tasks across ImageNet [3], LSUN-Bed, and LSUN-Cat [4] datasets under simulated correlated noise.

Denoising results. We evaluate CARD on image denoising under simulated correlated noise across three datasets: ImageNet, LSUN-Bed, and LSUN-Cat. The quantitative results are summarized in Tables 5, 6, and 7 for noise levels $\sigma_0 \in \{0.1, 0.5, 0.9\}$. Across all datasets and noise levels, CARD achieves the highest PSNR and SSIM, while also obtaining the lowest LPIPS. Learning-based denoisers trained under the i.i.d. noise assumption (Restormer, DnCNN, Noise2Info, PCST) experience substantial performance degradation as noise becomes more correlated or more severe. Classical and diffusion-based i.i.d. methods (BM3D, DDNM, DDRM) remain competitive at low noise levels but deteriorate rapidly for large σ_0. In contrast, CARD continues to outperform all baselines by a significant margin. Learning-based models designed specifically for correlated noise (APRRD-BSN, APRRD-NBSN, AP-BSN, LG-BPN) offer more robustness than i.i.d. baselines, yet they consistently fall short of CARD on all metrics.

Deblurring results. We evaluate the effectiveness of CARD under simulated correlated noise across three deblurring settings: Gaussian, Uniform, and Anisotropic blur. Figure 6 presents qualitative comparisons on ImageNet, LSUN-Cat, and LSUN-Bed at two noise levels (σ_0=0.2 and 0.5). Across all datasets and blur types, CARD produces sharper reconstructions and preserves fine structures more effectively than existing diffusion-based and learning-based approaches. Tables 8, 9, and 10 report quantitative results. CARD achieves the highest PSNR and SSIM, while also obtaining the lowest LPIPS across all combinations of blur type, dataset, and noise level.

Super-resolution results. We evaluate CARD for $2\times$ and $4\times$ super-resolution under medium and high correlated noise. Tables 11, 12, and 13 summarize results on ImageNet, LSUN-Bed, and LSUN-Cat. Across all datasets, scales, and noise levels, CARD consistently achieves the highest PSNR and SSIM, while obtaining the lowest LPIPS. The gains are particularly significant at higher noise levels and for the more challenging $4\times$ setting, where other methods struggle to maintain stability.

5.2. Experiments on CIN-D

We evaluate CARD on our dataset CIN-D, which contains indoor and outdoor scenes captured across three noise levels defined by camera gain and exposure.

Denoising results. Table 14 summarizes the quantitative results. CARD achieves the best PSNR, SSIM, and LPIPS across all noise levels and environments. The improvements are especially significant at medium and high noise, where existing learning-based and diffusion-based methods struggle to preserve structure under strong correlation artifacts. Figure 7 presents qualitative comparisons. Competing methods either over-smooth fine structures or leave noise artifacts behind, particularly in textured regions and low-light areas. CARD suppresses noise more effectively while retaining high-frequency detail such as textures.

Deblurring results. We evaluate CARD on deblurring using the forward operators described in Section 1.2. Table 15 shows results at the low-noise setting for both indoor and outdoor scenes. CARD achieves the highest PSNR and SSIM and the lowest LPIPS across all blur types and scene categories. Competing diffusion-based methods such as DDRM and DDNM recover overall structure but introduce over-smoothing under correlated noise, reflected in higher LPIPS and lower SSIM. Figure 8 shows qualitative comparisons. CARD removes noise while preserving fine details such as text and patterns, whereas other methods either over-smooth or leave structured residuals.

Super-resolution results. We evaluate CARD on $2\times$ and $4\times$ super-resolution using the CIN-D dataset. The for-

Table 5. **Denoising results on ImageNet** across low (0.1), medium (0.5), and high (0.9) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance across all noise levels.

Category	Model	Noise Level σ_0 0.1(P↑/S↑/L↓)	0.5(P↑/S↑/L↓)	0.9(P↑/S↑/L↓)
Learning-based (i.i.d.)	Restormer [2]	<u>31.3</u>/<u>0.93</u>/<u>0.10</u>	23.9/0.74/0.28	18.5/0.52/0.59
	DnCNN [5]	22.6/0.87/0.24	20.4/0.61/0.46	18.1/0.40/0.66
	Noise2Info [6]	25.2/0.80/0.24	24.6/0.74/<u>0.26</u>	<u>23.3</u>/0.65/<u>0.29</u>
	PCST [7]	25.6/0.71/0.36	19.8/0.48/0.63	16.6/0.36/0.79
Non-learning (i.i.d.)	BM3D [8]	30.1/0.78/0.11	<u>25.8</u>/0.71/0.33	22.4/0.58/0.50
	DDNM [9]	28.1/0.83/0.25	16.3/0.24/0.57	12.3/0.10/0.68
	DDRM [1]	31.0/0.92/0.14	24.8/<u>0.79</u>/0.33	22.7/<u>0.74</u>/0.40
Learning-based (correlated)	APRRD-BSN [10]	23.9/0.72/0.25	22.0/0.60/0.38	20.1/0.52/0.51
	APRRD-NBSN [10]	24.9/0.76/0.31	22.4/0.63/0.45	20.5/0.54/0.54
	AP-BSN [11]	22.3/0.54/0.25	20.7/0.38/0.60	19.2/0.28/0.80
	LG-BPN [12]	27.5/0.79/0.25	23.8/0.67/0.36	20.3/0.57/0.49
Ours	**CARD**	**34.0/0.96/0.07**	**29.1/0.90/0.15**	**26.7/0.86/0.22**

Table 6. **Denoising results on LSUN-Bed** across low (0.1), medium (0.5), and high (0.9) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance across all noise levels.

Category	Model	Noise Level σ_0 0.1(P↑/S↑/L↓)	0.5(P↑/S↑/L↓)	0.9(P↑/S↑/L↓)
Learning-based (i.i.d.)	Restormer [2]	<u>33.8</u>/0.93/<u>0.08</u>	23.7/0.75/0.32	18.6/0.56/0.61
	DnCNN [5]	23.2/0.90/0.20	21.1/0.63/0.45	18.7/0.41/0.67
	Noise2Info [6]	26.3/0.83/0.21	25.7/0.78/<u>0.23</u>	<u>24.1</u>/0.69/<u>0.29</u>
	PCST [7]	25.8/0.76/0.33	20.2/0.53/0.65	17.0/0.40/0.81
Non-learning (i.i.d.)	BM3D [8]	30.1/0.74/0.14	<u>26.1</u>/0.73/0.31	22.5/0.61/0.50
	DDNM [9]	28.5/0.83/0.23	16.3/0.24/0.55	12.2/0.10/0.67
	DDRM [1]	32.2/<u>0.95</u>/0.10	26.0/<u>0.85</u>/0.25	23.9/<u>0.81</u>/0.30
Learning-based (correlated)	APRRD-BSN [10]	24.5/0.75/0.27	22.5/0.65/0.40	20.8/0.57/0.53
	APRRD-NBSN [10]	25.8/0.79/0.31	23.1/0.67/0.47	21.2/0.59/0.58
	AP-BSN [11]	29.3/0.86/0.17	25.5/0.74/0.29	23.0/0.56/0.52
	LG-BPN [12]	27.4/0.81/0.24	24.1/0.71/0.38	20.9/0.61/0.52
Ours	**CARD**	**35.1/0.97/0.05**	**29.6/0.92/0.12**	**27.2/0.89/0.17**

ward operators follow the standard degradation models described in Section 1.2. Table 16 reports quantitative results at the low-noise setting for both indoor and outdoor scenes. CARD achieves the best performance across all configurations, delivering higher PSNR and SSIM as well as significantly lower LPIPS for both scales. The gains are more significant at $4\times$ upsampling, where correlated noise and aliasing artifacts make the reconstruction particularly challenging. Figure 9 shows qualitative comparisons. DiffIR and DDRM recover structure but either over-smooth textures or leave noise artifacts, whereas CARD preserves both sharpness and fine detail.

Table 7. **Denoising results on LSUN-Cat** across low (0.1), medium (0.5), and high (0.9) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance across all noise levels.

Category	Model	Noise Level σ_0 0.1(P↑/S↑/L↓)	0.5(P↑/S↑/L↓)	0.9(P↑/S↑/L↓)
Learning-based (i.i.d.)	Restormer [2]	<u>33.6</u>/0.92/<u>0.11</u>	23.9/0.75/0.30	18.6/0.55/0.58
	DnCNN [5]	24.0/0.90/0.20	21.3/0.61/0.45	18.5/0.39/0.67
	Noise2Info [6]	25.5/0.81/0.25	25.0/0.76/<u>0.26</u>	23.7/0.68/<u>0.29</u>
	PCST [7]	26.7/0.75/0.35	20.2/0.52/0.61	16.8/0.39/0.77
Non-learning (i.i.d.)	BM3D [8]	30.1/0.76/0.14	<u>26.1</u>/0.72/0.34	22.4/0.61/0.51
	DDNM [9]	28.3/0.82/0.27	16.5/0.24/0.57	12.4/0.10/0.68
	DDRM [1]	31.6/<u>0.93</u>/0.15	<u>26.1</u>/<u>0.83</u>/0.30	<u>24.1</u>/<u>0.80</u>/0.35
Learning-based (correlated)	APRRD-BSN [10]	24.2/0.76/0.25	22.7/0.65/0.35	20.9/0.57/0.47
	APRRD-NBSN [10]	25.5/0.80/0.31	23.1/0.67/0.44	21.2/0.59/0.52
	AP-BSN [11]	29.6/0.84/0.22	25.9/0.73/0.33	23.3/0.64/0.41
	LG-BPN [12]	28.3/0.82/0.25	24.3/0.71/0.36	20.6/0.61/0.47
Ours	**CARD**	**34.3/0.96/0.07**	**29.6/0.91/0.16**	**27.3/0.88/0.22**

Table 8. **Deblurring results on ImageNet** across medium (0.2) and high (0.5) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. Entries marked "–" indicate the method does not support the corresponding task. CARD achieves the best performance across all noise levels.

Model	Gaussian Deblur 0.2 (P↑/S↑/L↓)	Gaussian Deblur 0.5 (P↑/S↑/L↓)	Anisotropic Deblur 0.2 (P↑/S↑/L↓)	Anisotropic Deblur 0.5 (P↑/S↑/L↓)	Uniform Deblur 0.2 (P↑/S↑/L↓)	Uniform Deblur 0.5 (P↑/S↑/L↓)
Restormer [2]	19.2/0.39/0.76	17.9/0.20/1.01	20.9/0.37/0.62	18.1/0.17/1.03	19.3/0.29/0.71	17.1/0.12/1.12
DPS [13]	19.4/0.53/0.41	19.3/0.51/0.46	–	–	–	–
DiffIR [14]	20.1/0.43/0.65	17.8/0.19/0.98	11.0/0.19/0.92	10.7/0.08/1.10	20.7/0.37/0.80	18.2/0.15/1.05
DDNM [9]	21.8/0.53/0.45	15.9/0.21/0.60	–	–	21.2/0.50/0.49	15.8/0.20/0.62
DDRM [1]	<u>24.3</u>/<u>0.77</u>/<u>0.36</u>	<u>22.2</u>/<u>0.72</u>/<u>0.42</u>	<u>24.1</u>/<u>0.76</u>/<u>0.37</u>	<u>22.4</u>/<u>0.72</u>/<u>0.42</u>	<u>22.9</u>/<u>0.72</u>/<u>0.41</u>	<u>21.4</u>/<u>0.69</u>/<u>0.46</u>
CARD (ours)	**26.6/0.84/0.23**	**24.4/0.79/0.30**	**25.9/0.82/0.24**	**23.9/0.77/0.30**	**24.1/0.77/0.29**	**22.3/0.72/0.33**

Table 9. **Deblurring results on LSUN-Bed** across medium (0.2) and high (0.5) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. Entries marked "–" indicate the method does not support the corresponding task. CARD achieves the best performance across all noise levels.

Model	Gaussian Deblur 0.2 (P↑/S↑/L↓)	Gaussian Deblur 0.5 (P↑/S↑/L↓)	Anisotropic Deblur 0.2 (P↑/S↑/L↓)	Anisotropic Deblur 0.5 (P↑/S↑/L↓)	Uniform Deblur 0.2 (P↑/S↑/L↓)	Uniform Deblur 0.5 (P↑/S↑/L↓)
Restormer [2]	19.4/0.38/0.84	18.1/0.19/1.08	21.3/0.38/0.69	18.3/0.16/1.11	19.8/0.31/0.76	17.4/0.12/1.19
DPS [13]	19.7/0.58/0.40	19.3/0.56/<u>0.46</u>	–	–	–	–
DiffIR [14]	20.5/0.43/0.72	18.1/0.18/1.05	22.6/0.44/0.71	19.3/0.19/1.01	20.9/0.38/0.85	18.4/0.15/1.11
DDNM [9]	22.0/0.54/0.42	15.9/0.21/0.58	–	–	21.5/0.51/0.45	15.9/0.20/0.59
DDRM [1]	<u>25.1</u>/<u>0.82</u>/<u>0.28</u>	<u>23.0</u>/<u>0.78</u>/**0.34**	<u>24.8</u>/<u>0.81</u>/<u>0.29</u>	<u>23.1</u>/<u>0.78</u>/<u>0.33</u>	<u>23.7</u>/<u>0.78</u>/<u>0.32</u>	<u>22.2</u>/<u>0.75</u>/<u>0.36</u>
CARD (ours)	**26.6/0.86/0.19**	**24.5/0.81/0.34**	**25.9/0.82/0.24**	**24.0/0.80/0.25**	**24.4/0.80/0.24**	**22.6/0.76/0.29**

Table 10. **Deblurring results on LSUN-Cat** across medium (0.2) and high (0.5) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. Entries marked "–" indicate the method does not support the corresponding task. CARD achieves the best performance across all noise levels.

Model	Gaussian Deblur		Anisotropic Deblur		Uniform Deblur	
	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)
Restormer [2]	18.8/0.40/0.82	17.6/0.20/1.06	21.0/0.39/0.66	18.2/0.17/1.08	19.3/0.52/0.48	17.1/0.13/1.16
DPS [13]	19.1/0.57/0.40	18.9/0.54/0.44	–	–	–	–
DiffIR [14]	19.8/0.44/0.68	17.7/0.19/1.02	22.5/0.46/0.67	19.2/0.21/0.98	20.6/0.39/0.82	18.2/0.16/1.08
DDNM [9]	22.3/0.55/0.45	16.2/0.22/0.60	–	–	21.8/0.52/0.48	16.0/0.21/0.61
DDRM [1]	25.5/0.82/0.32	23.2/0.77/0.38	25.2/0.81/0.33	23.3/0.77/0.38	24.0/0.78/0.37	22.4/0.75/0.41
CARD (ours)	**27.6/0.87/0.22**	**25.2/0.82/0.28**	**26.8/0.85/0.23**	**24.7/0.81/0.28**	**25.1/0.81/0.26**	**23.2/0.77/0.31**

Table 11. **Super-resolution results on ImageNet** across $2\times$ and $4\times$ upscaling, medium (0.2) and high (0.5) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance for both $2\times$ and $4\times$ upscaling, and at both noise levels.

Model	SR2		SR4	
	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)
DiffIR [14]	25.7/0.71/0.34	23.9/0.65/0.42	23.8/0.63/0.47	23.0/0.53/0.60
DDRM [1]	25.7/0.81/0.32	22.7/0.74/0.41	22.9/0.74/0.40	20.4/0.69/0.46
CARD (ours)	**28.0/0.86/0.20**	**25.1/0.77/0.33**	**25.0/0.78/0.30**	**23.1/0.71/0.40**

Table 12. **Super-resolution results on LSUN-Bed** across $2\times$ and $4\times$ upscaling, medium (0.2) and high (0.5) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance for both $2\times$ and $4\times$ upscaling, and at both noise levels.

Model	SR2		SR4	
	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)
DiffIR [14]	26.8/0.72/0.32	23.6/0.47/0.62	24.5/0.67/0.40	23.7/0.56/0.56
DDRM [1]	26.1/0.85/0.24	23.5/0.80/0.32	23.1/0.78/0.31	20.9/0.74/0.37
CARD (ours)	**28.9/0.90/0.14**	**27.0/0.86/0.22**	**25.9/0.83/0.24**	**25.0/0.81/0.29**

Table 13. **Super-resolution results on LSUN-Cat** across $2\times$ and $4\times$ upscaling, medium (0.2) and high (0.5) noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance for both $2\times$ and $4\times$ upscaling, and at both noise levels.

Model	SR2		SR4	
	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)	0.2 (P↑/S↑/L↓)	0.5 (P↑/S↑/L↓)
DiffIR [14]	27.6/0.75/0.34	24.0/0.50/0.64	25.4/0.69/0.48	24.3/0.58/0.63
DDRM [1]	26.6/0.85/0.30	23.7/0.79/0.37	23.7/0.79/0.36	21.1/0.75/0.41
CARD (ours)	**29.1/0.88/0.19**	**26.9/0.83/0.27**	**26.5/0.83/0.27**	**25.2/0.80/0.33**

Figure 6. **Qualitative comparison of Gaussian, Uniform, and Anisotropic deblurring** at varying noise levels ($\sigma_0 = 0.2, 0.5$) on ImageNet (top three rows), LSUN-Cat (middle three rows), and LSUN-Bedroom (bottom three rows). The boxed regions highlight areas of interest where differences are most apparent. CARD outperforms other baselines while preserving fine details.

Table 14. **Denoising results on our dataset (CIN-D)**, evaluated at low, medium, and high noise levels. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD achieves the best performance at all noise levels.

Model	Indoor Scenes			Outdoor Scenes		
	Low (P↑/S↑/L↓)	Med (P↑/S↑/L↓)	High (P↑/S↑/L↓)	Low (P↑/S↑/L↓)	Med (P↑/S↑/L↓)	High (P↑/S↑/L↓)
DnCNN [5]	29.4/0.86/0.29	24.5/0.66/0.49	25.0/0.47/0.69	27.9/0.86/0.29	24.7/0.70/0.45	24.3/0.50/0.65
Noise2Info [6]	12.8/0.65/0.25	12.8/0.35/0.87	13.5/0.46/0.44	19.7/0.71/0.25	19.8/0.69/0.27	20.4/0.60/0.42
APRRD-BSN [10]	14.1/0.68/0.23	14.3/0.67/0.24	14.8/0.65/0.35	20.6/0.72/0.32	20.5/0.71/0.27	20.6/0.67/0.43
APRRD-NBSN [10]	13.3/0.68/0.26	13.3/0.67/0.28	13.7/0.65/0.34	20.7/0.73/0.30	20.2/0.73/0.28	20.5/0.70/0.42
PCST [7]	38.1/0.93/0.18	27.3/0.67/0.34	29.8/0.80/0.38	36.3/0.91/**0.22**	27.3/0.67/0.35	29.0/0.76/0.42
BM3D [8]	33.7/0.86/0.31	26.8/0.82/0.27	29.1/0.78/0.44	33.7/0.78/0.43	27.9/0.84/0.35	24.2/0.31/0.94
PnP-ADMM [15]	33.9/0.90/0.24	24.4/0.80/0.33	26.1/0.72/0.46	33.1/0.89/0.25	24.7/0.79/0.33	25.6/0.71/0.45
DDNM [9]	34.2/0.93/0.32	25.3/0.86/0.43	24.5/0.60/0.51	33.5/**0.92**/0.35	26.7/0.85/0.31	25.1/0.63/0.50
DDRM [1]	36.7/0.89/0.21	27.7/0.86/0.25	32.3/0.84/0.32	33.9/0.85/0.35	27.6/0.81/0.40	29.4/0.77/0.46
CARD (ours)	**39.8/0.95/0.17**	**27.8/0.88/0.20**	**33.1/0.86/0.28**	**36.4/0.92/0.22**	**28.0/0.87/0.26**	**30.0/0.80/0.40**

Table 15. **Deblurring results on our dataset (CIN-D)** evaluated at low noise level. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. Entries marked "–" indicate the method does not support the corresponding task. CARD consistently outperforms all baselines.

Model	Gaussian Deblur		Anisotropic Deblur		Uniform Deblur	
	Indoor (P↑/S↑/L↓)	Outdoor (P↑/S↑/L↓)	Indoor (P↑/S↑/L↓)	Outdoor (P↑/S↑/L↓)	Indoor (P↑/S↑/L↓)	Outdoor (P↑/S↑/L↓)
DPS [13]	18.0/0.74/0.42	17.2/0.67/0.54	–	–	–	–
DiffIR [14]	19.1/0.81/0.34	18.6/0.76/0.42	25.6/0.87/0.33	25.9/0.82/0.38	21.8/0.82/0.46	21.2/0.77/0.53
Restormer [2]	18.4/0.78/0.30	17.9/0.73/0.39	26.9/**0.88**/0.30	21.7/0.78/0.50	22.3/0.82/0.40	21.7/0.78/0.50
DDNM [9]	33.2/0.87/0.35	32.0/0.81/0.40	–	–	32.6/0.83/0.37	31.4/0.77/**0.42**
DDRM [1]	34.5/0.85/0.31	32.1/0.80/0.44	34.1/0.85/0.32	31.9/0.79/0.44	33.0/0.82/0.37	30.9/0.77/0.48
CARD (ours)	**36.3/0.88/0.23**	**33.9/0.84/0.35**	**35.9/0.88/0.25**	**33.5/0.83/0.37**	**34.9/0.86/0.30**	**32.6/0.81/0.42**

Table 16. **Super-resolution results on our dataset (CIN-D)**, across $2\times$ (SR2) and $4\times$ (SR4) upscaling, evaluated at low noise level. We report PSNR/SSIM/LPIPS as P/S/L. Best and second-best results are marked in bold and underlined, respectively. CARD consistently outperforms all baselines.

Model	SR2		SR4	
	Indoor (P↑/S↑/L↓)	Outdoor (P↑/S↑/L↓)	Indoor (P↑/S↑/L↓)	Outdoor (P↑/S↑/L↓)
DiffIR [14]	37.5/0.92/0.19	35.6/0.91/0.18	35.5/0.89/0.24	33.7/0.87/0.29
DDRM [1]	35.7/0.92/0.28	32.9/0.87/0.41	32.5/0.88/0.39	30.7/0.84/0.49
CARD (ours)	**39.5/0.94/0.14**	**37.4/0.93/0.12**	**37.2/0.91/0.20**	**35.1/0.89/0.22**

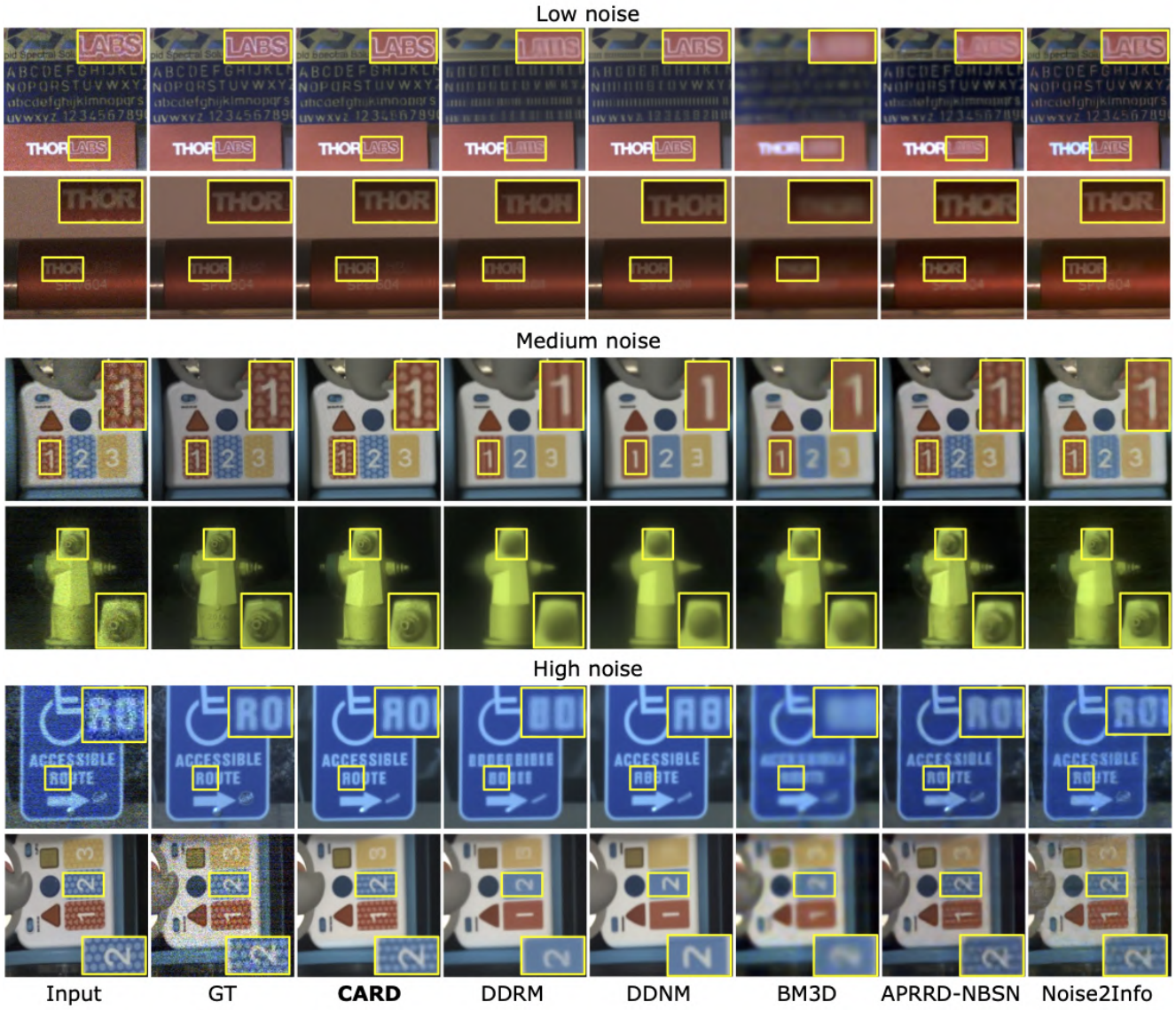

Figure 7. **Qualitative comparison of denoising on our dataset (CIN-D)** at low, medium, and high noise levels. The boxed regions highlight areas of interest where differences are most apparent. CARD outperforms other baselines while preserving fine details.

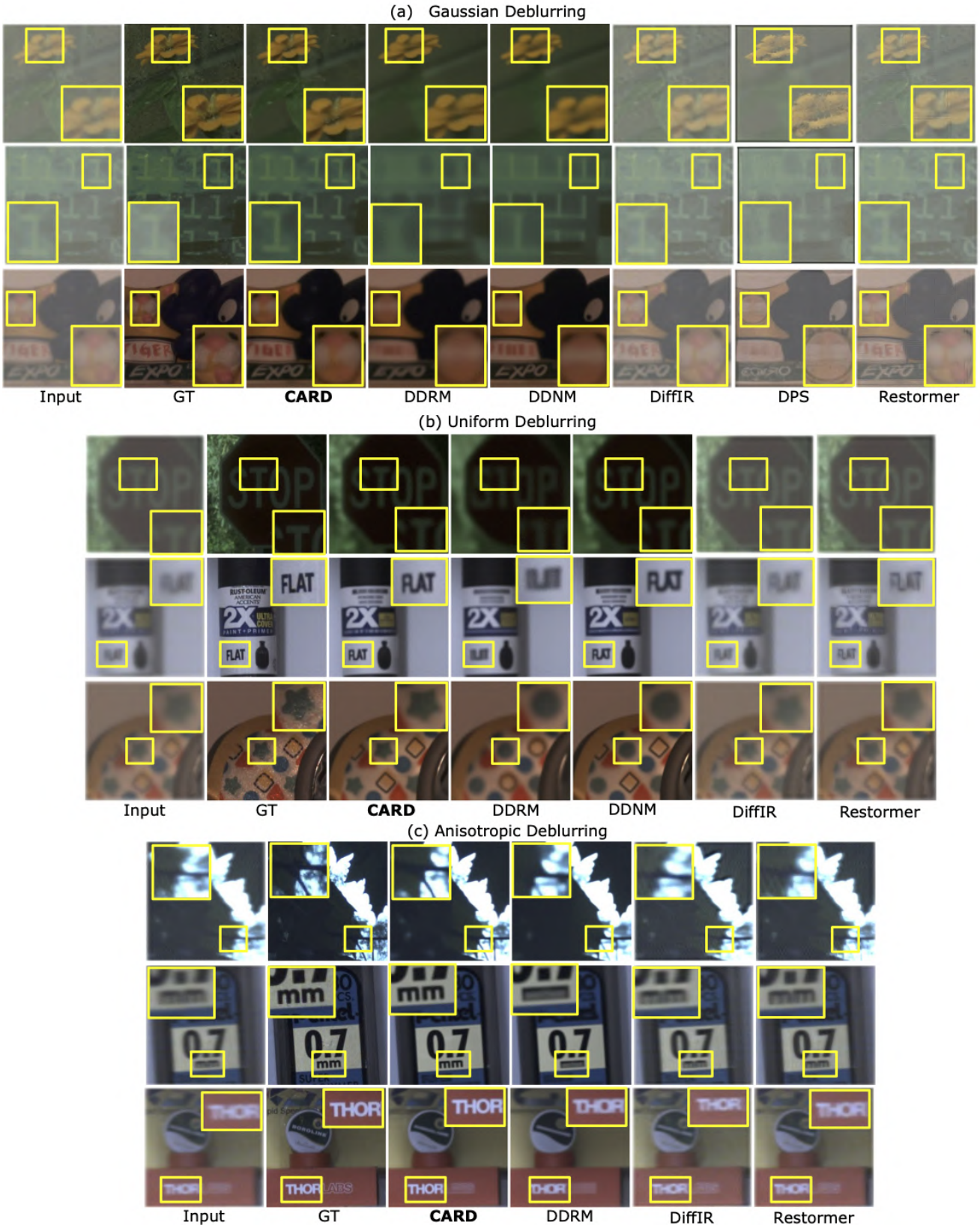

Figure 8. **Qualitative comparison of Gaussian, Uniform, and Anisotropic deblurring on our dataset (CIN-D)** at low noise level. The boxed regions highlight areas of interest where differences are most apparent. CARD outperforms other baselines while preserving fine details.

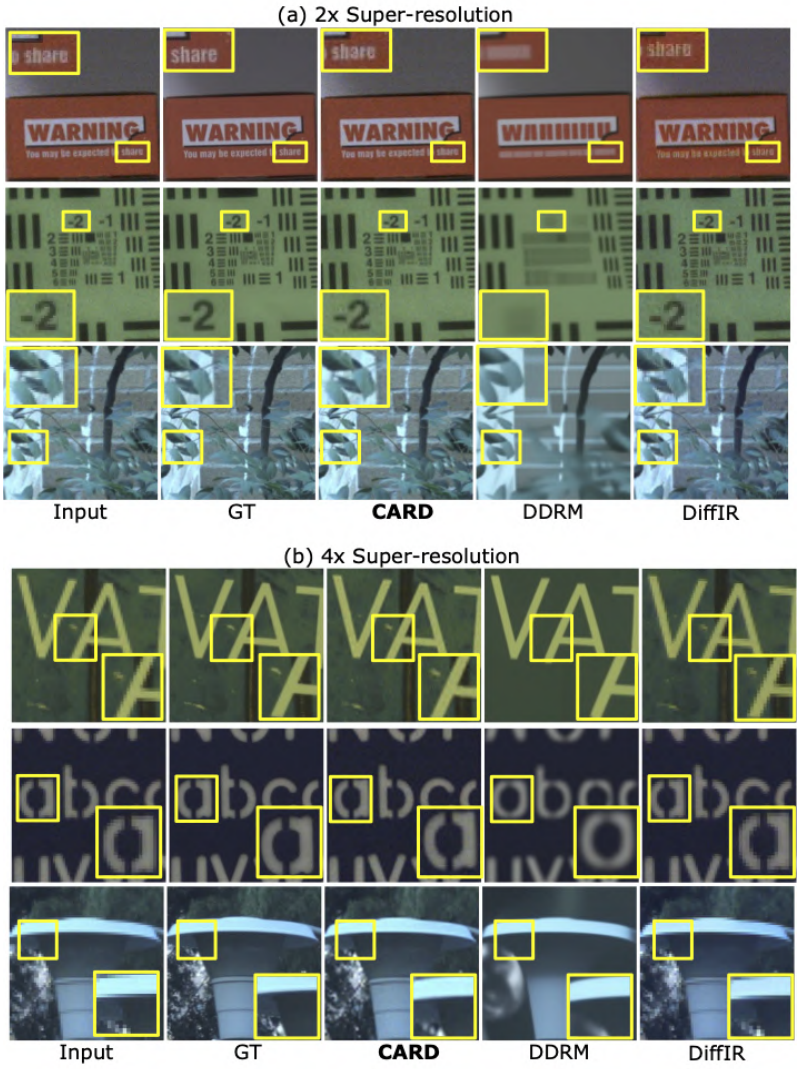

Figure 9. **Qualitative comparison of** $2\times$ **and** $4\times$ **super-resolution on our dataset (CIN-D)** at low noise level. The boxed regions highlight areas of interest where differences are most apparent. CARD outperforms other baselines while preserving fine details.

References

[1] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Adv. Neural Info. Processing Systems*, 2022. 1, 2, 8, 9, 10, 12

[2] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *IEEE Comp. Vision and Pattern Recognition (CVPR)*, 2022. 2, 8, 9, 10, 12

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Comp. Vision and Pattern Recognition (CVPR)*, 2009. 7

[4] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 7

[5] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Trans. Image Processing*, 26(7):3142–3155, 2017. 8, 9, 12

[6] Jiachuan Wang, Shimin Di, Lei Chen, and Charles Wang Wai Ng. Noise2info: Noisy image to information of noise for self-supervised image denoising. In *IEEE Comp. Vision and Pattern Recognition (CVPR)*, 2023. 8, 9, 12

[7] Gregory Vaksman and Michael Elad. Patch-craft self-supervised training for correlated image denoising. In *IEEE Comp. Vision and Pattern Recognition (CVPR)*, 2023. 8, 9, 12

[8] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. Image Processing*, 16(8):2080–2095, 2007. 8, 9, 12

[9] Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *Intl. Conf. Learning Representations*, 2023. 8, 9, 10, 12

[10] Hyunjun Kim and Nam Ik Cho. Apr-rd: Complemental two steps for self-supervised real image denoising. In *Proc. AAAI Conf. Artificial Intelligence (AAAI)*, 2025. 8, 9, 12

[11] Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. Ap-bsn: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *IEEE Comp. Vision and Pattern Recognition (CVPR)*, pages 17725–17734, 2022. 8, 9

[12] Zichun Wang, Ying Fu, Ji Liu, and Yulun Zhang. Lg-bpn: Local and global blind-patch network for self-supervised real-world denoising. In *IEEE Comp. Vision and Pattern Recognition (CVPR)*, pages 18156–18165, 2023. 8, 9

[13] Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *Intl. Conf. Learning Representations*, 2023. 9, 10, 12

[14] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. In *IEEE Intl. Conf. Computer Vision (ICCV)*, 2023. 9, 10, 12

[15] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE global conference on signal and information processing*, pages 945–948. IEEE, 2013. 12