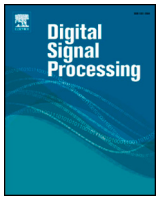
Digit. Signal Process. 159 (2025) 105022

Contents lists available at ScienceDirect

Digital Signal Processing

journal homepage: www.elsevier.com/locate/dsp



Implicit neural representation for image demosaicking

Tomáš Kerepecký [a,b,*], Filip Šroubek [a], Jan Flusser [a]

[a] Institute of Information Theory and Automation, The Czech Academy of Sciences, Pod Vodárenskou věží 4, Prague, CZ-18200, Czechia
[b] Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, Břehová 78/7, Prague, CZ-11519, Czechia

ARTICLE INFO

Keywords:
Demosaicking
Implicit neural representation
Inverse problems

ABSTRACT

We propose a novel approach to enhance image demosaicking algorithms using implicit neural representations (INR). Our method employs a multi-layer perceptron to encode RGB images, combining original Bayer measurements with an initial estimate from existing demosaicking methods to achieve superior reconstructions. A key innovation is the integration of two loss functions: a Bayer loss for fidelity to sensor data and a complementary loss that regularizes reconstruction using interpolated data from the initial estimate. This combination, along with INR's inherent ability to capture fine details, enables high-fidelity reconstructions that incorporate information from both sources. Furthermore, we demonstrate that INR can effectively correct artifacts in state-of-the-art demosaicking methods when input data diverge from the training distribution, such as in cases of noise or blur. This adaptability highlights the transformative potential of INR-based demosaicking, offering a robust solution to this challenging problem.

1. Introduction

Digital camera sensors typically capture raw image data through a Color Filter Array (CFA), resulting in sub-sampled color information that requires reconstruction through a process known as demosaicking. Traditional demosaicking algorithms, such as bilinear interpolation, Malvar [1], and Menon [2], offer computational efficiency but are prone to artifacts like color Moiré, zippering, and false color patterns. These artifacts degrade image quality by introducing undesirable visual effects. Moiré patterns appear as repetitive interference patterns in areas with high-frequency textures. Zippering manifests as jagged edges along sharp transitions. False color patterns distort natural color representation, often due to processing errors during demosaicking.

More advanced approaches have aimed to mitigate these issues by integrating demosaicking with other image processing tasks. For instance, joint demosaicking and denoising or deblurring methods [3–8] employ model-based optimization techniques to achieve better reconstruction quality.

Recent advancements in deep learning have significantly enhanced the performance of demosaicking algorithms [9–15]. These techniques have set new benchmarks by leveraging Convolutional Neural Networks (CNNs) or Transformers to reduce artifacts and improve the fidelity of reconstructed images. However, these methods often struggle when faced with input data that diverge from their training distribution, such as images affected by blur, common in both DSLR and mobile phone cameras (Fig. 1), even when the lens is in focus.

In response to these challenges, we propose a novel deep learning-based approach named INRID (Implicit Neural Representation for Image Demosaicking), which leverages Implicit Neural Representations (INR) [16] to enhance image reconstruction in both traditional and state-of-the-art demosaicking methods. By representing each individual image through the weights of a Multilayer Perceptron (MLP), our approach provides a more flexible and powerful reconstruction.

INRID reconstructs the image by adapting to the specific characteristics of two key inputs: the raw Bayer data and the initial demosaicked image from methods such as Malvar or Menon. A Bayer loss function enforces fidelity to the original raw sensor data, minimizing the mean squared error (MSE) between the reconstructed Bayer pattern and the raw measurements. Simultaneously, the complementary pixel values — those missing in the Bayer pattern — are reconstructed by aligning them with the initial estimate while ensuring consistency with the raw Bayer data. This combined process enables INRID to capture fine image details and correct residual artifacts, that traditional methods often leave unaddressed.

For state-of-the-art deep learning methods, INRID extends beyond refinement to address out-of-distribution scenarios, such as blurred or noisy inputs. By incorporating the forward degradation process—e.g., simulating blur or noise—directly into the optimization, INRID aligns

* Corresponding author.
E-mail address: kerepecky@utia.cas.cz (T. Kerepecký).

https://doi.org/10.1016/j.dsp.2025.105022

Available online 23 January 2025

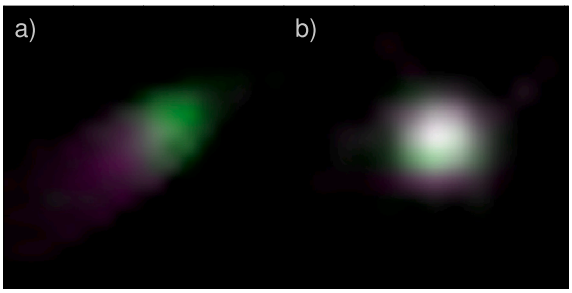


Fig. 1. Intrinsic camera blur (a combination of sensor blur and lens aberrations, present even when the lens is in focus): a) DSLR, b) mobile phone. These intrinsic blur kernels are about 7x7 pixels in size for 16 MPx images. Interpolation was used to magnify the blur kernels for visual presentation.

the reconstruction with both the degraded Bayer data and the initial estimate. This approach ensures robust adaptation to challenging conditions, recovering high-frequency details and reducing artifacts. As a result, INRID significantly enhances demosaicking performance, even when the input data diverge from the training distribution.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 describes the proposed methodology, including the inverse problem and definition of loss functions used for training. Section 4 presents experimental results that demonstrate our approach in enhancing existing methods. Section 5 discusses the limitations of our work and potential future directions for improvement. Finally, Section 6 offers concluding remarks.

2. Related work

2.1. Image demosaicking

Traditional demosaicking methods have predominantly relied on interpolation techniques, which, despite their computational efficiency, are prone to introducing artifacts, especially in regions with high-frequency content. Early methods, such as bilinear interpolation, provided a simple yet effective approach for reconstructing missing color information [17]. The work by Malvar et al. [1] improved upon these techniques by introducing a gradient-corrected bilinear interpolation method, optimized using a Wiener filtering approach, which aimed to reduce the visibility of common artifacts. Menon et al. [2] further advanced the field by incorporating directional filtering and a posteriori decision-making, which improved edge preservation and reduced color artifacts. However, these methods struggled with handling complex textures and often produced noticeable artifacts, such as color Moiré patterns and zipper effects.

Optimization-based methods tackle demosaicking by formulating it as an inverse problem and integrating regularization terms to enhance reconstruction quality. For instance, the multiframe demosaicking and super-resolution method by Farsiu et al. [18] applies a maximum a posteriori (MAP) estimation framework. This approach effectively reduces artifacts and addresses degradations such as noise and blur, while requiring increased computational complexity compared to interpolation-based methods.

The advent of deep learning has led to significant advancements in demosaicking. One notable approach, commonly named DeepDemosaick, is the method proposed by Kokkinos and Lefkimmiatis [10], which introduces a deep convolutional residual network designed to jointly perform demosaicking and denoising. This approach leverages a cascade of convolutional layers to model the underlying patterns in raw sensor data and predict a high-quality full-resolution RGB image. The network is inspired by optimization strategies from classical image regularization methods and is trained end-to-end on a dataset of mosaicked and ground-truth images. This design enables the model to capture complex pixel-level dependencies, resulting in superior color reconstruction and reduced artifacts compared to previous methods.

Another state-of-the-art method, RSTCANet [11], currently a leading method in the field, builds upon the Swin Transformer framework with the introduction of Residual Swin Transformer Channel Attention Blocks. This advanced design captures both spatial and channel-wise dependencies more effectively, thanks to its hierarchical structure and shifted windows, while the residual connections allow for deeper network architectures by mitigating the vanishing gradient problem. RSTCANet excels in preserving fine details and handling complex textures, delivering high-quality demosaicking results across various datasets.

These deep learning-based methods, including RSTCANet and DeepDemosaick, are pre-trained on large datasets to learn a mapping from mosaiced inputs to full-color images. While effective on images similar to the training data, their reliance on pre-training limits their ability to generalize to out-of-distribution data, such as images with blur or noise not represented in the training set. Pre-trained models cannot easily adapt to variations not seen during training, which can lead to suboptimal performance in challenging scenarios.

In contrast, our hybrid approach employs an INR that is optimized individually for each input image. Instead of relying on pre-trained weights, we solve an optimization problem over the network parameters specific to each image, rather than over pixel values as in traditional methods. This per-image optimization allows our model to adapt to the unique characteristics of each image, providing robustness to out-of-distribution data such as noisy or blurred images. By optimizing over network parameters, our method can capture fine image details and correct artifacts more effectively.

2.2. Implicit neural representation

INRs have emerged as a powerful tool in computer vision, representing images and 3D shapes continuously through fully connected feed-forward networks. Early work, such as DeepSDF [19], showcased the effectiveness of ReLU-based MLPs for shape representation.

For images, INR maps spatial coordinates to RGB values using an MLP, enabling continuous image representation, unlike conventional pixel grids. This approach allows high-quality reconstructions, even from sparse or incomplete data.

However, ReLU-based networks, while foundational, struggle to capture fine details, particularly high-frequency information, due to their piecewise linear structure.

To address this limitation, Fourier feature mapping, also known as positional encoding, was introduced [20]. This technique involves mapping the input spatial coordinates into a higher-dimensional space using sinusoidal functions, which helps the MLP capture finer details and improves the reconstruction quality. This approach was popularized by works such as NeRF (Neural Radiance Fields) [21], where it was used to represent 3D scenes with high fidelity.

Building on these advancements, SIREN (Sinusoidal Representation Networks) [22] was introduced, which replaced ReLU with sine activation functions. SIRENs demonstrated the ability to model high-frequency details with greater precision, as sine functions naturally encode oscillatory patterns that are prevalent in image data. This architecture significantly improved the performance of MLPs as image decoders, enabling them to achieve state-of-the-art results in various tasks, including image superresolution and inpainting.

Recently, WIRE (Wavelet Implicit Representations) [23] has pushed the boundaries of INR even further by introducing wavelet-based activation functions. WIRE leverages the multi-resolution properties of wavelets, allowing the MLP to model both coarse and fine details simultaneously.

INCODE [24] further advances INR by introducing a harmonizer network that dynamically adjusts the activation functions based on prior knowledge. This innovation allows INCODE to adaptively fine-tune key parameters like amplitude and frequency of sinusoidal activation func-

tions, enabling the MLP to better capture details and broader signal patterns.

Ramasinghe and Lucey [25] proposed additional activation functions such as Gaussian, Laplacian, and so-called Quadratic to broaden the family of INRs, offering alternatives for capturing fine details without relying on periodic functions.

However, for our demosaicking approach, SIREN and INCODE remain particularly promising due to their sinusoidal activation functions, which are well-suited for interpolating missing data and capturing the complex signal patterns required in this problem.

2.3. Implicit neural representation for image demosaicking

In our previous work, Neural field-based Demosaicking (NERD) [26], we extended the application of INR to the domain of demosaicking. NERD introduced a method that combined ResNet [27] and U-Net [28] architectures to condition the MLP using high-resolution image features extracted from ground-truth images and their corresponding Bayer patterns. This approach demonstrated the potential of INR in handling the challenging task of demosaicking by leveraging the strengths of coordinate based neural networks.

Compared to NERD, the approach presented in this paper significantly reduces computational complexity by eliminating the encoder component while also leveraging the strengths of existing demosaicking methods. Rather than merely introducing a new demosaicking technique, the proposed hybrid framework is designed to substantially enhance reconstruction capabilities and improve the robustness of both traditional and state-of-the-art methods.

3. Problem formulation

In the context of digital image processing, the forward problem involves modeling the degradation process that occurs during image acquisition with a digital camera. This process encompasses blurring due to the camera optical system, subsampling caused by the CFA, commonly implemented as a Bayer pattern, and noise introduced by the sensor. The forward model for Bayer measurement b is expressed as:

$$b = S_B H u + n_B \tag{1}$$

where $u \in \mathbb{R}^M$ represents the vectorized form of the unknown high-resolution sharp image, $H(\cdot) \equiv h * \cdot$ denotes the channel-dependent blurring operator, where h is the Point Spread Function (PSF) estimated from calibration data and $*$ indicates convolution. $n_B \approx \mathcal{N}(0, \sigma_B^2)$ represents additive white Gaussian noise with zero mean and variance σ_B^2, and S_B is the down-sampling operator corresponding to the Bayer pattern (e.g. RGGB), resulting in the observed mosaiced image $b \in \mathbb{R}^P$, where $M = 3P$.

Additionally, for the complementary pixel values, we can hypothesize a forward model:

$$c = S_C H u + n_C \tag{2}$$

where S_C is the down-sampling operator corresponding to the remaining 2/3 of the original pixel values that are complementary to the Bayer pattern (therefore $c \in \mathbb{R}^{2P}$). The term $n_C \approx \mathcal{N}(0, \sigma_C^2)$, with variance σ_C^2, represents additive noise associated with these complementary pixels.

3.1. Inverse problem

The inverse problem seeks to reconstruct the high-resolution image u from a degraded observation b. Our approach incorporates not only the forward model for the Bayer measurement b (Equation (1)) but also a second forward model for the complementary pixel values c (Equation (2)). Since c is not directly available, we estimate a rough reconstruction $u_0 = D(b)$ using an initial demosaicking method D. From this reconstruction, the complementary pixel values are approximated as $c \approx S_C u_0$. The inverse problem is inherently ill-posed due to the combined effects of blur, noise, and incomplete color information, requiring a robust optimization strategy.

In our framework, the inverse problem is formulated as training an INR, u_ψ, to reconstruct the high-resolution image u by parameterizing it as a continuous function modeled by the weights ψ of an MLP. Optimization of parameters ψ ensures that the outputs of u_ψ, when passed through the degradation models, match both the observed Bayer measurement b and the complementary pixel estimates c. Furthermore, added regularization promotes smoothness and edge preservation. This optimization is carried out for each individual image using stochastic gradient descent or its variants, with backpropagation applied to minimize loss functions derived from the forward models of b and c.

Formally, the optimization problem is expressed as:

$$\hat{\psi} = \arg\min_{\psi} \Big\{ \alpha \mathcal{L}_{\text{Bayer}}\Big(\hat{b}, b\Big) + \beta \mathcal{L}_{\text{Demo}}\Big(\hat{c}, S_C u_0\Big) + \gamma \mathcal{R}\Big(u_\psi\Big) \Big\}, \tag{3}$$

where $u_{\hat{\psi}}$ is the final reconstruction. $\hat{b} = S_B H u_\psi$ represents the predicted INR that is subject to the given degradation and corresponds to the Bayer pattern. To perform the degradation we sample u_ψ at all pixel locations and consider its vectorized form. The Bayer loss $\mathcal{L}_{\text{Bayer}}(\hat{b}, b)$ ensures fidelity to the original sensor data. Additionally, $\hat{c} = S_C H u_\psi$ denotes the degraded INR at complementary pixel locations, which lack direct Bayer measurements. The complementary loss, $\mathcal{L}_{\text{Demo}}$, minimizes the error between $\hat{c}$ and the corresponding values in the initial demosaiced image u_0. The overall optimization is balanced by the weighting factors α, β, and γ, which control the contributions of the Bayer loss, complementary loss, and the Total Variation (TV) regularization $\mathcal{R}(u_\psi)$.

In our ablation study for selecting optimal weighting factors (Section 4.4), β is fixed at 1 while α is varied to balance the Bayer and complementary losses. The parameter γ, when set to values between 10^{-6} and 10^{-5}, is used specifically for joint demosaicking, and deblurring tasks, as described in Section 4.6. The specific values of these weighting factors are further detailed in the experimental section.

3.2. Bayer loss

The Bayer loss $\mathcal{L}_{\text{Bayer}}$ is defined as the MSE between the predicted Bayer image and the observed (inherently blurred) mosaiced image b:

$$\mathcal{L}_{\text{Bayer}}\Big(\hat{b}, b\Big) = \frac{1}{P} \| S_B H u_\psi - b \|_2^2. \tag{4}$$

3.3. Complementary loss

The complementary loss $\mathcal{L}_{\text{Demo}}$ is calculated as the MSE between the predicted complementary pixel values and the corresponding values in the initial demosaiced (inherently blurred) image u_0:

$$\mathcal{L}_{\text{Demo}}\Big(\hat{c}, S_C u_0\Big) = \frac{1}{2P} \| S_C H u_\psi - S_C u_0 \|_2^2. \tag{5}$$

3.4. Total variation regularization

We apply Color TV regularization $\mathcal{R}(u_\psi)$ to ensure smoothness while preserving edges [29]. Total variation measures the gradient magnitude across the image, penalizing rapid intensity changes to reduce noise and retain key features. In INR models, the continuous image representation allows gradient computation at any point using automatic differentiation, enabling efficient total variation minimization. In our framework, TV regularization proves especially beneficial for tasks such as joint demosaicking and deblurring, where it not only stabilizes the reconstruction process but also helps preserve important image details, making it particularly impactful for processing real-world images.

Table 1
Configurations for INR Models.

Parameter	Gauss	ReLU	FFN	SIREN	WIRE	INCODE
Activation	Gaussian	ReLU	ReLU	Sine	Wavelet	Sine
Modulation	—	—	Positional Encoding (Gaussian)	—	—	Harmonizer (ResNet34)
Hidden Layers	5	5	5	5	5	5
Neurons per Layer	256	256	256	256	256	256
Learning Rate	1×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-4}	7×10^{-4}	1×10^{-4}
Batch Size	128×128	128×128	128×128	128×128	128×128	128×128
Trainable Parameters	330499	330499	461059	330499	330499	568359
Special Parameters	—	—	—	$\omega_{\text{first}} = 30$, $\omega_{\text{hidden}} = 30$	$\omega = 30$, $\sigma = 10$	$a = 0.1993$, $b = 0.0196$, $c = 0.0588$, $d = 0.0269$

Table 2
Image Reconstruction With INR: Average PSNR values for image representation using different INR models on the Kodak dataset, across various image sizes and training iterations. **Bold** and underline highlight the highest and second highest values, respectively.

INR Model	Original size (768×512)			1/2 Resize (384×256)			1/4 Resize (192×128)		
	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter
Gauss	31.01	33.68	35.99	34.26	38.97	43.25	60.24	**90.23**	80.24
ReLU	21.92	22.52	23.07	21.79	22.82	23.61	21.60	23.22	24.60
SIREN	37.89	40.29	41.92	39.62	46.46	49.88	44.50	51.73	60.52
WIRE	37.32	40.51	42.65	41.71	43.04	46.79	56.73	66.70	74.74
FFN	32.83	34.98	36.88	35.15	40.05	43.76	37.60	45.29	52.35
Incode	**39.65**	**41.35**	**42.87**	**52.78**	**50.21**	**51.93**	**72.23**	79.72	**90.94**

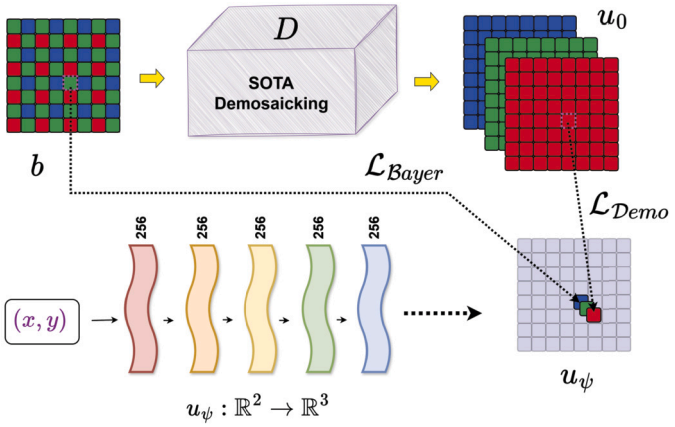


Fig. 2. Illustration of INRID: The proposed approach performs demosaicking using an implicit neural representation $u_\psi : \mathbb{R}^2 \to \mathbb{R}^3$, optimized by minimizing the mean squared error $\mathcal{L}_{Bayer}$ between the reconstruction u_ψ and the Bayer measurement b, as well as between the reconstruction u_ψ and the initial demosaicked image u_0 ($\mathcal{L}_{Demo}$). The INR consists of five layers, each with 256 neurons, and employs sinusoidal activations to effectively capture high-frequency image details. Unlike traditional activation functions such as ReLU or sigmoid, sinusoidal activations enable a more expressive representation, improving the reconstruction of fine structures and textures critical for accurate demosaicking (see ablation study in Section 4.3).

We call the algorithm that solves (3) INRID, standing for Implicit Neural Representation for Image Demosaicking. Fig. 2 provides a conceptual overview of the INRID framework. It highlights the key components: the raw Bayer measurement b, the initial demosaicked image u_0, and the learned implicit representation u_ψ. This high-level visualization is intended to help readers grasp the primary relationships and flow of the optimization process.

4. Experimental results

To solve the minimization in (3), we employ a self-supervised approach where the INR is trained directly on the degraded image data without requiring ground truth high-resolution images. This enables the INR model to reconstruct the high-resolution image solely based on the observed mosaiced image and complementary pixel information.

We begin by demonstrating image representation using INR and comparing various architectures. Next, we show that using the Bayer loss only for image representation exceeds basic demosaicking approaches such as nearest neighbor and bilinear interpolation, and in some cases outperforms traditional methods like Malvar and Menon. We then illustrate how the combination of Bayer and complementary loss within the INRID framework significantly improves reconstruction performance and exceeds all traditional methods. Furthermore, we showcase the joint demosaicking, denoising, or deblurring capabilities of INRID, enhancing state-of-the-art demosaicking methods such as DeepDemosaick and RSTCANet. Finally, we demonstrate the effectiveness of our approach on real-world data from mobile phone cameras.

4.1. Experimental setup

Table 1 summarizes the hyperparameter configurations for all INR models used in our experiments. Each model consists of five hidden layers with 256 neurons per layer. The Gauss and ReLU models employ Gaussian and ReLU activation functions, respectively, while the Fourier Feature Networks (FFN) utilize Gaussian positional encoding with ReLU activations. The SIREN model uses sine activation functions, parameterized by frequency terms ω for the first and hidden layers. The WIRE model incorporates a wavelet activation function, characterized by a frequency term ω and a scale term σ, which enable the balance of global and local signal representation. The INCODE model builds on a modified SIREN architecture, augmented with a harmonizer network based on the ResNet34 [27] backbone. The specific parameters, including frequency and scale terms for SIREN, WIRE, and INCODE, are summarized in Table 1 and detailed in their respective original works [22–24].

The training was conducted using an Nvidia L40s GPU. All models were optimized using the MSE loss function, and the Adam optimizer, with decay rates for gradient and squared gradient averages set to 0.9 and 0.999, respectively. A learning rate scheduler was applied to gradually reduce the learning rate during training. The initial learning rate was set to 0.0001 for most models, except for WIRE, which used a

Table 3
Image Reconstruction With INR: Average PSNR values for image representation using different INR models on the McM dataset (500 × 500 version), across various image sizes and training iterations. **Bold** and underline highlight the highest and second highest values, respectively.

INR Model	Original size (500 × 500)			1/2 Resize (250 × 250)			1/4 Resize (125 × 125)		
	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter
Gauss	29.65	32.75	36.14	36.50	45.51	51.32	86.48	113.20	**117.79**
ReLU	20.81	21.67	22.47	19.82	21.14	22.31	18.61	20.49	22.34
SIREN	39.10	41.79	43.61	39.19	46.08	51.71	44.07	52.38	62.06
WIRE	37.28	41.05	44.17	46.10	49.58	47.83	53.00	58.83	65.72
FFN	35.16	37.67	39.47	35.89	41.24	46.02	32.70	43.02	52.30
Incode	**41.56**	**43.17**	**44.64**	**64.79**	**67.54**	**53.96**	**107.24**	**116.99**	117.33

learning rate of 0.0007. Batch sizes were fixed at 128×128 for all experiments.

To evaluate the performance of the INR models, we tested on the Kodak [30] and McMaster [31] datasets. The evaluation metrics included Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM).

The source code used in this study is publicly available at https://github.com/kereptom/inrid2024.

4.2. Image reconstruction with INR

In the image reconstruction experiment, we evaluated the performance of different INR architectures in representing images. In other words, we trained INR models in a self-supervised manner to fit the original image. This corresponds to setting $\alpha = 1$, $\beta = 2$, and $\gamma = 0$ in Equation (3), with the initial reconstruction u_0 replaced by the original ground truth pixel values u. We set $\beta = 2$ because the complementary loss involves twice as many pixels as the Bayer loss.

Specifically, we tested six different INR architectures across three image sizes and three different numbers of iterations, calculating average PSNR results for both the Kodak and McM datasets. The results, as shown in Tables 2 and 3, indicate that the INR model with ReLU activation consistently performed the worst across all conditions. On average, INCODE delivered the best results in nearly all scenarios. SIREN and WIRE were strong contenders, especially at the original size and half size. While both Gauss and FFN showed moderate results overall, FFN performed slightly better at the original size, whereas Gauss was particularly effective for smaller images. SIREN, although not always achieving the highest scores, produced stable and reliable results across different image sizes and iteration counts, making it a strong performer in a wide range of conditions.

The visual demonstration in Fig. 3 supports these findings, showing the reconstruction of Kodak image #23 at its original size after 2000 iterations. The ReLU INR model shows significant blurring, particularly in areas with fine textures. In contrast, the other methods produce visually pleasing and accurate reconstructions, with INCODE, WIRE, and SIREN standing out for their near-perfect results (see Fig. 3, especially in the close-ups).

We also analyzed the progression of PSNR values with extended training on the McM dataset (500 × 500 version) beyond 2000 iterations, as shown in Fig. 4. The results reveal continued improvement across all models, but with a diminishing rate of gain after 2000 iterations. Models such as INCODE, WIRE, and SIREN exhibit high performance and retain their advantage. Given this diminishing improvement, it becomes important to consider the trade-off between further enhancing reconstruction quality and the associated computational cost, which will be discussed further in Section 5.

Although WIRE showed competitive performance, we encountered instability with the learning rate, making its training less reliable compared to other models. Based on these results, we chose to proceed with two INR architectures for further experiments: INCODE, which dominated in most scenarios, and SIREN, which consistently performed well and demonstrated stability across various conditions; and also included FFN and Gauss for reference.

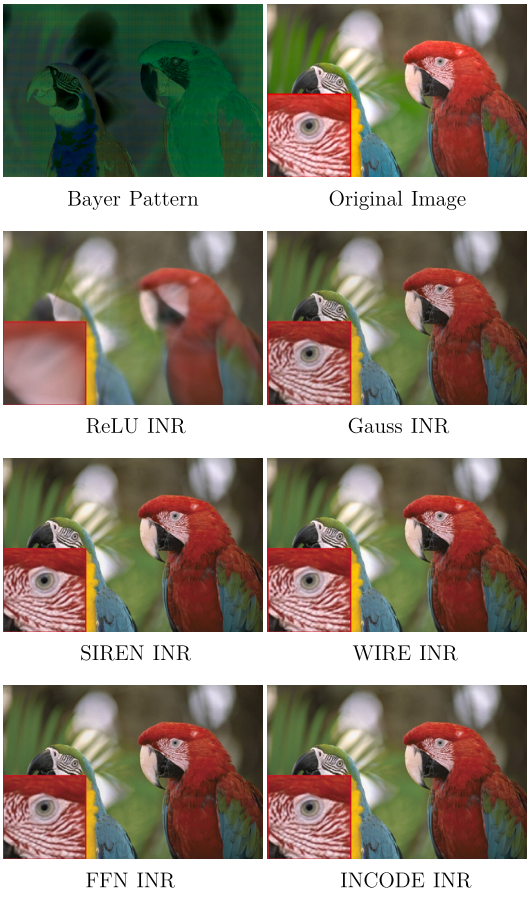


Fig. 3. Image reconstruction using different INR architectures on an example from the Kodak dataset. The Bayer Pattern (top left) shows the raw subsampled data for demonstration purposes. All INR models were trained in a self-supervised manner to fit the original image (top right). ReLU INR struggles to model high-frequency details, resulting in noticeable blurring, especially in regions with fine textures, such as the bird's feathers. In contrast, INCODE, SIREN and WIRE architectures provide the most visually pleasing reconstructions, capturing details with higher fidelity. This example illustrates the results after 2000 training iterations for each INR architecture. The corresponding average PSNR values for the entire dataset are reported in Table 2, 4th column.

4.3. Image demosaicking with INR

Following our image representation study, we extended our experiments to image demosaicking using INR architectures, focusing solely on Bayer measurements, which is equivalent to setting $\alpha = 1$, $\beta = 0$, and $\gamma = 0$ in Equation (3). The results, shown in Tables 4 and 5, indicate a decline in PSNR values as image size decreases, contrasting with the full

Table 4
Image Demosaicking With INR: Average PSNR values for image demosaicking using various INR models on the Kodak dataset. The models were overfitted on Bayer measurements across different image sizes and training iterations, as opposed to Table 2, where all image pixels were taken into account. In this setup, minimization was performed using the objective in (3), where the complementary loss was neglected ($\beta = 0$) and $\gamma = 0$. **Bold** and underline highlight the highest and second highest values, respectively.

INR Model	Original size (768 × 512)			1/2 Resize (384 × 256)			1/4 Resize (192 × 128)		
	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter
Gauss	25.66	26.76	27.75	16.23	16.04	17.91	14.23	14.43	14.29
SIREN	**34.21**	**34.11**	33.96	**31.25**	**31.19**	**30.93**	**31.11**	**31.38**	**31.33**
FFN	31.36	33.13	**34.41**	28.75	29.93	30.22	25.34	25.16	25.40
Incode	33.38	33.64	33.95	26.54	28.23	30.10	20.28	20.37	20.47

Table 5
Average PSNR values for image demosaicking using various INR models on the MCM dataset, following the same setup as described for Table 4. **Bold** and underline highlight the highest and second highest values, respectively.

INR Model	Original size (500 × 500)			1/2 Resize (250 × 250)			1/4 Resize (125 × 125)		
	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter	500 Iter	1000 Iter	2000 Iter
Gauss	21.83	22.48	23.57	13.71	13.82	14.13	12.30	12.27	12.23
SIREN	**34.87**	**34.75**	34.71	**31.49**	**31.63**	**31.48**	**29.48**	**29.88**	**29.88**
FFN	32.26	33.15	33.75	25.34	26.29	26.64	19.09	20.19	20.33
Incode	33.99	34.65	**35.16**	21.99	22.89	25.49	15.45	15.58	15.50

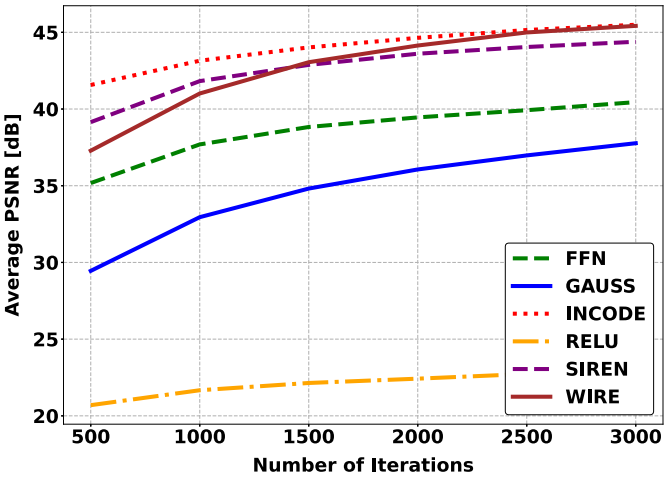


Fig. 4. Image Reconstruction With INR: Average PSNR values with increasing training iterations for various INR models on the McM dataset (500 × 500 version). The plot demonstrates continued improvement in PSNR with additional iterations, though the rate of gain decreases over time for all models.

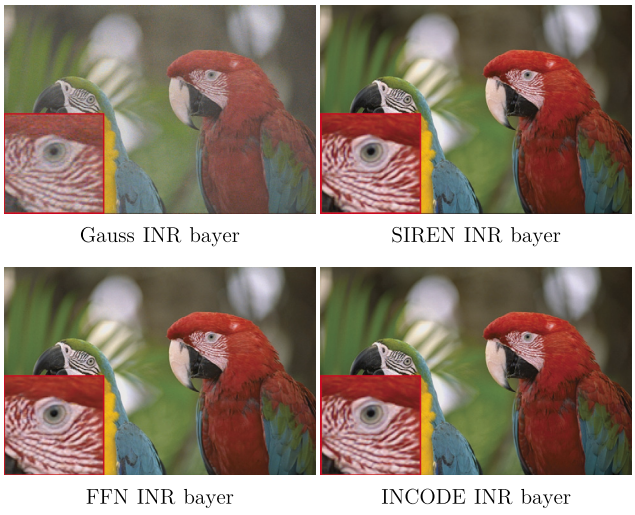


Fig. 5. Image Demosaicking With INR: Demosaicking results for Kodak image #23 (original size) using various INR architectures trained on Bayer measurements (Fig. 3, top-left). INCODE, SIREN, and FFN outperform the Gaussian model after 2000 iterations. PSNR values are listed in Table 4, 4th column.

image representation results in the previous section. This decline is due to the reduced availability of ground truth pixels and increased impact of CFA degradation in smaller images.

Interestingly, while INCODE excelled in full image representation, the SIREN architecture outperforms it in the demosaicking task, particularly with smaller images. SIREN's superior PSNR values highlight its robustness in scenarios requiring significant interpolation.

The visual demonstration is presented in Figs. 5 and 6. In the original size (Fig. 5), only Gauss exhibits improper reconstruction. When resized to half the original size (Fig. 6), SIREN begins to handle the reconstruction more effectively, producing a more colorful image. As the image size is reduced further, SIREN becomes the only model capable of adequately managing the interpolation.

The naive approach to INR-based demosaicking explained in this section, especially when using the SIREN architecture, surpasses basic algorithms like nearest neighbor and bilinear interpolation (see Table 6). As will be seen in the next section, it also highlights the potential of SIREN for boosting traditional demosaicking methods when initial information about missing pixels is provided.

4.4. Enhancing image demosaicking with INR

We now take full advantage of the INRID framework by incorporating both Bayer and complementary loss functions. This experiment corresponds to setting $\beta = 1$ in Equation (3), while varying α to balance the contributions of the Bayer and complementary losses. We keep the TV regularization turned off.

The inclusion of complementary loss leverages initial demosaicking reconstructions, regularizing the problem and, with the aid of Bayer loss, ultimately boosting the demosaicking capabilities of the original methods. This approach helps the INR model to learn from not only the available Bayer data but also the estimated values from the initial demosaicking process, thus improving the overall reconstruction quality.

Optimal Alpha Selection: To determine the optimal value for α, we conducted experiments using various demosaicking methods. Fig. 7 shows the average PSNR and SSIM from the McM dataset as a function of α for traditional demosaicking methods such as Malvar and Menon. The results indicate that, while the complementary loss plays a crucial role

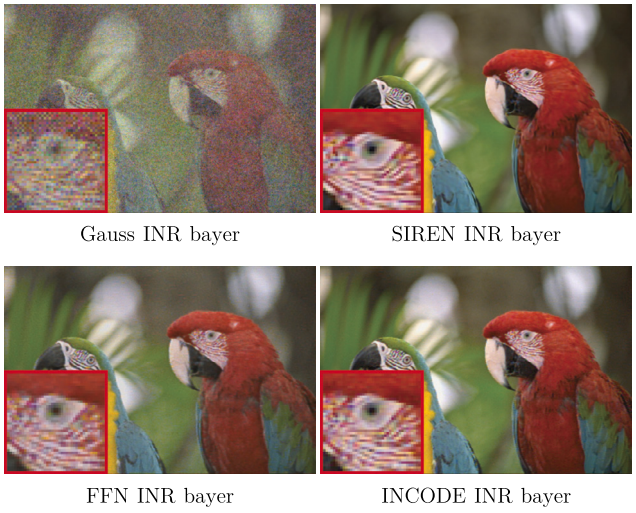


Fig. 6. Image Demosaicking With INR: Kodak image #23 resized to 384×256. SIREN INR produces the best visual outcome after 2000 iterations, while Gaussian INR shows significant artifacts. PSNR values are in Table 4, 7th column.

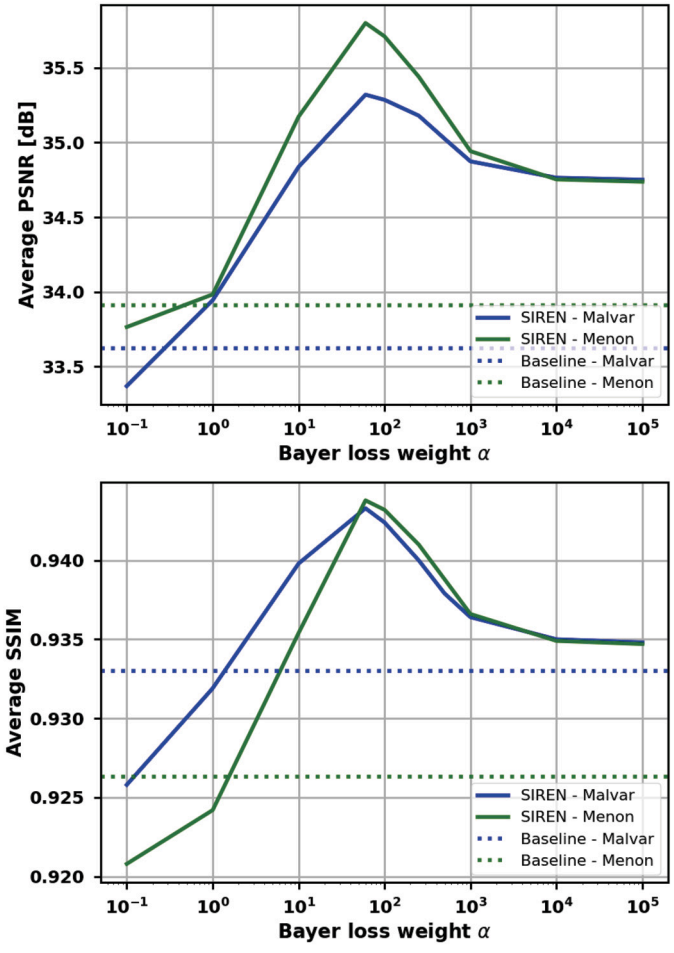


Fig. 7. Optimal Alpha Selection: Average PSNR and SSIM vs Alpha for SIREN-based demosaicking over the McM dataset (500×500). The plots show the performance of SIREN models to improve Malvar and Menon demosaicking methods as a function of α. The solid lines represent SIREN results, while the dotted lines show the baseline performance. The results indicate that for $\alpha > 1$, our INRID framework enhances demosaicking quality, with peak improvements around $\alpha = 60$, as seen in both PSNR and SSIM. As α approaches infinity, performance approaches the naive INR-based demosaicking.

Table 6
Enhancing Image Demosaicking with INR: Average PSNR and SSIM values for different demosaicking methods on the McM (500×500) dataset after 2000 iterations and the Kodak (192×128) dataset after 10000 iterations with $\alpha = 60$. **Bold** indicates the highest values.

Method	McM (500×500) PSNR/SSIM	Kodak (192×128) PSNR/SSIM
Nearest Neighbor	27.54/0.8594	25.11/0.7973
Bilinear	30.41/0.9276	26.61/0.8685
Bayer INRID	**34.71/0.9348**	**31.33/0.9283**
Malvar	33.62/0.9330	31.68/0.9420
Malvar INRID	**35.31/0.9433**	**32.58/0.9449**
Menon	33.91/0.9263	33.29/0.9571
Menon INRID	**35.80/0.9438**	**33.74/0.9586**
RSTCANet	**40.06/0.9739**	**38.40/0.9839**
RSTCANet INRID	36.95/0.9501	38.38/0.9836

in guiding the training process, the Bayer loss remains more dominant. When α is below 1, which emphasizes the complementary loss more than the Bayer loss, performance degrades compared to the baseline (dotted line), providing no enhancement at all. However, when α is greater than 1, INRID begins to enhance the original demosaicking methods, with the most significant improvements occurring when α is within the range of $(10, 200)$. As α increases further towards infinity, the complementary loss influence diminishes, and the model essentially reverts to the naive INR demosaicking approach discussed in the previous section. Based on these findings, we selected $\alpha = 60$, which yielded the best average improvements in both PSNR and SSIM.

Boosting Demosaicking Performance: Table 6 highlights the impact of the INRID approach in improving traditional demosaicking techniques, specifically Malvar and Menon, using the SIREN architecture. The results are consistent across both the Kodak and McM datasets, where the INRID framework significantly boosts the performance, leading to noticeable improvements in both PSNR and SSIM. For both traditional demosaicking methods, Malvar and Menon, integrating the INRID approach results in visibly better reconstruction quality, particularly in challenging areas with fine details or high-frequency content, as illustrated in Fig. 8.

Basic demosaicking algorithms like nearest neighbor and bilinear interpolation are surpassed by even the naive INR demosaicking (Bayer INRID) introduced in the previous section. For these methods, incorporating initial reconstruction degrades the enhancement.

It is worth noting, however, that INRID has limitations. Once the initial demosaicking reconstruction reaches a certain level of accuracy, further improvement of a given method is limited. This is evident in the PSNR values for the Transformer-based demosaicking method RSTCANet, as seen in Table 6. However, for state-of-the-art methods, the INRID framework can still be valuable when addressing joint problems such as demosaicking combined with denoising or deblurring.

4.5. Joint demosaicking and denoising

While INRID may not directly enhance state-of-the-art demosaicking methods, like RSTCANet, it shows significant improvements when dealing with out-of-distribution data, such as images corrupted by noise. To demonstrate this robustness, we conducted experiments on the joint demosaicking and denoising task using the Kodak dataset resized to 192×128. We introduced varying levels of Gaussian noise, with signal-to-noise ratios (SNRs) ranging from 10 dB to 40 dB, and then applied our INRID framework with RSTCANet as the initial demosaicking reconstruction.

We compared our approach against a baseline and two specialized methods for joint demosaicking and denoising. The first is a classical method that builds upon the demosaicking technique of Farsiu et

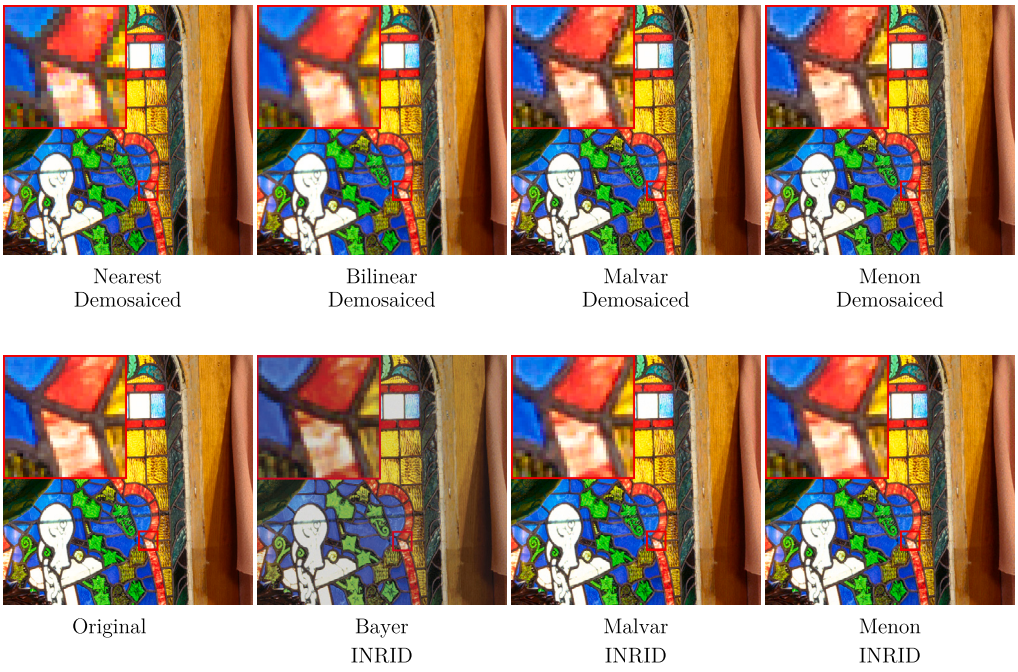


Fig. 8. Enhancing Image Demosaicking with INR: Visual Comparison of Demosaicking Methods with and without INRID Enhancement (applied to McM image #1, 500×500). The top row showcases traditional demosaicking results from Nearest Neighbor, Bilinear, Malvar, and Menon methods. The bottom row starts with the original image. The Bayer INRID approach overfits directly to the Bayer measurement without initial demosaicking, demonstrating a higher reconstruction quality compared to Nearest Neighbor and Bilinear methods. INRID significantly boosts the performance of the Malvar and Menon methods, particularly in high-frequency regions, such as along the stained glass edges (see red close-ups).

Table 7
Joint Demosaicking and Denoising: Average PSNR and SSIM values for different iterations and SNR levels on the Kodak dataset (192×128). The parameters are: $\alpha = 1$, $\beta = 1$, and $\gamma = 0$, with noise levels in SNR (dB). INRID uses initial reconstruction from RSTCANet.

Model	Iterations	PSNR / SSIM			
		10 dB	20 dB	30 dB	40 dB
Classical (HQ)	-	23.68/0.5826	27.75/0.7934	30.20/0.9063	30.78/0.9348
DeepDemosaick	-	17.80/0.2985	28.83/0.8076	34.27/0.9489	35.70/0.9666
RSTCANet	-	17.51/0.2878	26.57/0.6620	34.31/0.9225	**37.74/0.9777**
INRID (RSTCANet init)	500 Iter	**23.75/0.6352**	**29.19**/0.8238	33.20/0.9247	33.94/0.9385
	1000 Iter	23.73/0.6324	29.17/**0.8250**	34.67/0.9415	36.48/0.9661
	2000 Iter	23.69/0.6309	29.11/0.8238	**34.77/0.9425**	37.63/0.9770

al. [18], formulated via half-quadratic (HQ) approximation in a multiplicative form [32] and solved by alternating minimization. While this approach can also integrate deblurring using a suitable kernel, we used a delta kernel here to focus solely on denoising and demosaicking. The second method is DeepDemosaick (introduced in Section 2), a deep convolutional residual network designed for joint demosaicking and denoising.

Since RSTCANet already provides high-quality initial reconstructions, we placed equal emphasis on the Bayer and complementary losses, setting $\alpha = 1$ and $\beta = 1$. To showcase the denoising capabilities of INR, TV regularization was disabled ($\gamma = 0$). To mitigate overfitting to noisy measurements, an early-stopping mechanism is employed for INRID, with training concluding after 500 to 2000 iterations.

Table 7 presents a comparative analysis of the four methods across various noise levels. For heavy to moderate noise conditions (10–30 dB SNR), INRID consistently surpasses RSTCANet and outperforms both the classical and deep-learning-based approaches. While the classical method delivers competitive results under severe noise conditions (10–20 dB SNR), its performance diminishes as noise levels decrease. Notably, DeepDemosaick closely matches INRID's performance at 20 dB SNR.

At 20 dB SNR, INRID achieves a PSNR of 29.19 dB after 500 iterations, outperforming RSTCANet's 26.57 dB and the classical method's 27.75 dB. DeepDemosaick achieves 28.83 dB at this noise level and its visual quality is comparable to INRID (see Fig. 9). Visually, RSTCANet and the classical method exhibit noticeable artifacts, whereas INRID effectively removes noise, especially around detailed regions such as window shutters.

Under extreme noise conditions (10 dB SNR), the advantage of INRID becomes more pronounced, with a PSNR of 23.75 dB compared to RSTCANet's 17.51 dB, representing a significant improvement. At 30 dB SNR, where noise levels are lower, the performance gap between INRID and its initial RSTCANet reconstruction narrows. Finally, at 40 dB SNR, where noise is minimal, RSTCANet achieves the highest PSNR (37.74 dB), and further refinement by INRID does not yield additional improvements. These results align with the conclusions drawn in the preceding section.

4.6. Joint demosaicking and deblurring

This experiment evaluates INRID's impact on traditional and advanced demosaicking methods when integrated with deblurring. Image

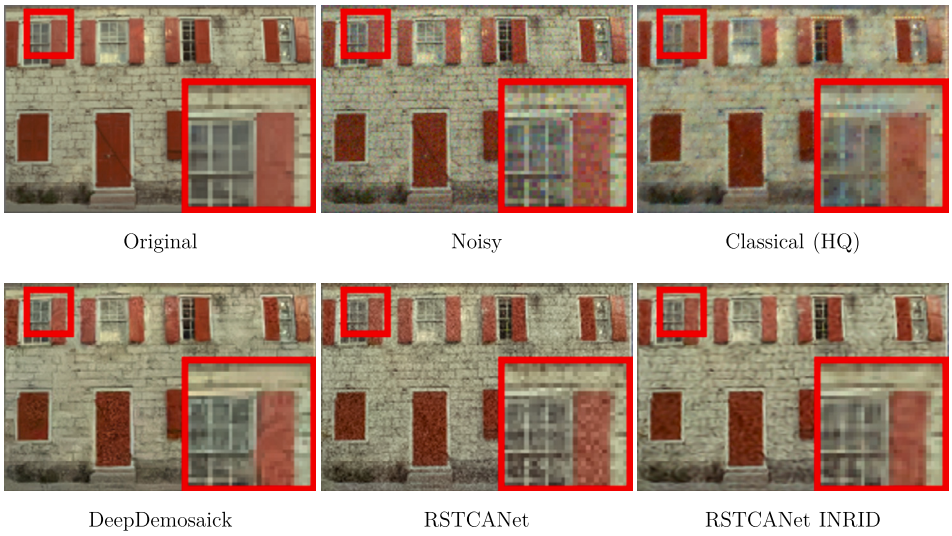


Fig. 9. Joint Demosaicking and Denoising: The first row presents the original image, its noisy counterpart (input SNR = 20 dB), and the output from the Classical method (HQ minimization). The second row showcases results from DeepDemosaick, RSTCANet, and RSTCANet INRID. The corresponding PSNR and SSIM results are shown in Table 7, 4th column. The image is from the Kodak dataset (resized to 192 × 128), with INRID parameters set to $\alpha = \beta = 1$ and $\gamma = 0$.

Table 8
Joint Demosaicking and Deblurring with Uniform Kernel: Average PSNR and SSIM for joint demosaicking and deblurring on the Kodak dataset (resized to 192 × 128) with a uniform kernel (3 × 3), 50 dB noise after 10,000 iterations. +TV indicates that total variation with $\gamma = 10^{-6}$ was added in the minimization problem (3). Significant improvements with INRID are particularly evident in methods like RSTCANet and DeepDemosaick, with the best results highlighted in **bold**. *Italicized* entries indicate demosaicking results without deblurring.

Method	Original	INRID Enhancement			
		$\alpha = 1$	$\alpha = 1$, +TV	$\alpha = 60$	$\alpha = 60$, +TV
Nearest	*25.71/0.7701*	27.51/0.8232	27.56/0.8236	31.57/0.9113	**31.58/0.9115**
Bilinear	*25.06/0.7999*	26.52/0.8420	26.52/0.8413	31.32/0.9120	**31.36/0.9124**
Malvar	*27.34/0.8293*	30.84/0.9071	30.96/0.9120	32.40/0.9252	**32.44/0.9257**
Menon	*27.03/0.8159*	31.19/0.9093	31.34/0.9144	32.67/0.9272	**32.74/0.9285**
DeepDemosaick	*27.29/0.8232*	32.17/0.9198	32.27/0.9211	32.95/0.9289	**33.01/0.9302**
RSTCANet	*27.31/0.8251*	33.38/0.9374	**33.60/0.9409**	33.24/0.9324	33.35/0.9344
Wiener Filtering	26.53/0.7959	—	—	—	—
IWFT	26.48/0.8330	—	—	—	—
D3Net	29.86/0.8736	—	—	—	—
HQ	31.66/0.9280	—	—	—	—
Bayer INRID	32.59/0.9241	—	—	—	—

quality degradation in such scenarios primarily arises from convolution operations, which introduce blur during image acquisition. Since initial demosaicking guides INRID in interpolating missing data, the effectiveness of deconvolution critically depends on the quality of this preliminary interpolation.

Experiments were conducted on the Kodak dataset using Gaussian and uniform blur kernels with added noise at 50 dB to simulate the forward problem (1). Results are summarized in Tables 8 and 9 for 3×3 kernels and in Tables 10 and 11 for 7×7 kernels. First, it shows that INRID enhancement of traditional techniques, such as Nearest neighbor, Bilinear interpolation and Malvar's method, is suboptimal and outperformed by INRID deblurring with Bayer measurements alone (Bayer INRID). The traditional approaches produce initial reconstructions that fail to adequately match the original image distribution, leading to insufficient deconvolution performance.

In contrast, state-of-the-art methods, such as DeepDemosaick and RSTCANet, deliver more accurate initial demosaicking results, which, when enhanced with INRID, yield significantly improved deblurring performance. For instance, under Uniform 3×3 blur (Table 9), DeepDemosaick improves from 27.29 dB to 33.01 dB, and RSTCANet improves from 27.31 dB to 33.60 dB.

In most scenarios, choosing $\alpha = 60$ yields higher PSNR and SSIM values, as discussed in Section 4.4. However, when using RSTCANet initialization for images blurred with a smaller 3×3 kernel, the best results occur at $\alpha = 1$.

The addition of TV regularization with $\gamma = 10^{-6}$ provides a modest improvement in PSNR for all methods (except RSTCANet with Gaussian 3×3 blur). Notably, for all tested scenarios, DeepDemosaick and RSTCANet paired with INRID outperform Bayer INRID, underscoring the importance of accurate initial demosaicking. For state-of-the-art methods, the complementary loss in INRID plays a significant role in enhancing reconstruction quality. Fig. 10 demonstrates how INRID integration performs demosaicking and deblurring effectively. Nevertheless, with larger blur kernels, the INRID enhancement of state-of-the-art demosaicked methods is less significant compared to the Bayer INRID (see Table 10).

We further compared INRID with other joint demosaicking and deblurring methods. Specifically, we extended Iterative Wiener Filtering and Thresholding (IWFT) [33] to handle demosaicking and deblurring by incorporating formation model (1) into equation (1) in [33]. After integrating the new degradation model, the algorithm was modified accordingly, and the rest follows the original IWFT pipeline. The first

Table 9
Joint Demosaicking and Deblurring with Gaussian Kernel: Average PSNR and SSIM for joint demosaicking and deblurring on the Kodak dataset (resized to 192 × 128) with a Gaussian kernel (3 × 3), 50 dB noise after 10,000 iterations. +TV indicates that total variation with $\gamma = 10^{-6}$ was added in the minimization problem (3). Significant improvements with INRID are particularly evident in methods like RSTCANet and DeepDemosaick, with the best results highlighted in **bold**. *Italicized* entries indicate demosaicking results without deblurring.

Method	Original	INRID Enhancement			
		$\alpha = 1$	$\alpha = 1$, +TV	$\alpha = 60$	$\alpha = 60$, +TV
Nearest	*25.98/0.7877*	27.57/0.8314	27.56/0.8311	31.54/0.9142	**31.54/0.9145**
Bilinear	*25.54/0.8218*	26.77/0.8552	26.77/0.8548	31.31/0.9152	**31.34/0.9155**
Malvar	*28.48/0.8666*	31.02/0.9125	31.03/0.9145	32.80/0.9319	**32.85/0.9320**
Menon	*28.24/0.8600*	31.32/0.9141	31.33/0.9158	32.89/0.9331	**32.90/0.9336**
DeepDemosaick	*28.39/0.8567*	32.22/0.9249	32.15/0.9234	33.15/0.9354	**33.17/0.9359**
RSTCANet	*28.64/0.8693*	**33.84/0.9440**	33.66/0.9415	33.74/0.9401	33.71/0.9398
Wiener Filtering	28.13/0.8336	—	—	—	—
IWFT	28.18/0.8540	—	—	—	—
D3Net	31.06/0.8988	—	—	—	—
HQ	32.45/0.9399	—	—	—	—
Bayer INRID	33.07/0.9329	—	—	—	—

Table 10
Joint Demosaicking and Deblurring with Uniform Kernel: Average PSNR and SSIM on the Kodak dataset (resized to 192 × 128) with a uniform kernel (7 × 7), 50 dB noise after 10,000 iterations. +TV indicates that total variation with $\gamma = 10^{-6}$ was added in the minimization problem. Each row's best PSNR/SSIM is in **bold**. *Italicized* entries indicate demosaicking results without deblurring.

Method	Original	INRID Enhancement			
		$\alpha = 1$	$\alpha = 1$, +TV	$\alpha = 60$	$\alpha = 60$, +TV
Nearest	*23.32/0.5986*	26.57/0.7627	26.61/0.7666	29.60/0.8450	**29.72/0.8499**
Bilinear	*22.60/0.6078*	24.47/0.6804	24.47/0.6812	28.94/0.8274	**28.96/0.8291**
Malvar	*23.55/0.6084*	27.87/0.8071	27.97/0.8125	29.94/0.8514	**30.10/0.8574**
Menon	*23.50/0.6065*	29.25/0.8422	29.38/0.8474	30.12/0.8559	**30.36/0.8637**
DeepDemosaick	*23.55/0.6105*	29.06/0.8329	28.87/0.8273	30.11/0.8545	**30.31/0.8615**
RSTCANet	*23.55/0.6096*	30.16/0.8587	29.97/0.8548	30.14/0.8549	**30.39/0.8623**
Wiener Filtering	25.28/0.7129	—	—	—	—
IWFT	25.21/0.7114	—	—	—	—
D3Net	26.16/0.7266	—	—	—	—
HQ	30.01/0.8773	—	—	—	—
Bayer INRID	30.11/0.8540	—	—	—	—

Table 11
Joint Demosaicking and Deblurring with Gaussian Kernel: Average PSNR and SSIM on the Kodak dataset (resized to 192 × 128) with a Gaussian kernel (7 × 7), 50 dB noise after 10,000 iterations. +TV indicates that total variation with $\gamma = 10^{-6}$ was added in the minimization problem. Each row's best PSNR/SSIM is in **bold**. *Italicized* entries indicate demosaicking results without deblurring.

Method	Original	INRID Enhancement			
		$\alpha = 1$	$\alpha = 1$, +TV	$\alpha = 60$	$\alpha = 60$, +TV
Nearest	*24.03/0.6509*	26.58/0.7752	26.43/0.7718	29.22/0.8420	**29.29/0.8448**
Bilinear	*23.24/0.6619*	24.87/0.7180	24.87/0.7190	28.58/0.8275	**28.64/0.8305**
Malvar	*24.47/0.6734*	27.58/0.8092	27.41/0.8048	29.35/0.8441	**29.37/0.8465**
Menon	*24.37/0.6681*	28.79/0.8360	28.28/0.8227	29.46/0.8451	**29.55/0.8489**
DeepDemosaick	*24.41/0.6685*	28.45/0.8248	27.84/0.8087	29.44/0.8456	**29.49/0.8482**
RSTCANet	*24.44/0.6716*	29.50/0.8480	28.55/0.8249	29.49/0.8459	**29.55/0.8488**
Wiener Filtering	25.34/0.7487	—	—	—	—
IWFT	25.78/0.7909	—	—	—	—
D3Net	26.58/0.7334	—	—	—	—
HQ	29.06/0.8543	—	—	—	—
Bayer INRID	29.46/0.8445	—	—	—	—

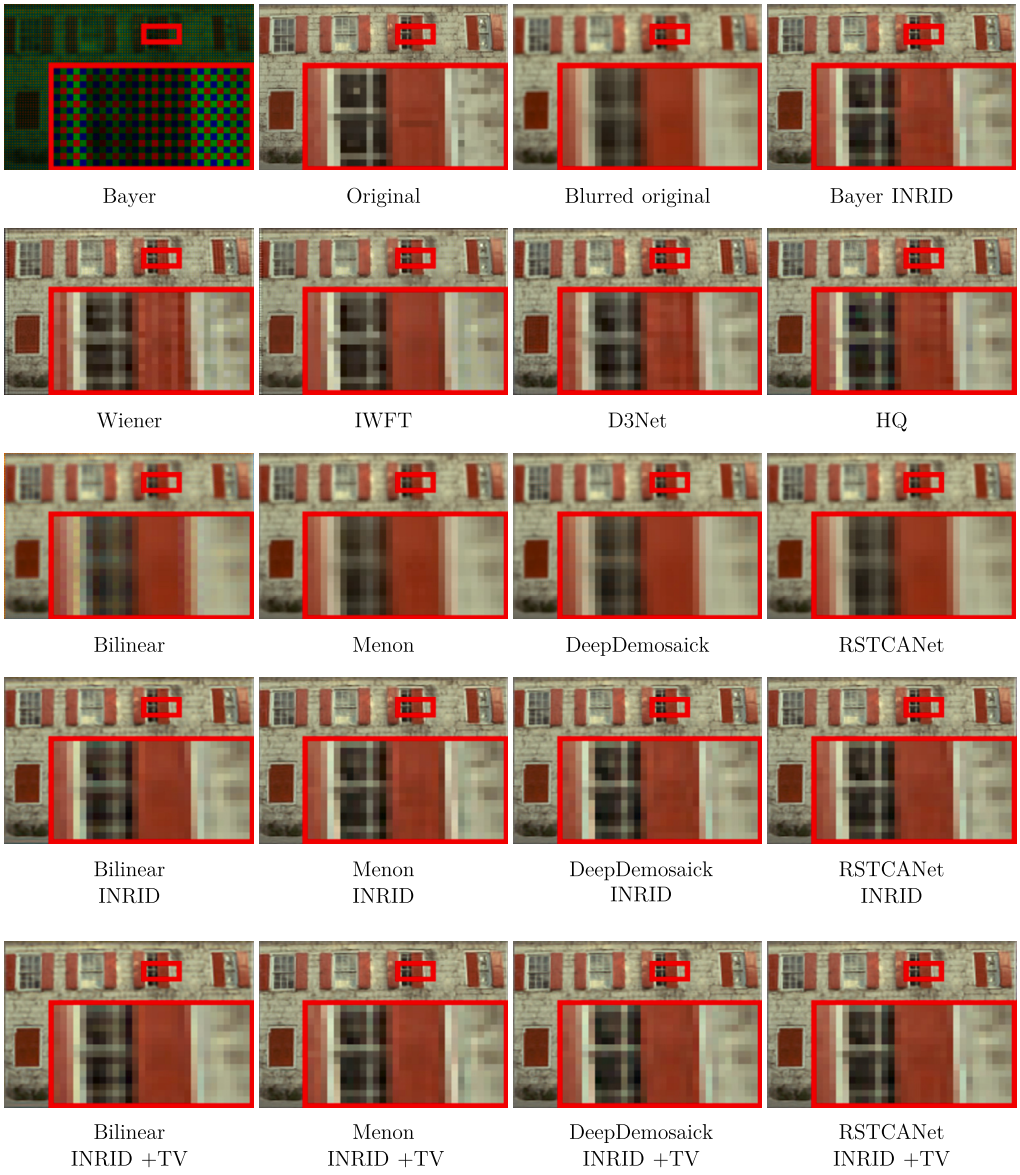


Fig. 10. Joint Demosaicking and Deblurring: Visual example of INRID's performance on Kodak image #1 (192 × 128), degraded with 3 × 3 Gaussian blur and 50 dB noise. Corresponding PSNR, SSIM, and other metrics are in Table 9.

reconstruction step in IWFT corresponds to Wiener Filtering, a popular deconvolution technique. When applied to our test set, this method introduced ringing artifacts around edges due to its linear nature (clearly visible around the window shutters in Fig. 10).

The complete IWFT algorithm then uses non-linear update steps to refine the Wiener-based reconstruction, effectively suppressing these artifacts and producing smoother images. However, IWFT tends to oversmooth images, leading to a slight reduction in PSNR (specifically in the case of uniform blur) despite the noticeable visual improvements.

D3Net [9] is an end-to-end CNN developed for joint demosaicking, deblurring, and deringing. It surpasses IWFT in terms of PSNR but remains constrained by its lightweight architecture, which targets embedded devices with limited computational resources. HQ is a robust optimization framework for joint demosaicking and deblurring method introduced in Section 4.5. While HQ outperforms D3Net, it still does not reach the reconstruction quality offered by INRID methods.

While INRID significantly enhances demosaicking and achieves superior reconstruction quality compared to baseline and joint techniques, its computational cost remains a notable drawback. Fig. 11 illustrates the average PSNR and processing time for RSTCANet INRID over the Kodak dataset. The PSNR improves steadily up to approximately 11,000 iterations, after which it saturates, whereas the runtime continues to increase linearly, exceeding 100 seconds for 11,000 iterations on an NVIDIA L40S GPU. In contrast, traditional methods such as Wiener filtering, IWFT, and HQ are significantly faster, completing reconstruction in just a few seconds. D3Net achieves similar inference times to these methods but requires several minutes of pretraining for each specific blur and noise level. Despite the superior reconstruction quality of INRID-enhanced methods, their computational cost limits their practicality for scenarios demanding fast or resource-efficient processing, a limitation that will be further addressed in Section 5.

4.7. Real data

To further validate the reconstruction effectiveness of our INRID approach, we tested it on real raw data captured from an LG Nexus 5 camera. The blur kernels (displayed in Fig. 1b) were estimated from calibration data.

This experiment demonstrates the practical benefits of applying joint demosaicking and deblurring on actual image data. As shown in Fig. 12,

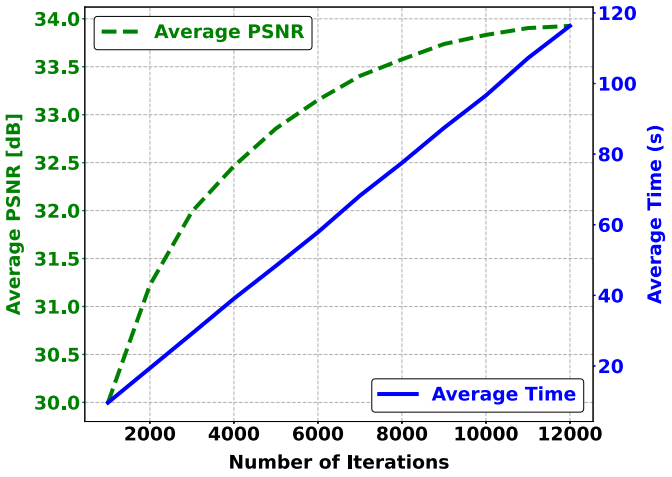


Fig. 11. Computational cost of RSTCANet-INRID with $\alpha = \beta = 1$ and $\gamma = 0$ on the Kodak dataset (resized to 192×128). The figure shows the average PSNR (green dashed line) and runtime (blue solid line) as a function of the number of iterations, using the parameters described in Table 9. The PSNR improves steadily until saturating around 11,000 iterations, while the runtime increases linearly, reaching 2 minutes for 12,000 iterations on an NVIDIA L40S GPU.

the INRID framework, combined with prior information about the camera's PSFs, yields a significant improvement over both the standard JPEG output and the advanced RSTCANet model.

The raw Bayer data are seen in the second column of Fig. 12. The JPEG output (third column) exhibits considerable compression artifacts and blurring, especially in magnified areas. RSTCANet, shown in the fourth column, improves the reconstruction quality but still leaves some residual blurring and noise.

In contrast, the final column illustrates the result of applying RSTCANet in conjunction with the INRID framework, leveraging PSF priors for all four RGGB channels in the raw Bayer data. By setting $\alpha = 1$ and $\beta = 1$, and enabling TV regularization with a value of 10^{-5}, our approach effectively removes noise and reduces blurring. This leads to a visually sharper and more accurate reconstruction, as highlighted by the red close-up in Fig. 12. Zoomed-in views of the green and blue bordered regions show further evidence of INRID's ability in enhancing the baseline RSTCANet method.

5. Discussion and future work

The results demonstrate the significant potential of INRID in enhancing traditional and state-of-the-art demosaicking methods. However, the computational cost associated with per-image training remains a notable limitation, particularly for large datasets or high-resolution images. While this study focuses on reconstruction quality, addressing efficiency is a crucial challenge for expanding the practical utility of INRID, especially in real-time applications.

Training INRID for each image is time-intensive. For example, processing a 192×128 image required approximately 96 seconds for 10,000 iterations on an NVIDIA L40s GPU, using around 1 GB of memory. This is orders of magnitude slower than traditional methods like Malvar or Menon, which complete within milliseconds for similar resolutions. Similarly, pre-trained models like RSTCANet offer real-time inference but fail to handle scenarios involving corrupted inputs, such as blurred or noisy data. In contrast, iterative joint demosaicking and deblurring methods process images within seconds but may not match INRID's reconstruction fidelity.

To make INRID more practical, future efforts should aim to reduce its computational overhead. Promising approaches include multiresolution hash encoding [34], which could cut training times to seconds, and dictionary-based representations like Neural Implicit Dictionary (NID) [35], which leverage pre-learned basis functions for efficient reconstructions without a long per-image training.

Scaling INRID to gigapixel-resolution images also presents challenges due to the extensive training times required for basic architectures. Techniques such as tiling, which processes overlapping sections of an image with smaller MLPs, can enable parallel computation but may introduce stitching artifacts at boundaries. Refinements like KiloNeRF [36], which divides scenes into thousands of compact neural networks, and Multiscale Implicit Neural Representation (MINER) [37], which processes images hierarchically, offer promising solutions. Additionally, hybrid frameworks like ACORN [38] dynamically allocate resources based on local signal complexity, optimizing both memory usage and training time for high-resolution applications.

Beyond computational improvements, extending INRID to related tasks such as super-resolution and inpainting is a natural progression, given the similar challenge of reconstructing missing data. Integrating conditioning mechanisms, such as activation function modulations [39] or meta-learning paradigms [40], could further enhance generalization across diverse images while reducing per-image training requirements. This is particularly relevant for refining state-of-the-art methods in scenarios with out-of-distribution data, such as blur or noise.

Theoretical advancements addressing spectral bias [41]—a tendency of MLPs to prioritize low-frequency components over high-frequency details—are also essential. A structured dictionary perspective [40], where MLPs learn representations from a set of predefined basis functions, offers a promising direction to improve high-frequency detail reconstruction and overall image fidelity.

Ultimately, while INRID achieves superior reconstruction quality, its computational demands highlight clear challenges and opportunities for future work. Advances in training efficiency, scalability, and generalization will be crucial in realizing the broader applicability of INRID across diverse image reconstruction tasks while preserving its fidelity.

6. Conclusion

This paper introduced INRID, a novel framework leveraging Implicit Neural Representations for image demosaicking. By integrating Bayer loss to enforce fidelity to sensor data and complementary loss to utilize initial reconstructions, INRID significantly enhances traditional methods like Malvar and Menon, achieving PSNR improvements of up to 2 dB. The framework also addresses limitations in deep learning-based methods, effectively correcting artifacts and demonstrating resilience in challenging scenarios, including blur and noise. Real-world validation on raw sensor data from mobile cameras further underscored INRID's capability to produce sharper and more accurate reconstructions compared to standard outputs and advanced pipelines like RSTCANet.

While INRID achieves state-of-the-art reconstruction fidelity, its computational demands highlight opportunities for further optimization. Future work will focus on improving efficiency through approaches such as multiresolution encoding and dictionary-based representations, and scaling to gigapixel images using advanced frameworks like MINER. Extending INRID to tasks like super-resolution and inpainting represents a promising direction, leveraging its capacity to adapt to diverse input characteristics while maintaining high fidelity.

In conclusion, INRID demonstrates the potential of implicit neural representations to not only improve demosaicking quality but also tackle joint problems such as denoising and deblurring, paving the way for their integration into advanced image reconstruction pipelines.

CRediT authorship contribution statement

Tomáš Kerepecký: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing. **Filip Šroubek:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Jan Flusser:**

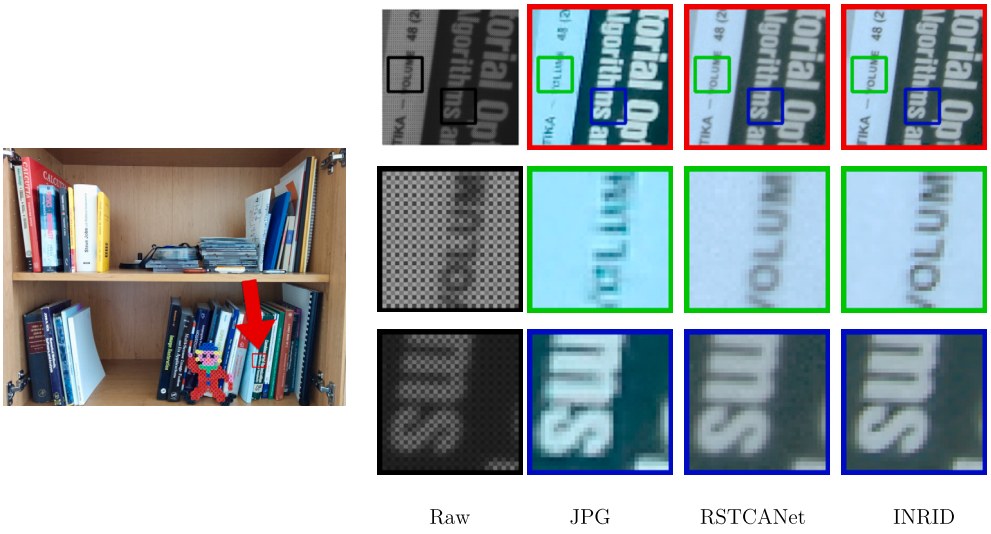


Fig. 12. Joint Demosaicking and Deblurring on real data from an LG Nexus 5 camera. The red arrow in the large image highlights the cropped area. The first column shows raw Bayer data, the second is the JPEG output, the third is the result from RSTCANet, and the fourth is INRID with initial RSTCANet estimate and deblurring using PSF priors. Green and blue borders in the first row show close-ups of specific regions.

Funding acquisition, Project administration, Supervision, Writing – review & editing.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used Grammarly and ChatGPT solely to improve the readability and language of the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Czech Science Foundation grant 25-15933S.

Data availability

The code repository and details, along with the publicly available Kodak and McM datasets used, are provided in the Experimental section with the GitHub link included in the manuscript.

References

[1] H.S. Malvar, L.-w. He, R. Cutler, High-quality linear interpolation for demosaicing of Bayer-patterned color images, in: 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 3, IEEE, 2004, pp. iii–485.
[2] D. Menon, S. Andriani, G. Calvagno, Demosaicing with directional filtering and a posteriori decision, IEEE Trans. Image Process. 16 (1) (2006) 132–141.
[3] K. Hirakawa, T.W. Parks, Joint demosaicing and denoising, IEEE Trans. Image Process. 15 (8) (2006) 2146–2157.
[4] C.J. Schuler, M. Hirsch, S. Harmeling, B. Schölkopf, Non-stationary correction of optical aberrations, in: 2011 International Conference on Computer Vision (CVPR), IEEE, 2011, pp. 659–666.
[5] L. Condat, S. Mosaddegh, Joint demosaicking and denoising by total variation minimization, in: 2012 19th IEEE International Conference on Image Processing (ICIP), IEEE, 2012, pp. 2781–2784.
[6] H.Q. Luong, B. Goossens, J. Aelterman, A. Pižurica, W. Philips, A primal-dual algorithm for joint demosaicking and deconvolution, in: 2012 19th IEEE International Conference on Image Processing (ICIP), IEEE, 2012, pp. 2801–2804.
[7] F. Heide, M. Steinberger, Y.-T. Tsai, M. Rouf, D. Pajak, D. Reddy, O. Gallo, J. Liu, W. Heidrich, K. Egiazarian, et al., Flexisp: a flexible camera image processing framework, ACM Trans. Graph. 33 (6) (2014) 1–13.
[8] D.S. Yoo, M.K. Park, M.G. Kang, Joint deblurring and demosaicing using edge information from Bayer images, IEICE Trans. Inf. Syst. 97 (7) (2014) 1872–1884.
[9] T. Kerepecky, F. Šroubek, D3net: joint demosaicking, deblurring and deringing, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 1–8.
[10] F. Kokkinos, S. Lefkimmiatis, Deep image demosaicking using a cascade of convolutional residual denoising networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 303–319.
[11] W. Xing, K. Egiazarian, Residual swin transformer channel attention network for image demosaicing, in: 2022 10th European Workshop on Visual Information Processing (EUVIP), IEEE, 2022, pp. 1–6.
[12] M. Gharbi, G. Chaurasia, S. Paris, F. Durand, Deep joint demosaicking and denoising, ACM Trans. Graph. 35 (6) (2016) 1–12.
[13] D.S. Tan, W.-Y. Chen, K.-L. Hua, Deepdemosaicking: adaptive image demosaicking via multiple deep fully convolutional networks, IEEE Trans. Image Process. 27 (5) (2018) 2408–2419.
[14] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, R. Timofte, Plug-and-play image restoration with deep denoiser prior, IEEE Trans. Pattern Anal. Mach. Intell. 44 (10) (2021) 6360–6376.
[15] Y. Zhang, K. Li, B. Zhong, Y. Fu, Residual non-local attention networks for image restoration, in: International Conference on Learning Representations (ICLR), 2019.
[16] Y. Xie, T. Takikawa, S. Saito, O. Litany, S. Yan, N. Khan, F. Tombari, J. Tompkin, V. Sitzmann, S. Sridhar, Neural fields in visual computing and beyond, in: Computer Graphics Forum, vol. 41, Wiley Online Library, 2022, pp. 641–676.
[17] X. Li, B. Gunturk, L. Zhang, Image demosaicing: a systematic survey, in: Visual Communications and Image Processing 2008, vol. 6822, SPIE, 2008, pp. 489–503.
[18] S. Farsiu, M. Elad, P. Milanfar, Multiframe demosaicing and super-resolution of color images, IEEE Trans. Image Process. 15 (1) (2005) 141–159.
[19] J.J. Park, P. Florence, J. Straub, R. Newcombe, S. Lovegrove, Deepsdf: learning continuous signed distance functions for shape representation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 165–174.
[20] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, R. Ng, Fourier features let networks learn high frequency functions in low dimensional domains, Adv. Neural Inf. Process. Syst. 33 (2020) 7537–7547.
[21] B. Mildenhall, P.P. Srinivasan, M. Tancik, J.T. Barron, R. Ramamoorthi, R. Ng, Nerf: representing scenes as neural radiance fields for view synthesis, Commun. ACM 65 (1) (2021) 99–106.
[22] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, G. Wetzstein, Implicit neural representations with periodic activation functions, Adv. Neural Inf. Process. Syst. 33 (2020) 7462–7473.
[23] V. Saragadam, D. LeJeune, J. Tan, G. Balakrishnan, A. Veeraraghavan, R.G. Baraniuk, Wire: wavelet implicit neural representations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 18507–18516.

[24] A. Kazerouni, R. Azad, A. Hosseini, D. Merhof, U. Bagci, Incode: implicit neural conditioning with prior knowledge embeddings, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2024, pp. 1298–1307.
[25] S. Ramasinghe, S. Lucey, Beyond periodicity: towards a unifying framework for activations in coordinate-mlps, in: European Conference on Computer Vision (ECCV), Springer, 2022, pp. 142–158.
[26] T. Kerepecky, F. Sroubek, A. Novozamsky, J. Flusser, Nerd: neural field-based demosaicking, in: 2023 IEEE International Conference on Image Processing (ICIP), IEEE, 2023, pp. 1735–1739.
[27] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
[28] O. Ronneberger, P. Fischer, T. Brox, U-net: convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18, Springer, 2015, pp. 234–241.
[29] P. Blomgren, T.F. Chan, Color tv: total variation methods for restoration of vector-valued images, IEEE Trans. Image Process. 7 (3) (1998) 304–309.
[30] E. Kodak, Kodak lossless true color image suite (photocd pcd0992), http://r0k.us/graphics/kodak, 1993, 6.
[31] L. Zhang, X. Wu, A. Buades, X. Li, Color demosaicking by local directional interpolation and nonlocal adaptive thresholding, J. Electron. Imaging 20 (2) (2011) 023016.
[32] D. Geman, G. Reynolds, Constrained restoration and the recovery of discontinuities, IEEE Trans. Pattern Anal. Mach. Intell. 14 (03) (1992) 367–383.
[33] F. Šroubek, T. Kerepecky, J. Kamenicky, Iterative Wiener filtering for deconvolution with ringing artifact suppression, in: 2019 27th European Signal Processing Conference (EUSIPCO), IEEE, 2019, pp. 1–5.
[34] T. Müller, A. Evans, C. Schied, A. Keller, Instant neural graphics primitives with a multiresolution hash encoding, ACM Trans. Graph. 41 (4) (2022) 1–15.
[35] P. Wang, Z. Fan, T. Chen, Z. Wang, Neural implicit dictionary learning via mixture-of-expert training, in: International Conference on Machine Learning (ICML), in: PMLR, 2022, pp. 22613–22624.
[36] C. Reiser, S. Peng, Y. Liao, A. Geiger, Kilonerf: speeding up neural radiance fields with thousands of tiny mlps, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR), 2021, pp. 14335–14345.
[37] V. Saragadam, J. Tan, G. Balakrishnan, R.G. Baraniuk, A. Veeraraghavan, Miner: multiscale implicit neural representation, in: European Conference on Computer Vision (ECCV), Springer, 2022, pp. 318–333.
[38] J.N. Martel, D.B. Lindell, C.Z. Lin, E.R. Chan, M. Monteiro, G. Wetzstein, Acorn: adaptive coordinate networks for neural scene representation, ACM Trans. Graph. 40 (4) (2021) 1–13.
[39] E. Dupont, H. Kim, S.A. Eslami, D.J. Rezende, D. Rosenbaum, From data to functa: your data point is a function and you can treat it like one, in: International Conference on Machine Learning (ICML), in: PMLR, 2022, pp. 5694–5725.
[40] G. Yüce, G. Ortiz-Jiménez, B. Besbinar, P. Frossard, A structured dictionary perspective on implicit neural representations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 19228–19238.
[41] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, A. Courville, On the spectral bias of neural networks, in: International Conference on Machine Learning (ICML), in: PMLR, 2019, pp. 5301–5310.

Tomáš Kerepecký earned his M.Sc. degree in Computational Physics from the Faculty of Nuclear Sciences and Physical Engineering at Czech Technical University in Prague in 2017. From 2017 to 2019, he worked as a Junior Researcher at ELI Beamlines in Dolní Břežany, Czechia, focusing on computational simulations in laser physics. He is currently pursuing a Ph.D. in Image Processing at the Institute of Information Theory and Automation, in collaboration with Czech Technical University, with research centered on inverse problems, particularly demosaicing and deconvolution in digital photography. He is also pursuing an M.A. in Leadership and Practical Theology at TCM International Institute, Austria. In 2021-2022, he was a Fulbright Visiting Scholar at Washington University in St. Louis, where he advanced his research in implicit neural representations.

Filip Šroubek received the M.Sc. degree in computer science from the Czech Technical University, Prague, Czech Republic in 1998 and the Ph.D. degree in computer science from Charles University, Prague, Czech Republic in 2003. From 2004 to 2006, he was on a postdoctoral position in the Instituto de Optica, CSIC, Madrid, Spain. In 2010/2011 he received a Fulbright Visiting Scholarship at the University of California, Santa Cruz. In 2014, he became a research professor in Physico-Mathematical Sciences (Informatics and Cybernetics) at the Czech Academy of Sciences. In 2016, he became an associate professor at the Faculty of Mathematics and Physics, Charles University. Currently he is the deputy head of the Department of Image Processing.

Jan Flusser received the M.Sc. degree in mathematical engineering from the Czech Technical University, Prague, Czech Republic, in 1985; the PhD degree in computer science from the Czechoslovak Academy of Sciences in 1990; and the DrSc. degree in technical cybernetics in 2001. Since 1985 he has been with the Institute of Information Theory and Automation, Czech Academy of Sciences, Prague. In 1995–2007, he was holding the position of a head of Department of Image Processing. In 2007–2017, he was a Director of the Institute. Since 2017 he has been at the position of Research Director. He is a full professor of computer science at the Czech Technical University, Faculty of Nuclear Science and Physical Engineering, and at the Charles University, Faculty of Mathematics and Physics, Prague, Czech Republic, where he gives undergraduate and graduate courses on Digital Image Processing, Pattern Recognition, and Moment Invariants and Wavelets. Jan Flusser's research interest covers moments and moment invariants, image registration, image fusion, multichannel blind deconvolution and super-resolution imaging. He has authored and coauthored more than 200 research publications in these areas, including the monographs Moments and Moment Invariants in Pattern Recognition (Wiley, 2009) and 2D and 3D Image Analysis by Moments (Wiley, 2016). In 2010, Jan Flusser was awarded by the SCOPUS 1000 Award. He received the Felber Medal of the Czech Technical University for excellent contribution to research and education in 2015 and the Praemium Academiae of the Czech Academy of Sciences for outstanding researchers in 2017.