LoFi: Neural Local Fields for Scalable Image Reconstruction

AmirEhsan Khorashadizadeh[ID], Tobías I. Liaudat[ID], Tianlin Liu[ID], Jason D. McEwen[ID], and Ivan Dokmanić[ID]

Abstract—**Neural fields or implicit neural representations (INRs) have attracted significant attention in computer vision and imaging due to their efficient coordinate-based representation of images and 3D volumes. In this work, we introduce a coordinate-based framework for solving imaging inverse problems, termed LoFi (Local Field). Unlike conventional methods for image reconstruction, LoFi processes local information at each coordinate *separately* by multi-layer perceptrons (MLPs), recovering the object at that specific coordinate. Similar to INRs, LoFi can recover images at any continuous coordinate, enabling image reconstruction at multiple resolutions. With comparable or better performance than standard deep learning models like convolutional neural networks (CNNs) and vision transformers (ViTs), LoFi achieves excellent generalization to out-of-distribution data with memory usage almost independent of image resolution. Remarkably, training on 1024×1024 images requires less than 200MB of memory—much below standard CNNs and ViTs. Additionally, LoFi's local design allows it to train on extremely small datasets with 10 samples or fewer, without overfitting and without the need for explicit regularization or early stopping.**

I. INTRODUCTION

IMAGING inverse problems are ubiquitous in domains like medicine [1], material science [2], remote sensing [3] and cosmology [4]. Deep learning is often the method of choice for solving inverse problems. In particular, convolutional neural networks (CNNs) like U-Net [5] have shown remarkable performance across diverse tasks including computed tomography (CT) [6], magnetic resonance imaging (MRI) [7], radio interferometric imaging [8] and dark matter mapping [9] in cosmology. The success of the U-Net is primarily attributed to its multiscale architecture with a large receptive field that extracts features from the input image at different scales [10].

Despite their strong performance on low-dimensional 2D imaging problems, deep learning architectures can become computationally expensive for large images [11]–[14]. This inefficiency arises because current deep neural networks reconstruct the entire image simultaneously, necessitating substantial memory for back-propagation during training, especially for large images. Additionally, interpreting such complex architectures [12], [15], [16] and analyzing their reconstructions can be challenging. Simplified neural architectures can enhance our understanding of reconstruction mechanisms, enabling the design of robust models with strong generalization [17].

Recently, implicit neural representations (INRs) [18]–[20] have emerged as a promising coordinate-based pipeline for representing continuous signals, images, and 3D volumes. Unlike most existing deep learning models that treat signals as discrete arrays, INRs map coordinates to signal values using a deep neural network, typically a multi-layer perceptron (MLP), resulting in a *continuous* signal representation. INRs have several advantages over standard deep learning models. Rather than representing signals at a single resolution, INRs can conveniently interpolate signals in a continuous space. This coordinate-based representation is particularly interesting because we can adjust the required memory making INRs well-suited for high-dimensional 3D reconstructions [21]–[26] and scene representation [27].

In this paper, we introduce a scalable coordinate-based local reconstruction pipeline for solving imaging inverse problems. In many tasks such as image denoising and low-dose computed tomography (LDCT), the image intensity at a specific pixel is primarily influenced by the observed image in the pixel's neighborhood [28]. Our proposed model, termed LoFi (Local Field), recovers the image intensity at each pixel *separately*, using local information extracted from the input image around that pixel. This information is processed by a neural network (composed of MLP blocks) to determine the image intensity at the target pixel, enabling image reconstruction at any resolution or arbitrary continuous coordinate.

LoFi's coordinate-based design brings several advantages. Unlike standard CNNs and ViTs which requires substantial memory during training, we can train LoFi, Similar to INRs, on mini-batches of both objects and pixels. This results in resolution-agnostic memory usage, making it highly efficient for high-resolution image processing. As illustrated in Figure 1, LoFi demonstrates significantly lower memory and time requirements compared to CNNs and ViTs. For instance, training LoFi on 1024×1024 images requires less than 200MB memory. Furthermore, its local design provides a strong inductive bias for image reconstruction; unlike CNNs and ViTs, which tend to overfit when trained on tiny datasets with very few

(a) Memory footprint (batch size 1)
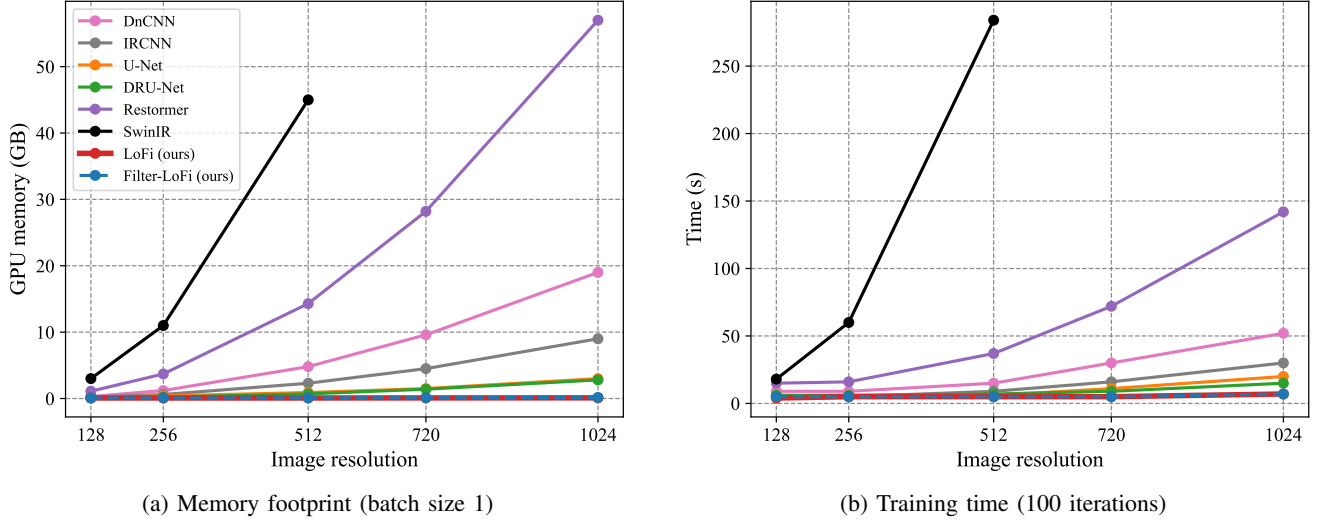
(b) Training time (100 iterations)

Fig. 1: The memory and time requirements during training for different models; LoFi is significantly faster and more memory-efficient than CNNs and ViTs. Notably, LoFi's memory usage remains almost independent of image resolution, making it an ideal choice for high-dimensional image reconstruction. All experiments are conducted using a single A100 GPU with 80GB memory. Missing data points indicate that the corresponding model exceeds the GPU memory capacity for the given resolution.

samples, LoFi consistently shows robust performance without overfitting, eliminating the need for regularization or early stopping.

In addition, deep learning models for image reconstruction like multiscale CNNs [5], [15] and vision transformers (ViTs) [12], [16], [29], [30] often use sophisticated architectures that complicate downstream interpretations. Through our proposed coordinate-conditioned patch geometry, LoFi can learn the position of the relevant features in the input image for each pixel, providing insights for downstream image analysis and interpretation.

II. BACKGROUND AND RELATED WORKS

The goal of an imaging inverse problem is to recover the image of interest $\mathbf{f}$ from noisy measurements $\mathbf{q}$ acquired via the potentially non-linear forward operator $\mathbf{A}$,

$$\mathbf{q} = \mathbf{A}(\mathbf{f}) + \mathbf{n}, \tag{1}$$

where $\mathbf{n}$ is noise. Inverse problems are often ill-posed, implying the existence of multiple plausible reconstructions for a given set of measurements. It is thus important to incorporate an image prior to enable stable and accurate reconstruction. While traditional hand-crafted priors and regularizers such as total variation (TV) [31] can yield good reconstructions, learning-based priors have brought about substantial improvements [32]–[35].

A. Inverse problems with local operators

In many inverse problems, the forward operator is local; an image pixel primarily relates to its neighboring pixels in the observed image. Here the assumption is that the observed data is shaped like an image with the same coordinate system as

the target. A local forward operator suggests an opportunity to design efficient deep-learning architectures. In the following, we discuss three inverse problems characterized by this localization property.

a) Image denoising: The most straightforward inverse problem is image denoising, where the forward operator $\mathbf{A}$ in (1) is the identity matrix; each pixel in the clean image is solely connected to the corresponding pixel in the noisy image, assuming independent noise (between pixels) and natural images [28]. Despite this simplicity, the presence of noise prompts the incorporation of 'contextual' information from neighboring noisy pixels. In principle, the required local neighborhood size depends on the noise level and the prior distribution; stronger noise usually calls for larger receptive fields [28], [36].

b) Low-dose computed tomography: Low-dose computed tomography (LDCT) is a medical imaging technique that reduces radiation exposure to patients during a CT scan by lowering the X-ray beam intensity [37]. Traditional CT imaging uses X-rays to capture cross-sectional images of the body, but these scans can result in significant radiation doses, potentially increasing the risk of long-term side effects like cancer [38], [39]. LDCT mitigates these risks by significantly lowering the X-ray dose without compromising diagnostic quality, making it a safer imaging process.

CT image reconstruction can be viewed as an inverse problem,

$$\mathbf{q} = \mathcal{R}(\mathbf{f}) + \mathbf{n}, \tag{2}$$

where the forward operator $\mathcal{R}$ is the radon transform [40] applied in angles $\{\alpha_i\}_{i=1}^{r}$ and $\mathbf{n}$ is the additive noise. In LDCT, the projections $\mathbf{y}$ are collected from sufficient angles r but using low-intensity beams resulting in noisy projections.
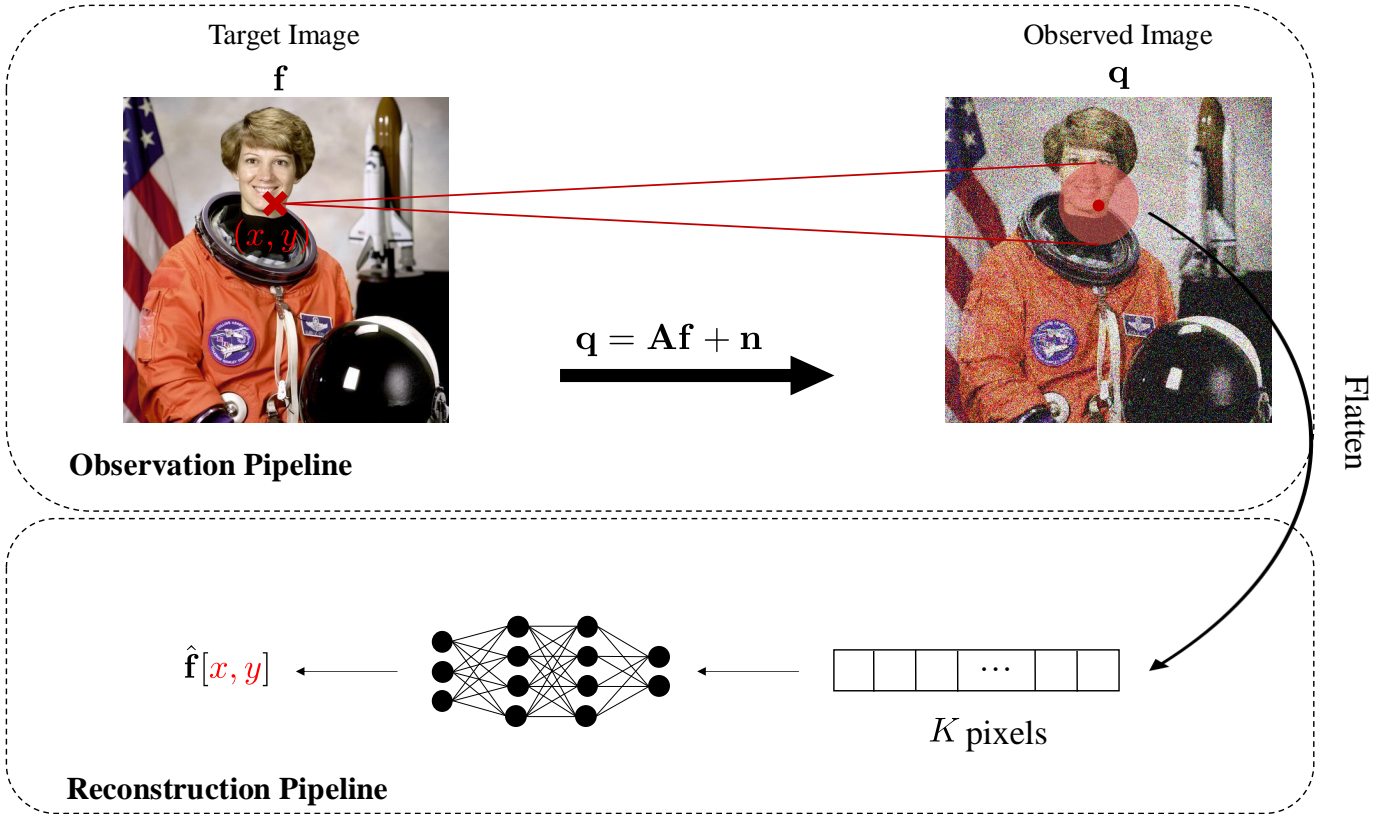
Fig. 2: LoFi; the neural network NN_θ, typically composed of MLP modules, processes the local information extracted from the observed image around the given pixel (x, y). LoFi's inductive bias for image reconstruction brings strong generalization on OOD data, it requires less training data and uses small memory when training on high-resolution images.

The standard approach for CT image reconstruction is filtered backprojection (FBP),

$$\mathbf{f}_{\mathrm{FBP}} = \mathcal{R}_{\mathrm{FBP}}(\mathbf{q}). \qquad (3)$$

In an ideal scenario where we have an infinite number of noise-free projections, FBP is the exact inverse, namely, $\mathbf{f}_{\mathrm{FBP}} = \mathbf{f}$. In LDCT, we have a sufficient number of noisy projections N, the image reconstruction from FBP can be seen as a denoising problem,

$$\mathbf{f}_{\mathrm{FBP}} \approx \mathbf{f} + \mathcal{R}_{\mathrm{FBP}}(\mathbf{n}). \qquad (4)$$

The image reconstruction from FBP can be seen as an inverse problem with a local forward operator for LDCT. For a comprehensive evaluation of current LDCT image reconstruction methods, we refer to [41], where the authors conducted an in-depth analysis of the performance of CNNs, using both simulated and real experimental data for training and inference.

c) Dark matter mapping in cosmology: As another example, we consider mapping the dark matter distribution of the Universe [4] by solving a weak gravitational lensing [42] inverse problem. We will study the convergence that traces the projected mass distribution of dark matter in the Universe. The goal is to estimate the target convergence field κ that traces the projected mass distribution, from a related observable, the shear field γ. Higher order statistical properties of the convergence

field are highly informative for studying the nature of dark energy and dark matter. The two fields are related as,

$$\gamma = \mathbf{A}\kappa + \mathbf{n}, \qquad (5)$$

where $\mathbf{n}$ is the shear field noise, and the forward operator $\mathbf{A}$ is a convolutional filter (a Fourier multiplier) with the 2D Fourier transform given by,[1]

$$\mathbf{D}_{k_x, k_y} = \frac{k_x^2 - k_y^2 + 2ik_x k_y}{k_x^2 + k_y^2}, \quad \text{for } k_x^2 + k_y^2 \neq 0. \qquad (6)$$

A simple approach to recover the convergence field κ from the shear field γ is the Kaiser-Squires (KS) method [4], the current standard in cosmology. The method consists of directly applying $\mathbf{A}^{-1}$, the inverse of the forward operator, to the shear field, followed by a subsequent Gaussian smoothing with an ad hoc smoothing scale. We consider the naive KS inversion, without the Gaussian smoothing step, as follows,

$$\kappa_{\mathrm{KS}} = \mathbf{A}^{-1}\mathbf{A}\gamma = \kappa + \mathbf{A}^{-1}\mathbf{A}\mathbf{n} = \kappa + \tilde{\mathbf{n}}, \qquad (7)$$

where $\tilde{\mathbf{n}} = \mathbf{A}^{-1}\mathbf{A}\mathbf{n}$. The recovery of the convergence field from the naive inverted KS image is thus an inverse problem with a local forward operator. We note that the processed noise $\tilde{\mathbf{n}}$ is now spatially correlated and modeling it may still benefit

---

[1]This is known as the weak lensing planar forward model. Note that $\mathbf{D}_{k_x, k_y}$ is undefined for $(k_x, k_y) = (0, 0)$, meaning that the observable γ cannot constrain the average of κ. This is known as the mass sheet degeneracy. The choice $\mathbf{D}_{0,0,} = 1$ is made for convenience and is secondary to what follows.

from a larger receptive field; this is addressed by the learnable noise suppression filter described in Section III-D.

B. Deep learning for image reconstruction

Over the last decade, various deep learning approaches have achieved state-of-the-art (SOTA) performance in image reconstruction. Some early approaches segmented the image into patches and employed an MLP to denoise each patch individually [43]. However, the optimal patch size for high-quality reconstructions depends on the noise level and data distribution [28]. Zhang et al. [11] introduced DnCNN, an image-to-image CNN designed for image denoising where the network depth is adjusted based on the required receptive field size suggested by traditional methods [44]. To expand CNN's receptive field and use more contextual information, they used dilated convolutions [45]. Concurrently, the U-Net [5] has been the backbone model for SOTA image reconstruction architectures [10], [46]–[48], in particular, the DRU-Net [15] achieved remarkable performance by integrating residual blocks [49] into the U-Net architecture.

More recently, vision transformers (ViTs) [50] have achieved comparable or even better performance than CNNs for various image reconstruction tasks [12], [29], [51], [52]. The core concept of ViTs is to divide the input image into non-overlapping patches, transform each patch into an embedding using a learned linear projection, and then process the resulting tokens through a series of stacked self-attention layers and MLP blocks. The advantage of ViTs over CNNs is attributed to their ability to capture long-range dependencies in the images through the self-attention mechanism [12], [30]. As shown in Figure 1, a major challenge for ViTs in image reconstruction is the quadratic computational complexity relative to the input image size [13], [53] which makes ViTs impractical for high dimensional image reconstruction [14]. Recent works attempted to reduce the computational complexity of vision transformers. Restormer [53] applies the self-attention along the channel dimension to reduce memory footprint. Some approaches design hierarchical transformers [13], [54]. The authors of [16] use convolutional blocks to reduce the spatial dimensions followed by self-attention blocks applied in different scales. Despite the marginally better reconstructions compared to CNNs for inverse problems with non-local forward operators like magnetic resonance imaging (MRI), they show comparable performance with CNNs for local problems like image denoising [16].

In this paper, we demonstrate that for inverse problems with local forward operators, such as those discussed in Section II-A, it is not necessary to process large-scale features using multiscale CNNs or ViTs. Instead, our proposed architecture efficiently captures local features, resulting in strong generalization while significantly reducing computational costs and memory requirements, well-suited for high-dimensional image reconstruction.

C. Neural fields for inverse problems

INRs, particularly those with sinusoidal activations, efficiently model signals and their spatial derivatives. This is particularly useful for solving partial differential equations (PDEs) [18], [55]. In the domain of continuous super-resolution, INRs have been effectively combined with CNNs. For instance, the authors of [56] proposed conditioning the INR's generation process on local features extracted by a CNN from the low-resolution image. Closer to our work, the authors of [57] used a similar approach, where local features extracted from a low-resolution image are processed with a lightweight CNN to enable continuous super-resolution.

D. Patch-based image reconstruction

While traditional methods for image reconstruction often process small patches extracted from the input image [58], [59], deep learning models with patch-based design have recently gained considerable attention. ViTs [50] and MLP-mixers [60] are a well-known family of patch-based models whose strong performance is often attributed to patch-based design [61]. Deep generative models trained on small patches have been used for high dimensional image generation [62] and more efficient training on datasets with significantly fewer images [63]. Patch priors [64]–[67] as efficient alternatives to image priors have been used for image reconstruction which allows training on small datasets with few images well-suited for scientific applications where we have access to a limited number of ground truth images.

III. LoFi: A LOCAL FIELD PROCESSOR

Our objective is to reconstruct the target image $\mathbf{f} \in \mathbb{R}^{N \times N \times C}$ from the observed image $\mathbf{q} \in \mathbb{R}^{M \times M \times C}$, as described in the forward equation (1), where C is the channel dimension, N and M denote the resolution of the target and observed images. The resolutions of $\mathbf{f}$ and $\mathbf{q}$ may be different but we assume that they share the same coordinate system and semantics; thus $\mathbf{f}[x, y]$ and $\mathbf{q}[x, y]$ refer to the same continuous location in both images. Example problems where this is meaningful are denoising and deblurring. More generally, for many linear inverse problems, including computed tomography, it will hold after applying the adjoint of the forward operator as shown in (4).

We then consider a local forward operator $\mathbf{A}$ where the intensity $\mathbf{f}[x, y]$ at pixel $(x, y) \in \mathbb{R}^2$ relates only to a small neighborhood of (x, y) in $\mathbf{q}$. Based on this locality hypothesis, we design a coordinate-based neural network inspired by implicit neural representations.

To recover the image (only) at the target pixel (x, y), we extract a neighborhood of K pixels around (x, y) from the observed image $\mathbf{q}$. We denote the (flattened) extracted pixels by $\mathbf{p}_{x,y} \in \mathbb{R}^{K \times C}$. As illustrated in Figure 2, we process $\mathbf{p}_{x,y}$ using a neural network $\text{NN}_\theta : \mathbb{R}^{K \times C} \to \mathbb{R}^C$ with parameters θ to approximate the image intensity $\mathbf{f}[x, y]$,

$$\hat{\mathbf{f}}[x, y] = \text{NN}_\theta(\mathbf{p}_{x,y}). \tag{8}$$

This coordinate-based image representation enables image recovery at any continuous coordinate with small memory. We call our framework LoFi (Local Field). As we typically use a small neighborhood size K, we can parametrize NN_θ using an MLP. In the following sections, we provide further
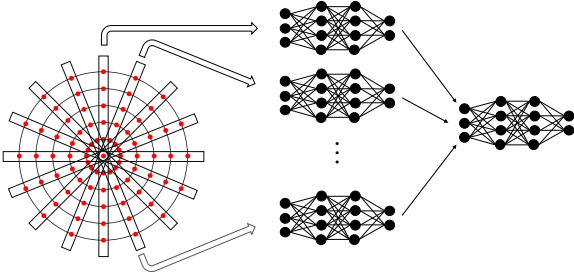
Fig. 3: MultiMLP architecture; the input information is split into smaller chunks each processed with a separate MLP, the extracted information is then mixed by another MLP.

details regarding the LoFi architecture and introduce the pre-processing filter, patch geometry, and training strategy.

A. MultiMLPs for large neighborhoods

One important factor in LoFi architecture is the neighborhood dimension K; larger K can provide more detailed information to improve reconstructions. However, the size of the MLP quadratically scales with K resulting in large networks and slow training. To address this issue, we propose MultiMLP, an alternative architecture to handle large neighborhoods. Inspired by vision transformers [50], we split the extracted neighboring pixels (here over a circular geometry) into smaller chunks, each processed by a separate MLP as shown in Figure 3. Their outputs are then mixed by another MLP.

B. Patch geometry and equivariance

A straightforward method to extract the neighboring pixels is to directly select the pixels adjacent to the target pixel (x, y). This however has several drawbacks: it requires a manual selection of the neighborhood size, akin to traditional methods, which is hard to estimate; it only permits image evaluation at on-grid pixels resulting in a single-resolution image reconstruction.

1) Differentiable patch extraction: While we could use the standard square patch to extract the local information, this would not result in a differentiable map from the input coordinate to the pixel intensity. To enable image reconstruction at arbitrary *continuous* coordinate, we extract $\mathbf{p}_{x,y}$ using bicubic interpolation which results in a differentiable architecture.

2) Learnable patch geometry: This differentiable patch extraction opens up the possibility to learn the patch geometry centered around the target coordinate (x, y) by first defining a set of *learnable* coordinate offsets,

$$\Delta I = \left[(x_n^\Delta, y_{n'}^\Delta) \right]_{n,n'=1}^K. \tag{9}$$

By learning the optimal coordinate offsets ΔI we can also control the scale and shape of the patch receptive field. We then extract the patch by evaluating $\mathbf{q}$ at coordinates $I_{(x,y)} = (x, y) + \Delta I$ to get

$$\mathbf{p}_{x,y} = \mathbf{q}[I_{(x,y)}]. \tag{10}$$

We emphasize that the coordinates $I_{(x,y)}$ need not be aligned with the pixel grid as we use differentiable bicubic interpolation to enable off-the-grid evaluation. Unlike most other work using

square patches, we initialize ΔI to a circular geometry; this results in better rotation invariance as discussed in Section VIII-B in the supplementary materials.

C. Coordinate-conditioned patch geometry

The learned patch geometry in (9) is fixed. In practice, the relative locations of the most relevant information changes for different pixels. To enable coordinate-conditioned patch geometry (CCPG), we use another neural network (typically a simple MLP) $\text{CCPG}_\psi : \mathbb{R}^{K \times C} \to \mathbb{R}^{2K}$ which takes a local patch around pixel (x, y) and estimates the position of the coordinates inside the patch as follows,

$$\Delta I_{CCPG}(x, y) = \text{CCPG}_\psi(p_{x,y}). \tag{11}$$

We then evaluate $\mathbf{q}$ at $(x, y) + \Delta I_{CCPG}(x, y)$. The learned patch geometry $\Delta I_{CCPG}(x, y)$ can localize the position of the relevant information for image recovery at pixel (x, y) for corrupted image $\mathbf{q}$. A similar strategy has been developed by [68] for object detection using CNNs.

Now we can improve the localization performance by repeating this process T times,

$$\Delta I_{CCPG}^{(i)}(x, y) = \text{CCPG}_\psi^{(i)}(p_{x,y}^{(i-1)}) \tag{12}$$

$$p_{x,y}^{(i)} = \mathbf{q}[(x, y) + \Delta I_{CCPG}^{(i)}(x, y)] \tag{13}$$

where in each iteration $1 \le i \le T$, we feed noisy image $\mathbf{q}$ evaluated at the estimated patch position in the previous step $p_{x,y}^{(i-1)}$ to a separate neural network $\text{CCPG}_\psi^{(i)}$. As we will show in our experiments in Section V-C, the learned patch geometry indeed improves over iterations.

D. Noise suppression filter

The observed image $\mathbf{q}$ is corrupted with noise. Optimal filters for correlated noise can have a large receptive field which cannot be implemented by the localized NN_θ in (8). To address this, we apply pre-processing convolutional filters $\mathbf{h} \in \mathbb{R}^{s \times s \times L}$ to the observed image before patch extraction,

$$\tilde{\mathbf{q}} = \mathbf{q} \star \mathbf{h}, \tag{14}$$

where $\star$ denotes the convolution operator, s is the filter size and L is the number of filters. This process is reminiscent of filtering techniques commonly used in computed tomography imaging [69]. We can also concatenate the filtered and noisy images along the channel dimension.

To avoid manual adjustment of the receptive field, we apply the filter in the frequency domain,

$$\tilde{\mathbf{q}} = \mathbf{F}^{-1} \mathbf{H} \mathbf{F} \mathbf{q}, \tag{15}$$

where $\mathbf{F}$ and $\mathbf{F}^{-1}$ are the forward and inverse 2D Fourier transforms, and the diagonal matrix $\mathbf{H}$ is a learnable parameter fitted to the data. The resulting filter can have large support in image space, capturing global information. We initialize the filter $\mathbf{H}$ with all ones in the Fourier domain (an identity filter) which corresponds to the smallest support in pixel space. During training, the filter expands its support based on the image resolution, noise level, and data distribution.
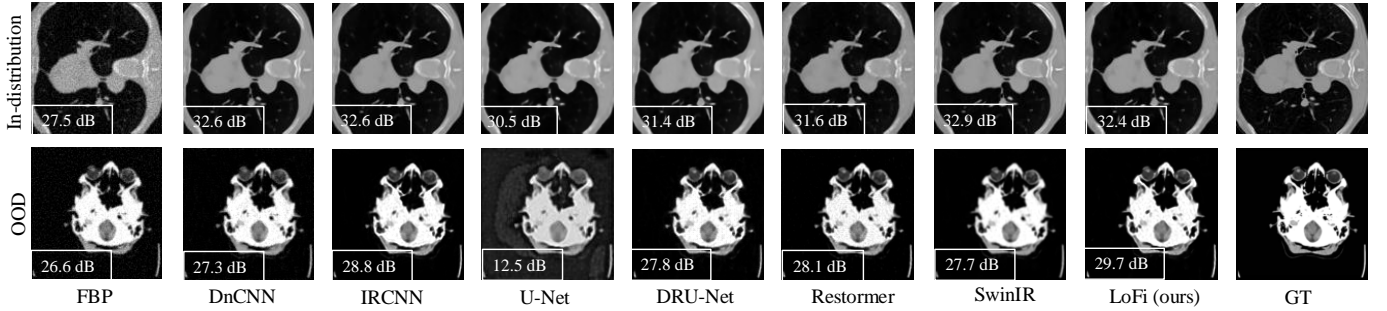
Fig. 4: Performance comparison on LDCT (30dB noise) at resolution 128×128 for in-distribution (chest) and OOD (brain) data.

E. Resolution-agnostic memory usage in training

For simplicity, we write the entire LoFi pipeline as $\hat{\mathbf{f}}(\mathbf{x}) = \text{LoFi}_\phi(\mathbf{x}, \mathbf{q})$ where $\mathbf{x} = (x, y)$ is the target pixel. The model approximates the image intensity $\hat{\mathbf{f}}(\mathbf{x})$ from the observed image $\mathbf{q}$. We denote all the trainable parameters of LoFi by ϕ. This includes θ, pre-processing filter parameters $\mathbf{h}$, and CCPG weights ψ. We consider a set of training data $\{(\mathbf{q}_i, \mathbf{f}_i)\}_{i=1}^{D}$ from observed and target images. We optimize the LoFi parameters ϕ using a gradient-based optimizer to solve,

$$\phi^\star = \underset{\phi}{\arg\min} \sum_{i=1}^{D} \sum_{j=1}^{N^2} |\mathbf{f}_i(\mathbf{x}_j) - \text{LoFi}_\phi(\mathbf{x}_j, \mathbf{q}_i)|. \qquad (16)$$

At inference time, we can recover the image at any target pixel $\mathbf{x}$ from the observed image $\mathbf{q}_{\text{test}}$ as $\hat{\mathbf{f}}_{\text{test}}(\mathbf{x}) = \text{LoFi}_{\phi^\star}(\mathbf{x}, \mathbf{q}_{\text{test}})$. From (16), it is clear that LoFi can be trained on mini-batches of both objects and pixels, enabling an almost constant-in-resolution memory usage during training and inference. This flexibility also allows us to train LoFi on datasets with images in various resolutions.

IV. FURTHER EXTENSIONS AND APPLICATIONS

In this section, we discuss further applications of the proposed local processing framework.

A. Learning the imaging geometry

Many seemingly complex image-to-image transforms, like those involving translations and rotations, alter image content using simpler coordinate transformations. For example, transposing a 2D image causes significant pixel differences in the high-dimensional pixel space; but the transformation relies on a simple, low-dimensional coordinate swap. When CNNs are fitted on image-to-image tasks, they are directly trained to model the changes in pixel values and, therefore, tend to overlook the coordinate transformations that underlie the pixel value changes. This is particularly problematic for large-scale changes in the coordinate system as the CNN should have a large receptive field to capture these geometric changes. This limitation has been explored by existing works to build deep neural networks that can model the changes in the coordinate system as well as the image content. For example, the authors of [70] separately model changes of the coordinate system and the image content using CNNs.

Although LoFi can only learn the local changes in the *image content*, its coordinate-based design allows us to easily equip it to learn the changes in the coordinate system as well. To this end, we use an INR_η with parameters η, that takes the given coordinate $\mathbf{x}$ and returns the new coordinate $\tilde{\mathbf{x}}$. This INR is then followed by LoFi to sample from the local neighborhood around $\tilde{\mathbf{x}}$ from the input image $\mathbf{q}$ as follows,

$$\tilde{\mathbf{x}} = \text{INR}_\eta(\mathbf{x}), \qquad (17)$$
$$\hat{\mathbf{f}}(\mathbf{x}) = \text{LoFi}_\phi(\tilde{\mathbf{x}}, \mathbf{q}). \qquad (18)$$

We train the above architecture end-to-end where the task of INR is to learn the changes in the coordinate system, while LoFi learns the changes in the image content from the corrected coordinate system. We call this new architecture INR-LoFi and compare its performance with CNNs for a toy experiment in Section V-D.

B. Low-resolution priors for high-resolution unsupervised image reconstruction

As shown in Section III-E, the proposed LoFi is a supervised learning framework and requires paired training data of measurements and corresponding target images. This is hard to obtain in many real scientific problems [6], [71]–[73]. Additionally, supervised models require re-training even for small changes in the forward model.

To address this issue, unsupervised methods learn a data-driven prior from only target images. Once trained, they can be used as a prior for solving any inverse problem using an iterative process. Unsupervised methods based on deep generative models [32], [74]–[77] and plug-and-play (PnP) [78]–[80] show strong performance on various image reconstruction tasks. We provide a brief overview of the PnP-ADMM framework in Section VII-E of the supplementary material.

In this section, we use a pre-trained LoFi denoiser as a prior in the plug-and-play (PnP) framework. This integration allows us to use a LoFi denoiser trained on low-resolution images to solve inverse problems at any higher resolution. Since we use the PnP-ADMM backbone we call the resulting scheme LoFi-ADMM (cf. Algorithm 1). Even though the LoFi denoiser is trained on low-resolution images, LoFi-ADMM can be applied at arbitrary higher resolutions thanks to LoFi's continuous image representation.

Algorithm 1: LoFi-ADMM

Input: q, A, ϕ^*
Parameter: α

$\mathbf{f} = 0,\ \mathbf{u} = 0,\ \mathbf{v} = 0$;
for $k = 0$ **to** $K - 1$ **do**
  $\mathbf{f}_k \leftarrow h(\mathbf{v}_{k-1} - \mathbf{u}_{k-1}; \alpha)$;
  $\mathbf{v}_k(\mathbf{x}) \leftarrow \text{LoFi}_{\phi^*}(\mathbf{x}, \mathbf{f}_k - \mathbf{u}_{k-1}), \quad \forall \mathbf{x} \in \mathbf{v}_k$;
  $\mathbf{u}_k \leftarrow \mathbf{u}_{k-1} + (\mathbf{f}_k - \mathbf{v}_k)$;
end

V. EXPERIMENTS

We now present a suite of experimental results which illustrate the various aspects of LoFi. In Section V-A, we compare LoFi with successful CNNs and ViTs on several image reconstruction tasks. We compare LoFi with CNNs and ViTs in terms of memory usage and training time in Section V-B. In Section V-C, we use our proposed method in Section III-C for analysis and interpretation of the reconstructed image. In Section V-D, we showcase the success of INR-LoFi and the failure of CNNs on a simple toy experiment where we have changes in the coordinate system. In Section V-E, we apply our unsupervised framework LoFi-ADMM to two inverse problems, image in-painting and radio interferometry.

A. Scalable image reconstruction with LoFi

We assess the performance of LoFi on inverse problems with local operators including low-dose computed tomography (LDCT) upon applying FBP, image denoising, and dark matter mapping in cosmology. As baselines, we consider two standard single-scale CNNs, DnCNN [11] and IRCNN [45] and two multiscale CNNs, U-Net [5] and DRU-Net [15]. We also consider two strong vision transformers, Restormer [53] and SwinIR [12], which achieved state-of-the-art performance on various image reconstruction problems. We assess the reconstruction quality with peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [81]. Further details regarding the network architectures and training details are given in Section VII in the supplementary materials.

a) Low-dose computed tomography (LDCT): We simulate parallel-beam LDCT with $r = 180$ uniformly distributed projection orientations. We add white Gaussian noise to obtain the signal-to-noise (SNR) ratio of 30dB. We use 1000 training samples of chest images from the LoDoPaB-CT dataset [82] at resolution 128×128. All models take FBP as input and return improved reconstruction in the output. The model performance is assessed on 64 in-distribution test samples of chest images. We additionally use 16 brain CT samples [83] to evaluate model generalization on out-of-distribution (OOD) data.

The upper row of Figure 4 illustrates the performance of different models on in-distribution chest images. We see that LoFi achieves performance comparable to strong CNNs and ViTs while using only simple MLP modules. The bottom row shows model performance on OOD brain samples. It is remarkable that LoFi significantly outperforms CNNs and ViTs, indicating the strong generalization achieved by the locality of the architecture. Quantitative results are presented in Table I.

TABLE I: Model comparison on the reconstruction quality for LDCT (30dB SNR) averaged on 64 in-distribution chest and 16 OOD brain samples.

	In-dist (chest)		OOD (brain)	
	PSNR	**SSIM**	**PSNR**	**SSIM**
FBP	26.0	0.55	26.0	0.61
DnCNN [11]	34.2	0.90	27.6	0.82
IRCNN [45]	**34.5**	**0.91**	29.8	0.93
U-Net [5]	33.0	0.90	13.9	0.81
DRU-Net [15]	33.8	0.90	29.0	0.90
Restormer [53]	33.5	0.89	28.9	0.79
SwinIR [12]	**34.5**	0.90	29.3	**0.94**
LoFi (ours)	34.1	0.90	**30.6**	**0.94**

b) Image denoising with tiny datasets: Here we train LoFi on image denoising on a small dataset of only 10 images at resolution 512×512; cf. Figure 13 in the supplementary materials. 9 images are used for training and 1 for evaluation. The images are scaled between 0 and 1 and we add white Gaussian noise with $\sigma = 0.15$. We compare LoFi with CNN and Restormer models with a comparable number of trainable parameters (3M). SwinIR was excluded from this experiment due to its high computational cost and very long processing time at this resolution.

Figure 5 presents the PSNR of the reconstructed test image across training iterations for each model. While CNNs initially perform well, they quickly overfit the training data, leading to a sharp drop in test image performance. Restormer and IRCNN demonstrate better generalization but also begin to overfit after 2500 and 10000 iterations, respectively. LoFi on the other hand generalizes strongly, avoiding overfitting even with 3M parameters and a tiny dataset. This is without any explicit regularization or early stopping. Final reconstructions after 20000 iterations are illustrated in Figure 6. Please also see Section VIII-A in the supplementary materials for additional experiments showcasing LoFi's performance when trained on a large-scale dataset (CelebA-HQ) with high-resolution images.

c) Dark matter mapping: We finally apply LoFi on the naive KS inversion κ_{KS} to recover the convergence field κ. In Figure 7, the reconstructed convergence fields from the KS image are presented for various models at resolution 128×128. The performance parity between LoFi and baselines, especially ViTs which can process long-range dependencies, indicates the validity of the locality property derived from (7). In Figure 17, we also demonstrate the learned filter **H** in (15); this figure can be useful to understand how it processes input KS images.

B. Computational efficiency

As discussed in Section III-E, LoFi can be trained and run on machines with small GPUs by reducing the pixel batch size in both training and inference. Figure 1 compares memory usage and training time for 100 iterations with batch size 1 across different models and image resolutions. In this experiment, we
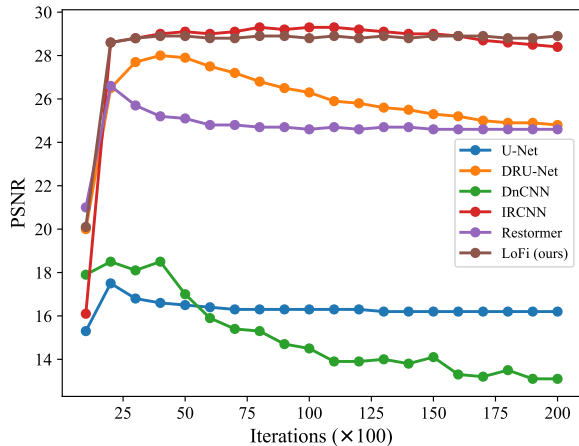
Fig. 5: Comparative analysis for image denoising at resolution 512×512 where different models are trained on a tiny dataset with 9 training samples. The PSNR of the reconstructed test samples is demonstrated per iterations during training. CNNs, in particular multiscale versions, show severe overfitting while LoFi shows a robust convergence and significantly outperforms CNNs thanks to its locality design.

used pixel batch size 512 for LoFi during training. This analysis shows that while the required memory and training time for CNNs and ViTs scales with image resolution, LoFi's memory usage and training time remains nearly constant enabling efficient training on high-resolution images.

Moreover, LoFi training is fast even for large images. For example, training on 1024×1024 images using 29900 samples from the CelebA-HQ dataset was completed in only 8 hours (200 epochs) on a single A100 GPU, requiring only 15 GB memory with object and pixel batch sizes 64 and 512 respectively. Additional details and qualitative results for this experiment can be found in Section VIII-A.

While LoFi is efficient during training, inference on large images can become expensive due to coordinate-based image synthesis. This requires iteratively reconstructing large images, as generating all pixels at once would need a bigger memory. Figure 8 compares the memory usage and inference time of different architectures. In this experiment, LoFi generates 16384 pixels per iteration. We mention potential strategies to improve inference efficiency in Section VI.

C. Reconstruction analysis and interpretation

As discussed in Section III-C, LoFi adaptively estimates patch geometry for each pixel based on local information. Figure 9 illustrates the learned patch geometry across different pixels for image denoising, where the estimated positions of relevant features align with intuitive expectations. This ability to pinpoint features contributing to each pixel's estimation can be valuable for interpretation. Additionally, the figure shows that more iterations (larger values of T) result in a more precise localization of relevant information.

D. Learning the imaging geometry with INR-LoFi

In this section, we study the performance of different models on a toy experiment where we have changes in the coordinate system. We consider recovering an image from its transpose; a simple task where we only have changes in the coordinate system while the image content remains unchanged. We train LoFi and CNNs on 29900 training samples from CelebA-HQ [84] dataset. Figure 10 showcases the performance of different models on both in-distribution (CelebA-HQ test samples) and OOD data (LSUN-bedroom [85]). Despite the simplicity of the task and access to a large-scale dataset, CNNs show poor performance as they can only learn the changes in the image content while neglecting the changes in the coordinate system. Unlike CNNs, our proposed INR-LoFi can conveniently learn these changes in the coordinate system resulting in significantly better reconstructions.

E. Low-resolution priors for high-resolution image reconstruction with LoFi-ADMM

As discussed in Section IV-B, LoFi can be used as a low-dimensional prior for processing the measurements acquired at higher resolutions. To this end, we train LoFi denoiser on low-resolution images and use it as a prior in LoFi-ADMM for solving two inverse problems at higher resolutions; image in-painting and radio interferometric imaging.

a) Image inpainting: We train LoFi denoiser on 29900 samples from the CelebA-HQ [84] in low resolution 128×128 where we consider zero-mean Gaussian noise $\mathbf{n}$ with $\sigma = 0.15$ and normalize images between 0 and 1. In LoFi-ADMM, we set $\alpha = 0.05$ and execute it for 90 iterations, the hyper-parameters we found suitable to ensure convergence. Now, we run LoFi-ADMM to solve the image-inpainting problem at a higher resolution 512×512 where $p = 30\%$ of the image pixels are randomly masked. Figure 11 showcases the performance of the LoFi-ADMM on both in-distribution and OOD data. This experiment shows that once LoFi denoiser is trained, it can be used for solving general inverse problems at any higher resolution with a strong generalization.

b) Radio interferometry: We next apply LoFi-ADMM to radio interferometric imaging, please refer to Section VII-F in the supplementary material for further information regarding the forward model. We use non-uniform fast Fourier transform (NUFFT) to simulate radio interferometric measurements from 128×128 images using an upsampling factor of 2 and Kaiser–Bessel kernels with a size of 6×6 pixels; we use the simulated uv-coverage of the MeerKAT telescope and the measurements with complex Gaussian noise at the SNR of 30dB. We train LoFi denoiser on 13000 images of simulated galaxies created from the IllustrisTNG simulations [86]. Figure 12 illustrates the performance of the LoFi-ADMM at original and higher resolutions. This experiment shows that LoFi-ADMM can conveniently exploit the measurements obtained from higher-resolution images for more accurate reconstructions with more details.

VI. LIMITATIONS AND CONCLUSION

We introduced LoFi, a scalable local image reconstruction framework that stands out for its strong generalization on OOD
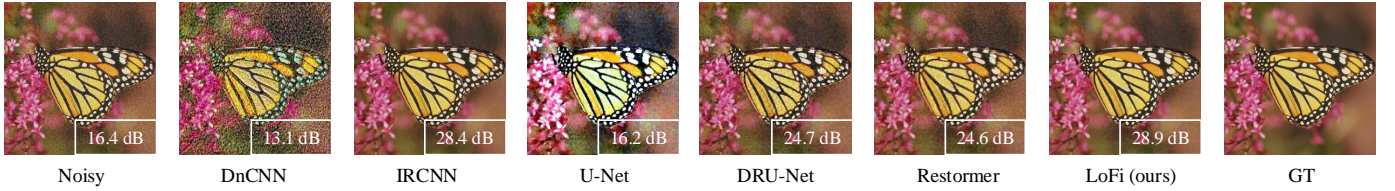
Fig. 6: Performance comparison on image denoising where different models are trained for 20000 iterations on a tiny training set with 9 images shown in Figure 13.
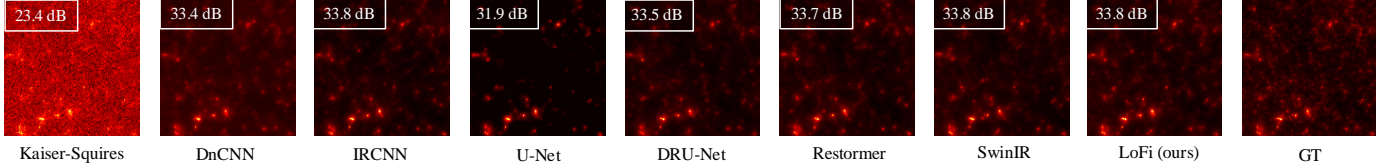

Fig. 7: Performance comparison on reconstructed convergence fields for dark matter mapping at resolution 128×128.
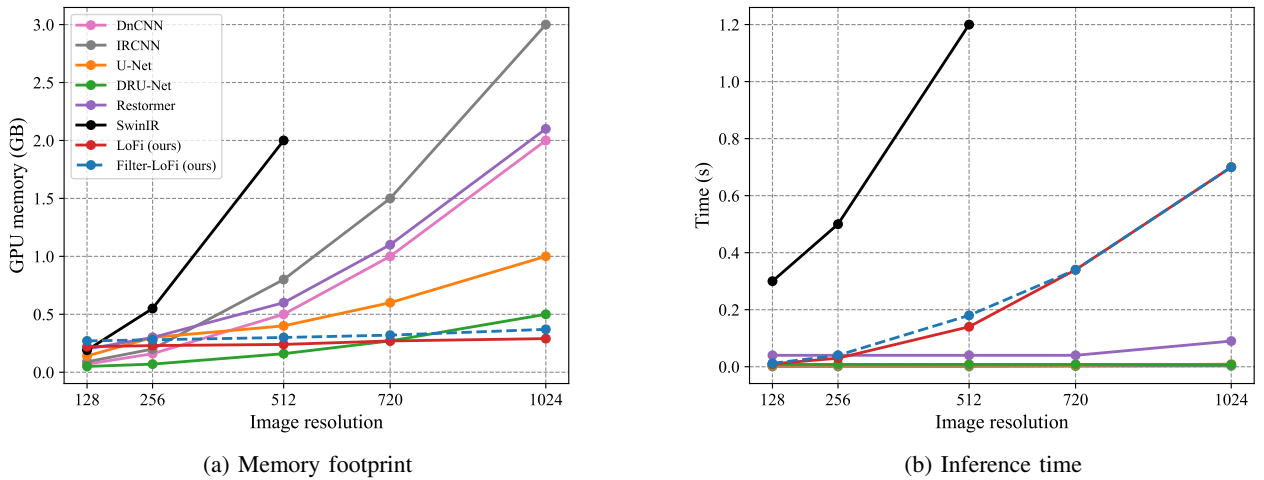

(a) Memory footprint

(b) Inference time

Fig. 8: The memory and time requirements during inference for different models
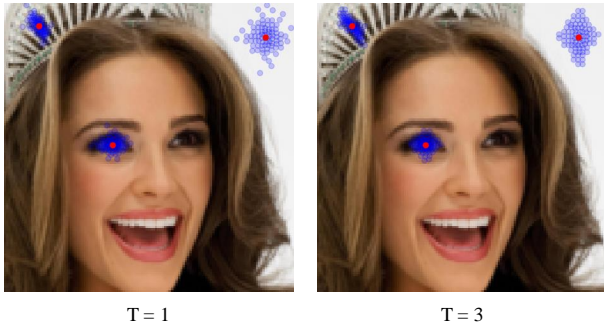

T = 1   T = 3

Fig. 9: The learned patch geometry for different pixels. As expected, larger T results in more accurate localization of the relevant information. The red dots indicate the query pixels, while the blue dots represent the estimated positions of relevant features to be extracted from the noisy image.

data and resolution agnostic memory usage during training. LoFi achieves strong performance on local inverse problems such as LDCT and image denoising, all while maintaining a significantly smaller memory footprint compared to standard CNNs and vision transformers. The favorable inductive bias of LoFi allows us to train it on very small datasets without overfitting or explicit regularization. Our experiments further demonstrate the effectiveness of LoFi as a denoising prior within the PnP framework, showcasing its ability to reconstruct images at arbitrary resolutions for tasks such as image inpainting and radio interferometric imaging.

Although LoFi's pixel-based pipeline significantly reduces memory requirements during training, it comes with trade-offs. Like INRs, the computational cost of LoFi in inference increases quadratically with image resolution as shown in Figure 8b. Recent studies [87], [88] have proposed strategies to enhance the efficiency of continuous image representation in INRs by increasing shared computations across pixels, thereby reducing computational complexity. Implementing similar strategies in LoFi could potentially reduce computations during inference, which we leave as a direction for future work.
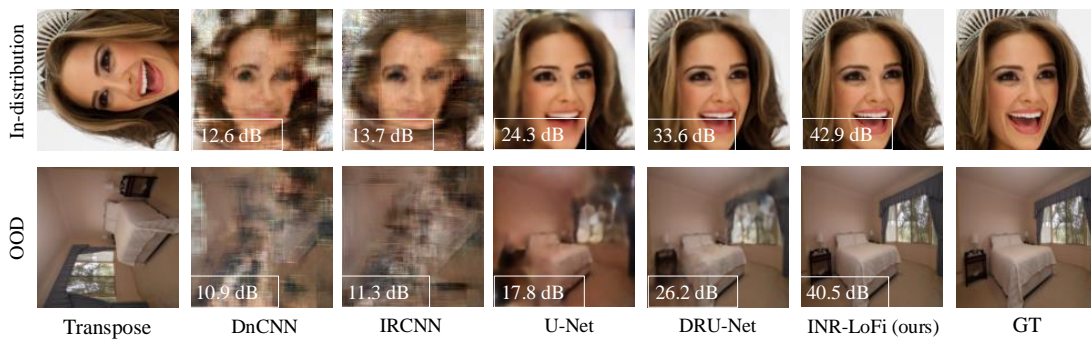
Fig. 10: Performance comparison on toy experiment of recovering an image from its transpose.
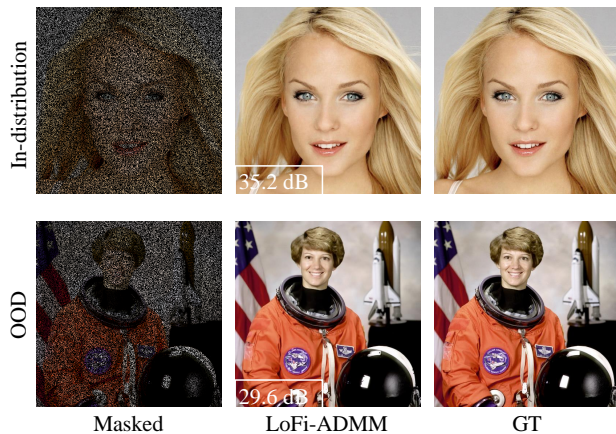

Fig. 11: LoFi-ADMM performance on image in-painting ($p = 30\%$) for in-distribution and OOD data at resolution 512×512 when LoFi denoiser is trained on CelebA-HQ samples in low resolution 128×128.
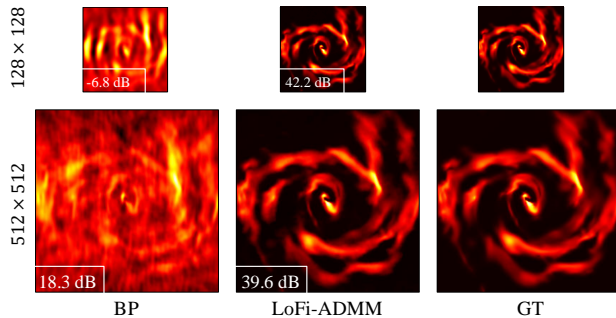

Fig. 12: LoFi-ADMM performance on radio interferometry; the LoFi prior is trained on low-resolution images 128×128.

REFERENCES

[1] G. Wang, H. Yu, and B. De Man, "An outlook on x-ray ct research and development," *Medical physics*, vol. 35, no. 3, pp. 1051–1064, 2008.

[2] M. Holler, M. Guizar-Sicairos, E. H. Tsai, R. Dinapoli, E. Müller, O. Bunk, J. Raabe, and G. Aeppli, "High-resolution non-destructive three-dimensional imaging of integrated circuits," *Nature*, vol. 543, no. 7645, pp. 402–406, 2017.

[3] R. E. Blahut, *Theory of remote image formation*. Cambridge University Press, 2004.

[4] N. Kaiser and G. Squires, "Mapping the Dark Matter with Weak Gravitational Lensing," *ApJ*, vol. 404, p. 441, Feb. 1993.

[5] O. Ronneberger, P. Fischer, and 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*, pp. 234–241, Springer, 2015.

[6] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE transactions on image processing*, vol. 26, no. 9, pp. 4509–4522, 2017.

[7] C. M. Hyun, H. P. Kim, S. M. Lee, S. Lee, and J. K. Seo, "Deep learning for undersampled mri reconstruction," *Physics in Medicine & Biology*, vol. 63, no. 13, p. 135007, 2018.

[8] M. Mars, M. M. Betcke, and J. D. McEwen, "Learned radio interferometric imaging for varying visibility coverage," *arXiv preprint arXiv:2405.08958*, 2024.

[9] N. Jeffrey, F. Lanusse, O. Lahav, and J.-L. Starck, "Deep learning dark matter map reconstructions from des sv weak lensing data," *Monthly Notices of the Royal Astronomical Society*, vol. 492, no. 4, pp. 5023–5029, 2020.

[10] T. Liu, A. Chaman, D. Belius, and I. Dokmanić, "Learning multiscale convolutional dictionaries for image reconstruction," *IEEE Transactions on Computational Imaging*, vol. 8, pp. 425–437, 2022.

[11] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE transactions on image processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[12] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Swinir: Image restoration using swin transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1833–1844, 2021.

[13] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 568–578, 2021.

[14] M. S. Sepehri, Z. Fabian, and M. Soltanolkotabi, "Serpent: Scalable and efficient image restoration via multi-scale structured state space models," *arXiv preprint arXiv:2403.17902*, 2024.

[15] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6360–6376, 2021.

[16] Z. Fabian, B. Tinaz, and M. Soltanolkotabi, "Humus-net: Hybrid unrolled multi-scale network architecture for accelerated mri reconstruction," *Advances in Neural Information Processing Systems*, vol. 35, pp. 25306–25319, 2022.

[17] H. K. Aggarwal, M. P. Mani, and M. Jacob, "Modl: Model-based deep learning architecture for inverse problems," *IEEE transactions on medical imaging*, vol. 38, no. 2, pp. 394–405, 2018.

[18] V. Sitzmann, J. Martel, A. Bergman, D. Lindell, and G. Wetzstein, "Implicit neural representations with periodic activation functions," *Advances in neural information processing systems*, vol. 33, pp. 7462–7473, 2020.

[19] M. Atzmon and Y. Lipman, "Sal: Sign agnostic learning of shapes from raw data," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2565–2574, 2020.

[20] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. Newcombe, "Deep local shapes: Learning local sdf priors for detailed 3d reconstruction," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pp. 608–625, Springer, 2020.

[21] Z. Chen and H. Zhang, "Learning implicit fields for generative shape

modeling," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5939–5948, 2019.

[22] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger, "Convolutional occupancy networks," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 523–540, Springer, 2020.

[23] C. Jiang, A. Sud, A. Makadia, J. Huang, M. Nießner, T. Funkhouser, *et al.*, "Local implicit grid representations for 3d scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6001–6010, 2020.

[24] E. Dupont, H. Kim, S. Eslami, D. Rezende, and D. Rosenbaum, "From data to functa: Your data point is a function and you can treat it like one," *arXiv preprint arXiv:2201.12204*, 2022.

[25] E. Dupont, Y. W. Teh, and A. Doucet, "Generative models as distributions of functions," *arXiv preprint arXiv:2102.04776*, 2021.

[26] A. Susmelj, M. Macuglia, N. Tagasovska, R. Sutter, S. Caprara, J.-P. Thiran, and E. Konukoglu, "Uncertainty modeling for fine-tuned implicit functions," *arXiv preprint arXiv:2406.12082*, 2024.

[27] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[28] S. Mohan, Z. Kadkhodaie, E. P. Simoncelli, and C. Fernandez-Granda, "Robust and interpretable blind image denoising via bias-free convolutional neural networks," *ICLR*, 2020.

[29] Z. Wang, X. Cun, J. Bao, W. Zhou, J. Liu, and H. Li, "Uformer: A general u-shaped transformer for image restoration," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17683–17693, 2022.

[30] H. Zhao, Y. Gou, B. Li, D. Peng, J. Lv, and X. Peng, "Comprehensive and delicate: An efficient transformer for image restoration," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14122–14132, 2023.

[31] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 460–489, 2005.

[32] K. Kothari, A. Khorashadizadeh, M. de Hoop, and I. Dokmanić, "Trumpets: Injective flows for inference and inverse problems," in *Uncertainty in Artificial Intelligence*, pp. 1269–1278, PMLR, 2021.

[33] A. Jalal, M. Arvinte, G. Daras, E. Price, A. G. Dimakis, and J. Tamir, "Robust compressed sensing mri with deep generative priors," *Advances in Neural Information Processing Systems*, vol. 34, pp. 14938–14954, 2021.

[34] H. Chung, J. Kim, M. T. Mccann, M. L. Klasky, and J. C. Ye, "Diffusion posterior sampling for general noisy inverse problems," *arXiv preprint arXiv:2209.14687*, 2022.

[35] Y. Hu, M. Delbracio, P. Milanfar, and U. S. Kamilov, "A restoration network as an implicit prior," *ICLR*, 2024.

[36] A. Levin and B. Nadler, "Natural image denoising: Optimality and inherent bounds," in *CVPR 2011*, pp. 2833–2840, IEEE, 2011.

[37] J. R. Mayo, K. P. Whittall, A. N. Leung, T. E. Hartman, C. S. Park, S. L. Primack, G. K. Chambers, M. K. Limkeman, T. L. Toth, and S. H. Fox, "Simulated dose reduction in conventional chest ct: validation study.," *Radiology*, vol. 202, no. 2, pp. 453–457, 1997.

[38] D. J. Brenner and E. J. Hall, "Computed tomography—an increasing source of radiation exposure," *New England journal of medicine*, vol. 357, no. 22, pp. 2277–2284, 2007.

[39] A. J. Einstein, M. J. Henzlova, and S. Rajagopalan, "Estimating risk of cancer associated with radiation exposure from 64-slice computed tomography coronary angiography," *Jama*, vol. 298, no. 3, pp. 317–323, 2007.

[40] S. Helgason and S. Helgason, *The radon transform*, vol. 2. Springer, 1980.

[41] M. B. Kiss, A. Biguri, C.-B. Schönlieb, K. J. Batenburg, and F. Lucka, "Learned denoising with simulated and experimental low-dose ct data," *arXiv preprint arXiv:2408.08115*, 2024.

[42] R. Mandelbaum, "Weak lensing for precision cosmology," *Annual Review of Astronomy and Astrophysics*, vol. 56, no. 1, pp. 393–433, 2018.

[43] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with bm3d?," in *2012 IEEE conference on computer vision and pattern recognition*, pp. 2392–2399, IEEE, 2012.

[44] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *2011 international conference on computer vision*, pp. 479–486, IEEE, 2011.

[45] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3929–3938, 2017.

[46] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-cnn for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 773–782, 2018.

[47] B. Park, S. Yu, and J. Jeong, "Densely connected hierarchical network for image denoising," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 0–0, 2019.

[48] F. Jia, W. H. Wong, and T. Zeng, "Ddunet: Dense dense u-net with applications in image denoising," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 354–364, 2021.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[50] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[51] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, and W. Gao, "Pre-trained image processing transformer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12299–12310, 2021.

[52] Y. Mansour, K. Lin, and R. Heckel, "Image-to-image mlp-mixer for image reconstruction," *arXiv preprint arXiv:2202.02018*, 2022.

[53] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, "Restormer: Efficient transformer for high-resolution image restoration," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5728–5739, 2022.

[54] B. Heo, S. Yun, D. Han, S. Chun, J. Choe, and S. J. Oh, "Rethinking spatial dimensions of vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11936–11945, 2021.

[55] T. Vlašić, H. Nguyen, A. Khorashadizadeh, and I. Dokmanić, "Implicit neural representation for mesh-free inverse obstacle scattering," in *2022 56th Asilomar Conference on Signals, Systems, and Computers*, pp. 947–952, IEEE, 2022.

[56] Y. Chen, S. Liu, and X. Wang, "Learning continuous image representation with local implicit image function," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8628–8638, 2021.

[57] A. Khorashadizadeh, A. Chaman, V. Debarnot, and I. Dokmanić, "Funknn: Neural interpolation for functional generation," *ICLR*, 2023.

[58] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 2, pp. 60–65, Ieee, 2005.

[59] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on image processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[60] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, *et al.*, "Mlp-mixer: An all-mlp architecture for vision," *Advances in neural information processing systems*, vol. 34, pp. 24261–24272, 2021.

[61] A. Trockman and J. Z. Kolter, "Patches are all you need?," *arXiv preprint arXiv:2201.09792*, 2022.

[62] Z. Ding, M. Zhang, J. Wu, and Z. Tu, "Patched denoising diffusion models for high-resolution image synthesis," in *ICLR*, 2024.

[63] Z. Wang, Y. Jiang, H. Zheng, P. Wang, P. He, Z. Wang, W. Chen, M. Zhou, *et al.*, "Patch diffusion: Faster and more data-efficient training of diffusion models," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[64] D. Gilton, G. Ongie, and R. Willett, "Learned patch-based regularization for inverse problems in imaging," in *2019 ieee 8th international workshop on computational advances in multi-sensor adaptive processing (camsap)*, pp. 211–215, IEEE, 2019.

[65] F. Altekrüger, A. Denker, P. Hagemann, J. Hertrich, P. Maass, and G. Steidl, "Patchnr: Learning from small data by patch normalizing flow regularization," *arXiv preprint arXiv:2205.12021*, 2022.

[66] J. Hertrich, A. Houdard, and C. Redenbach, "Wasserstein patch prior for image superresolution," *IEEE Transactions on Computational Imaging*, vol. 8, pp. 693–704, 2022.

[67] M. Piening, F. Altekrüger, J. Hertrich, P. Hagemann, A. Walther, and G. Steidl, "Learning from small data sets: Patch-based regularizers in inverse problems for image reconstruction," *GAMM-Mitteilungen*, p. e202470002, 2024.

[68] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 764–773, 2017.

[69] M. Diwakar and M. Kumar, "A review on ct image noise and its denoising," *Biomedical Signal Processing and Control*, vol. 42, pp. 73–88, 2018.

[70] K. Kothari, M. de Hoop, and I. Dokmanić, "Learning the geometry of wave-based imaging," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8318–8329, 2020.

[71] M. T. McCann, K. H. Jin, and M. Unser, "Convolutional neural networks for inverse problems in imaging: A review," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 85–95, 2017.

[72] Z. Wei and X. Chen, "Deep-learning schemes for full-wave nonlinear inverse scattering problems," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 4, pp. 1849–1860, 2018.

[73] A. Khorashadizadeh, K. Kothari, L. Salsi, A. A. Harandi, M. de Hoop, and I. Dokmanić, "Conditional injective flows for bayesian imaging," *IEEE Transactions on Computational Imaging*, vol. 9, pp. 224–237, 2023.

[74] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, "Compressed sensing using generative models," in *International conference on machine learning*, pp. 537–546, PMLR, 2017.

[75] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising diffusion restoration models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23593–23606, 2022.

[76] A. Khorashadizadeh, V. Khorashadizadeh, S. Eskandari, G. A. Vandenbosch, and I. Dokmanić, "Deep injective prior for inverse scattering," *IEEE Transactions on Antennas and Propagation*, 2023.

[77] T. Liu, T. Yang, Q. Zhang, and Q. Lei, "Optimization for amortized inverse problems," in *International Conference on Machine Learning*, pp. 22289–22319, PMLR, 2023.

[78] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *2013 IEEE global conference on signal and information processing*, pp. 945–948, IEEE, 2013.

[79] S. H. Chan, X. Wang, and O. A. Elgendy, "Plug-and-play admm for image restoration: Fixed-point convergence and applications," *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 84–98, 2016.

[80] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (red)," *SIAM Journal on Imaging Sciences*, vol. 10, no. 4, pp. 1804–1844, 2017.

[81] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[82] J. Leuschner, M. Schmidt, D. O. Baguer, and P. Maass, "Lodopab-ct, a benchmark dataset for low-dose computed tomography reconstruction," *Scientific Data*, vol. 8, no. 1, p. 109, 2021.

[83] M. Hssayeni, M. Croock, A. Salman, H. Al-khafaji, Z. Yahya, and B. Ghoraani, "Computed tomography images for intracranial hemorrhage detection and segmentation," *Intracranial hemorrhage segmentation using a deep convolutional model. Data*, vol. 5, no. 1, p. 14, 2020.

[84] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[85] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop," *arXiv preprint arXiv:1506.03365*, 2015.

[86] D. Nelson, A. Pillepich, V. Springel, R. Pakmor, R. Weinberger, S. Genel, P. Torrey, M. Vogelsberger, F. Marinacci, and L. Hernquist, "First results from the tng50 simulation: galactic outflows driven by supernovae and black hole feedback," *Monthly Notices of the Royal Astronomical Society*, vol. 490, no. 3, pp. 3234–3261, 2019.

[87] Z. He and Z. Jin, "Dynamic implicit image function for efficient arbitrary-scale image representation," *arXiv preprint arXiv:2306.12321*, 2023.

[88] Z. He and Z. Jin, "Latent modulated function for computational optimal continuous image representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26026–26035, 2024.

[89] K. Osato, J. Liu, and Z. Haiman, "κTNG: effect of baryonic processes on weak lensing with IllustrisTNG simulations," *Monthly Notices of the Royal Astronomical Society*, vol. 502, pp. 5593–5602, 02 2021.

[90] A. Pillepich, D. Nelson, L. Hernquist, V. Springel, R. Pakmor, P. Torrey, R. Weinberger, S. Genel, J. P. Naiman, F. Marinacci, and M. Vogelsberger, "First results from the IllustrisTNG simulations: the stellar mass content of groups and clusters of galaxies," *Monthly Notices of the Royal Astronomical Society*, vol. 475, pp. 648–675, Mar. 2018.

[91] V. Springel, R. Pakmor, A. Pillepich, R. Weinberger, D. Nelson, L. Hernquist, M. Vogelsberger, S. Genel, P. Torrey, F. Marinacci, and J. Naiman, "First results from the IllustrisTNG simulations: matter and galaxy clustering," *Monthly Notices of the Royal Astronomical Society*, vol. 475, pp. 676–698, Mar. 2018.

[92] J. P. Naiman, A. Pillepich, V. Springel, E. Ramirez-Ruiz, P. Torrey, M. Vogelsberger, R. Pakmor, D. Nelson, F. Marinacci, L. Hernquist, R. Weinberger, and S. Genel, "First results from the IllustrisTNG simulations: a tale of two elements - chemical evolution of magnesium and europium," *Monthly Notices of the Royal Astronomical Society*, vol. 477, pp. 1206–1224, June 2018.

[93] D. Nelson, A. Pillepich, V. Springel, R. Weinberger, L. Hernquist, R. Pakmor, S. Genel, P. Torrey, M. Vogelsberger, G. Kauffmann, F. Marinacci, and J. Naiman, "First results from the IllustrisTNG simulations: the galaxy colour bimodality," *Monthly Notices of the Royal Astronomical Society*, vol. 475, pp. 624–647, Mar. 2018.

[94] F. Marinacci, M. Vogelsberger, R. Pakmor, P. Torrey, V. Springel, L. Hernquist, D. Nelson, R. Weinberger, A. Pillepich, J. Naiman, and S. Genel, "First results from the IllustrisTNG simulations: radio haloes and magnetic fields," *Monthly Notices of the Royal Astronomical Society*, vol. 480, pp. 5113–5139, Nov. 2018.

[95] M. A. Price, J. D. McEwen, L. Pratley, and T. D. Kitching, "Sparse Bayesian mass-mapping with uncertainties: Full sky observations on the celestial sphere," *Monthly Notices of the Royal Astronomical Society*, vol. 500, pp. 5436–5452, 11 2020.

[96] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[97] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 10, pp. 2305–2318, 2018.

[98] K. Wei, A. Aviles-Rivero, J. Liang, Y. Fu, H. Huang, and C.-B. Schönlieb, "Tfpnp: Tuning-free plug-and-play proximal algorithms with applications to inverse imaging problems," *The Journal of Machine Learning Research*, vol. 23, no. 1, pp. 699–746, 2022.

[99] R. Ahmad, C. A. Bouman, G. T. Buzzard, S. Chan, S. Liu, E. T. Reehorst, and P. Schniter, "Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery," *IEEE signal processing magazine*, vol. 37, no. 1, pp. 105–116, 2020.

[100] K. Wei, A. Aviles-Rivero, J. Liang, Y. Fu, C.-B. Schönlieb, and H. Huang, "Tuning-free plug-and-play proximal algorithm for inverse imaging problems," in *International Conference on Machine Learning*, pp. 10158–10169, PMLR, 2020.

[101] U. S. Kamilov, C. A. Bouman, G. T. Buzzard, and B. Wohlberg, "Plug-and-play methods for integrating physical and learned models in computational imaging: Theory, algorithms, and applications," *IEEE Signal Processing Magazine*, vol. 40, no. 1, pp. 85–97, 2023.

[102] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," *Advances in neural information processing systems*, vol. 31, 2018.

[103] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[104] Y. Zhu, K. Zhang, J. Liang, J. Cao, B. Wen, R. Timofte, and L. Van Gool, "Denoising diffusion models for plug-and-play image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1219–1229, 2023.

[105] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[106] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[107] H. Y. Tan, S. Mukherjee, J. Tang, and C.-B. Schönlieb, "Provably convergent plug-and-play quasi-newton methods," *SIAM Journal on Imaging Sciences*, vol. 17, no. 2, pp. 785–819, 2024.

[108] W. Gan, Y. Hu, J. Liu, H. An, U. Kamilov, *et al.*, "Block coordinate plug-and-play methods for blind inverse problems," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[109] A. Hauptmann, S. Mukherjee, C.-B. Schönlieb, and F. Sherry, "Convergent regularization in inverse problems and linear plug-and-play denoisers," *Foundations of Computational Mathematics*, pp. 1–34, 2024.

[110] L. Pratley, J. D. McEwen, M. d'Avezac, R. E. Carrillo, A. Onose, and Y. Wiaux, "Robust sparse image reconstruction of radio interferometric observations with purify," *Monthly Notices of the Royal Astronomical Society*, vol. 473, pp. 1038–1058, 09 2017.

[111] T. I. Liaudat, M. Mars, M. A. Price, M. Pereyra, M. M. Betcke, and J. D. McEwen, "Scalable Bayesian uncertainty quantification with data-driven priors for radio interferometric imaging," *arXiv e-prints*, p. arXiv:2312.00125, Nov. 2023.

[112] T. Cohen and M. Welling, "Group equivariant convolutional networks," in *International conference on machine learning*, pp. 2990–2999, PMLR, 2016.

[113] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric deep learning: Grids, groups, graphs, geodesics, and gauges," *arXiv preprint arXiv:2104.13478*, 2021.

[114] T. S. Cohen and M. Welling, "Steerable cnns," *arXiv preprint arXiv:1612.08498*, 2016.

[115] B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling, "Rotation equivariant cnns for digital pathology," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II 11*, pp. 210–218, Springer, 2018.

[116] A. Chaman and I. Dokmanić, "Truly shift-equivariant convolutional neural networks with adaptive polyphase upsampling," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pp. 1113–1120, IEEE, 2021.

[117] A. Chaman and I. Dokmanić, "Truly shift-invariant convolutional neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3773–3783, 2021.

[118] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2, pp. 1150–1157, Ieee, 1999.

[119] O. Puny, M. Atzmon, H. Ben-Hamu, I. Misra, A. Grover, E. J. Smith, and Y. Lipman, "Frame averaging for invariant and equivariant network design," *arXiv preprint arXiv:2110.03336*, 2021.

[120] A. Sannai, M. Kawano, and W. Kumagai, "Equivariant and invariant reynolds networks," *arXiv preprint arXiv:2110.08092*, 2021.

[121] Z. Zhao and A. Singer, "Rotationally invariant image representation for viewing direction classification in cryo-em," *Journal of structural biology*, vol. 186, no. 1, pp. 153–166, 2014.

Supplementary Materials

VII. Network architecture and training details

We describe the network architecture and training details of LoFi and CNN baselines used in the experiments in Section V.

A. Low-dose computed tomography

For this experiment, we parameterize LoFi with multiMLP architecture depicted in Figure 3 where we used 9 MLP modules, each with three hidden layers of dimension 370 with ReLu activations and the output layer has 100 neurons. Their outputs are then concatenated and mixed by another MLP with an input layer of size $900 = 9 \times 100$ and 3 hidden layers of dimension 370 with ReLu activations. We consider $K = 81$ neighboring pixels initialized with circular geometry, 9 circles each with 9 points uniformly distributed around the centric coordinate. Regarding the pre-processing filter, we concatenated the real and complex values of the filtered input image along the channel dimension before patch extraction. Regarding the architecture of baselines, we used the standard implementations published by the authors with the following configurations.

DnCNN: number of layers=15, number of channels=160
IRCNN: number of layers = 13, number of channels = 256
U-Net: Init-features = 32
DRU-Net: Init-features = 32
Restormer: Transformer blocks = [4, 6, 6, 8], attention heads = [1, 2, 4, 8] and number of channels = [16, 32, 64, 128].
SwinIR: Patch embedding dimension = 120, patch size = 1, window size = 8, attention heads = [6,6,6,6,6,6].

B. Image denoising

Similar to the experiment for LDCT, we used multiMLP architecture with the same structure. For this particular experiment, we did not use the pre-processing filter to avoid overfitting as our training set is very small. Regarding the architecture of baselines, we used smaller networks for U-Net ahnd DRU-Net for having a fair comparison with other baselines and LoFi in terms of overfitting.

U-Net: Init-features = 19
DRU-Net: Init-features = 19

Wit these parameters, all baselines and LoFi have approximately 3M parameres.

C. Dark matter mapping

For this experiment, we used the same network architecture as the experiment of LDCT for both LoFi and baselines. Regarding dark matter mapping simulations, we used the κTNG convergence maps [89], generated from the widely used IllustrisTNG hydrodynamical simulation suite [90]–[94], to build the train and test datasets for the models presented in Figure 7. The κTNG dataset consists of $10\,000$ realizations of $5 \times 5\mathrm{deg}^2$ convergence maps for each of the 40 different source redshifts with $0 < z_s < 2.6$. For the sake of simplicity, we have

selected a single redshift slice, the 20th plane, corresponding to $z_s = 0.858$ for our experiments.

From the simulated convergence fields, κ, we have constructed the observed shear fields using Equation 5. We follow [95, §5.2.1] for the noise simulation. The shear noise $\mathbf{n}$ for each component of the shear and for each pixel i is simulated as $\mathbf{n}_i \sim \mathcal{N}(0, \sigma_i^2)$. The per-pixel standard deviation can be computed as follows

$$\sigma_i = \frac{\sigma_e}{\sqrt{(\theta^2/n_{\text{grid}}^2) \times n_{\text{gal}}}}, \qquad (19)$$

where σ_e is the standard deviation of the intrinsic ellipticity distribution, θ^2 is the area of the simulated field in arcmin^2, n_{grid}^2 is the total number of pixels in the field, and n_{gal} is the number density of galaxy observations given in number of galaxies per arcmin^2. We have used $\sigma_e = 0.37$, which is the typical intrinsic ellipticity standard deviation, $\theta = 300$ arcmin, which corresponds to the κTNG simulated area of $5 \times 5\text{deg}^2$, $n_{\text{gal}} = 30$ arcmin^{-2}, which corresponds to the projected number density expected in Stage IV surveys like *Euclid* , and $n_{\text{grid}} = 128$.

D. Training details

LoFi is implemented in PyTorch on a machine with an Nvidia A100 GPU with 80GB memory. LoFi and baselines were trained for 200 epochs with ℓ_1 loss using Adam optimizer [96] with learning rate 10^{-4} and object batch size 64. In the case of LoFi, for each mini-batch of random objects, we performed optimization on a random mini-batch of 512 pixels 2 times.

E. Plug-and-Play ADMM

In this section, we briefly review plug-and-play (PnP) framework for image reconstruction. PnP algorithms achieved SOTA performance across a variety of image reconstruction tasks including image super-resolution and deblurring [15], [97], computed tomography (CT) [98] and magnetic resonance imaging (MRI) [99], [100]; see also the recent review [101]. In PnP, a pre-trained Gaussian denoiser, typically a CNN, is used as a prior. Gaussian denoisers are often cheaper in both than generative priors [102], [103] in both training and inference. Training generative models also requires large-scale datasets and substantial computational resources. Recently, the authors of [104] leveraged diffusion models [103], [105] as a pre-trained denoiser in PnP framework and achieved excellent performance for various image restoration tasks.

Originally, PnP [78] is developed based on the alternating direction method of multipliers (ADMM) [106], where new variables $\mathbf{u}$ and $\mathbf{v}$ are introduced to decouple the data fidelity and regularization terms in (1) as follows,

$$\min_{\mathbf{f}, \mathbf{v}} \max_{\mathbf{u}} \left\{ \frac{1}{2\sigma^2} \|\mathbf{q} - \mathbf{A}\mathbf{f}\|_2^2 + R(\mathbf{v}) + \frac{1}{2\eta} \|\mathbf{f} - \mathbf{v} + \mathbf{u}\|_2^2 \right.$$
$$\left. - \frac{1}{2\eta} \|\mathbf{u}\|_2^2 \right\}, \qquad (20)$$

where η is a penalty parameter that adjusts the convergence rate. We alternate the optimization on $\mathbf{f}$, $\mathbf{u}$ and $\mathbf{v}$ as follows,

$$\mathbf{f}_k = h(\mathbf{v}_{k-1} - \mathbf{u}_{k-1}; \alpha), \qquad (21)$$
$$\mathbf{v}_k = \text{prox}_R(\mathbf{f}_k - \mathbf{u}_{k-1}; \eta), \qquad (22)$$
$$\mathbf{u}_k = \mathbf{u}_{k-1} + (\mathbf{f}_k - \mathbf{v}_k), \qquad (23)$$

where $\alpha = \dfrac{\sigma^2}{\eta}$ and,

$$h(\mathbf{z}; \alpha) \triangleq (\mathbf{A}^*\mathbf{A} + \alpha)^{-1}(\mathbf{A}^*\mathbf{q} + \alpha\mathbf{z}), \qquad (24)$$

$$\text{prox}_R(\mathbf{z}; \eta) \triangleq \underset{\mathbf{f}}{\text{argmin}} \frac{1}{2\eta} \|\mathbf{f} - \mathbf{z}\|_2^2 + R(\mathbf{f}), \qquad (25)$$

where $\mathbf{A}^*$ is the adjoint forward operator. The proximal operator in (25) can be interpreted as the denoiser of $\mathbf{z}$ with image prior $R(\cdot)$ and AWGN variance η. The key idea of the PnP is to employ a powerful image denoiser, like a pre-trained CNN, for (25). The convergence analysis of PnP is studied in the literature [107]–[109].

F. Radio interferometric imaging

Radio interferometry is an imaging technique where we acquire samples of the Fourier transform of the image along curves [110], [111]. In radio interferometric imaging, aperture synthesis techniques are used to acquire specific Fourier measurements giving an incomplete coverage of the Fourier plane. The inverse problem consists of recovering the entire intensity image from Fourier measurements. In this case, the forward operator in a simplified form writes

$$\mathbf{A} = \mathbf{M}_{\text{RI}}\mathbf{F} \qquad (26)$$

where $\mathbf{M}_{\text{RI}}$ is a Fourier mask indicating the Fourier coverage of the observations. We have omitted gridding and degridding operations in between other corrections from the forward operator for the sake of simplicity; a more detailed description of the operator can be found in [110, §3.2].

VIII. FUTHER ANALYSIS AND ADDITIONAL EXPERIMENTS

In this section we provide further analysis of LoFi architecture and additional experiments.

A. High-dimensional image denoising

In this section, we train LoFi on 29900 training samples from the CelebA-HQ dataset at resolution 1024×1024 for image denoising task ($\sigma = 0.2$).

Figure 14 illustrates the reconstructed images for both in-distribution and OOD samples. This experiment shows that LoFi can achieve high-quality reconstructions for large images despite the simplicity of its architecture.

B. Equivariance to shifts and rotations

In many imaging and pattern recognition tasks, it is desirable to work with functions that are invariant or equivariant to shifts and rotations. Building such invariances into CNNs, typically via group representation theory [112], [113] improves generalization both in and out of distribution [114], [115].

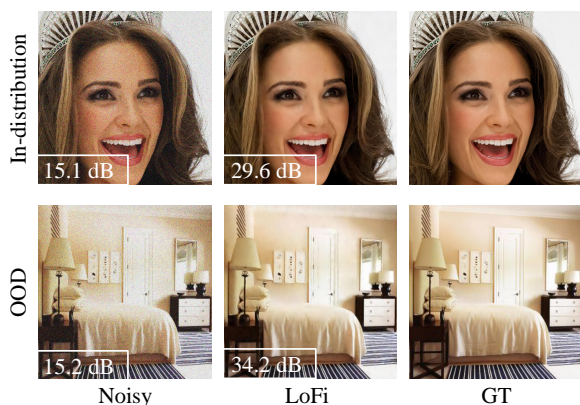Fig. 13: Training dataset used for image denoising experiment at resolution 512×512.

Fig. 14: LoFi reconstructions on in-distribution and OOD data for image denoising ($\sigma = 0.2$) at resolution 1024×1024.
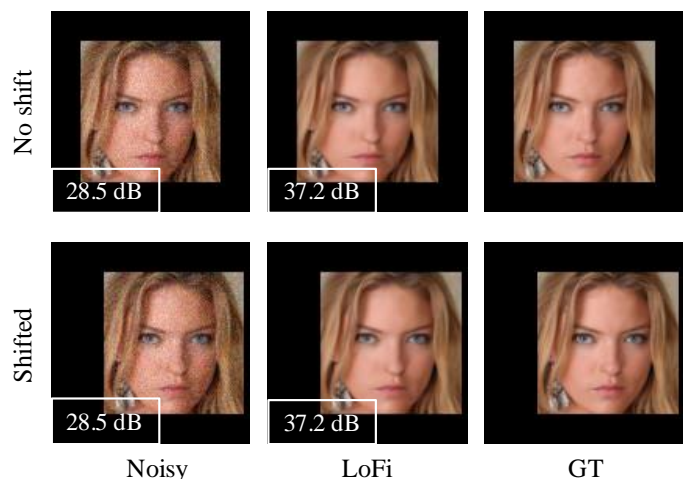

Fig. 15: Shift equivariance analysis for LoFi architecture; LoFi exhibits no degradation in reconstruction quality when the input noisy image is shifted for 15 pixels.

It is easy to see that LoFi is shift equivariant by design. Indeed, the patch at position $\mathbf{x} + \Delta\mathbf{x}$ in an image shifted by $\Delta\mathbf{x}$ is exactly the same as the patch at position $\mathbf{x}$ in a non-shifted image; the NN_θ (MultiMLP) thus receives the same input. This stands in contrast to most mainstream multi-scale CNNs which are surprisingly sensitive even to 1-pixel shifts, although equivariance can be restored by a careful design of pooling and downsampling layers [116], [117].

Rotation equivariance is precluded for square patches, except for angles that are multiples of $\frac{\pi}{2}$. With circular patches shown in Figure 3, when the image is rotated the NN_θ receives the same input only in a different order up to interpolation error. Rotation equivariance could be implemented by applying NN_θ to rotation invariant features [118]–[121] extracted from the circular patch or by averaging the output over rotations. Even without this, however, LoFi is approximately rotation equivariant because natural images are approximately *locally* rotation invariant and the patch geometry is initialized as circular.

Now, we assess LoFi performance on the image denoising task when the input image is shifted or rotated. As shown in Figure 15, we horizontally shifted the input noisy image for 15 pixels. As expected, LoFi's reconstruction is shifted for 15 pixels and exhibits no degradation in the reconstruction quality.

This experiment confirms that LoFi is a truly shift equivariant network.

Figure 16 demonstrates the LoFi performance when the input noisy image undergoes a 90-degree rotation. As expected, we observe minimal degradation in reconstruction quality which shows LoFi's strong robustness under rotation. Note that, we did not use any data augmentation to make the model robust under translation and rotation during training.

Fig. 16: Rotation equivariance analysis for LoFi architecture; LoFi exhibits minimal degradation in reconstruction quality when the input noisy image undergoes a 90-degree rotation.
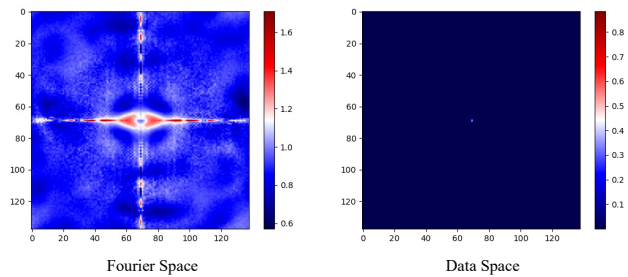

Fig. 17: The learned filter $\mathbf{H}$ in (15).