3D Field of Junctions: A Noise-Robust, Training-Free Structural Prior for Volumetric Inverse Problems

Narges Moeini*, Namhoon Kim*, Justin Romberg, and Sara Fridovich-Keil

Georgia Institute of Technology
{nmoeini3@, namhoon@, jrom@ece., sfk@}gatech.edu

Abstract. Volume denoising is a foundational problem in computational imaging, as many 3D imaging inverse problems face high levels of measurement noise. Inspired by the strong 2D image denoising properties of Field of Junctions (ICCV 2021), we propose a novel, fully volumetric 3D Field of Junctions (3D FoJ) representation that optimizes a junction of 3D wedges that best explain each 3D patch of a full volume, while encouraging consistency between overlapping patches. In addition to direct volume denoising, we leverage our 3D FoJ representation as a structural prior that: (i) requires no training data, and thus precludes the risk of hallucination, (ii) preserves and enhances sharp edge and corner structures in 3D, even under low signal to noise ratio (SNR), and (iii) can be used as a drop-in denoising representation via projected or proximal gradient descent for any volumetric inverse problem with low SNR. We demonstrate successful volume reconstruction and denoising with 3D FoJ across three diverse 3D imaging tasks with low-SNR measurements: low-dose X-ray computed tomography (CT), cryogenic electron tomography (cryo-ET), and denoising point clouds such as those from lidar in adverse weather. Across these challenging low-SNR volumetric imaging problems, 3D FoJ outperforms the evaluated classical denoisers, untrained neural denoisers, and denoisers trained only on noisy examples. Code is available at https://github.com/voilalab/3D-Field-of-Junctions.

Keywords: Volumetric denoising · Structural priors · Inverse problems · Tomography

1 Introduction

Denoising is a cornerstone of signal processing. While image denoising is well-studied, volume denoising remains an active research area as many modern computational imaging modalities are plagued by low signal to noise ratio (SNR), and denoising is a key step in many iterative algorithms for volumetric inverse problems. In medical imaging, low SNR arises from reduced radiation dose in computed tomography (CT) [20,28] or weaker magnetic fields in lower-cost magnetic resonance imaging (MRI) [3]. Micro-scale electron tomography imaging

* Equal contribution; order determined by coin flip.

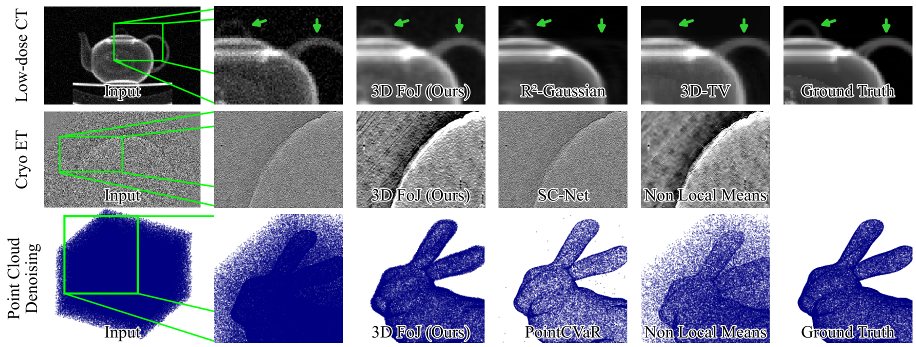


Fig. 1: Our 3D Field of Junctions (3D FoJ) is an effective volumetric denoiser across diverse inverse problems: low-dose computed tomography (*top row*), cryogenic electron tomography (*middle row*), and point cloud denoising (*bottom row*). Our cryo-ET experiment uses real data for which noiseless ground truth is not available. Only 3D FoJ successfully captures both the top and side handles of the teapot in low-dose CT.

uses low electron counts to prevent sample degradation [33], while macro-scale lidar sensors experience low SNR in adverse weather [42].

Existing approaches to denoising often are not well-suited to 3D signals, require extensive training data, or struggle to resolve sharp boundary structure at low SNR. Classical structural and statistical priors [7,38], and untrained neural networks like Deep Image Prior [34], Deep Decoder [13], and implicit neural representations [26,32], can effectively denoise volumes without any training data, but tend to blur edges and corners at low SNR. Data-driven priors including diffusion models [8,37] can recover sharp signal structure, but require large training datasets and are thus often restricted to learning priors over 2D images [10] or small 3D patches [30] rather than full 3D volumes. While recent volumetric reconstruction methods increasingly rely on large pretrained generative models, our work explores a complementary direction: explicit geometric modeling with strong structural inductive bias but no prior training. We demonstrate that, in extremely low-SNR regimes, such explicit structure can outperform the evaluated classical smoothness priors, untrained neural priors, and learned denoisers trained on noisy data.

Inspired by Field of Junctions (FoJ) [35], we introduce 3D Field of Junctions (3D FoJ), a fully explicit, geometric volume representation that preserves sharp corners and surfaces while rejecting noise. Compared to the original 2D formulation [35], 3D FoJ introduces a volumetric junction parameterization of 3D patches, an efficient implementation for large 3D volumes, and a proximal optimization framework enabling its use as a structural prior in diverse 3D inverse problems, as shown in Figure 1.

3D FoJ works by parameterizing a volume as overlapping 3D patches, with each patch modeled by a junction consisting of (i) a set of planes that divide the patch into 3D wedges, and (ii) a constant value inside each wedge. By changing the positions of the planes and the values of the wedges, a 3D junction can exactly represent a sharp corner, straight or bent boundary, or constant region, and can approximate a curved boundary. Hyper-parameters control the degree to which 3D FoJ penalizes sharp curvature. The junction parameters are optimized through a nonconvex yet empirically stable process that begins by greedily optimizing the parameters of each 3D patch individually in parallel, and then globally refines all junction parameters jointly by encouraging plane and wedge-value consistency across overlapping 3D patches. Together, the 3D FoJ parameterization and optimization process provide robust, boundary preserving volume denoising even under extremely low SNR. To summarize:

- We introduce 3D Field of Junctions (3D FoJ), an explicit, training-free structural prior that preserves sharp geometry even under extremely low SNR.
- We formulate 3D FoJ both as a direct volume denoiser and as a proximal regularizer for volumetric inverse problems, and show that it achieves strong performance across low-dose sparse-view CT reconstruction, cryo-ET denoising, and point cloud denoising.

Our implementation of 3D FoJ supports single and multi-GPU parallel processing of 3D patches and chunks of 3D patches to enable reconstruction and denoising of high-resolution volumes; code will be released upon publication.

2 Related Work

Many strategies have been proposed for volume denoising, broadly in the following categories.

Classical denoising priors. Classical structural priors such as bounded total variation (TV) [7], sparsity in a transform domain [23, 41], median filtering [38, 46], and nonlocal means filtering [5, 6, 11, 29] can be effective denoisers for both images and volumes. These require only a few hyper-parameters and no training data, but tend to blur edges under high noise. Our method is inspired by Field of Junctions [35], an explicit structural prior developed for robustly extracting sharp boundary structure in 2D images at low SNR. Our work is the first to apply similar principles to low-SNR 3D inverse problems.

Untrained neural priors. Untrained neural networks such as implicit neural representations (INRs) [25–27, 32], Deep Image Prior [34], and Deep Decoder [13] leverage the architectural restrictions of a neural network as a form of implicit regularization that can remove high-frequency noise through restricting model size or training time [14]. Like their classical counterparts, untrained neural networks can be effective denoisers and require no training data, but suffer loss of boundary structure at very low SNR.

Data-driven neural priors. Trained deep network denoisers can learn complex statistical features of a training dataset, allowing them to preserve edge structure even under high noise [15,16,45,47]. However, such models require large training datasets of clean [8,37] and/or noisy examples [1,19], that are often prohibitive to collect for 3D signals. While denoisers trained on 2D signals [9, 10, 18] or small 3D patches [30] can be retroactively applied to denoise a 3D volume, these applications are approximate and some 3D structure is often lost.

3 Preliminaries: 2D Field of Junctions

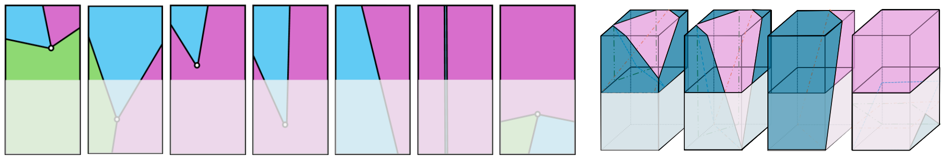
The Field of Junctions model [35] provides a unified framework for representing contours, corners, and multi-way junctions in 2D images, and robustly extracts this boundary structure even under low SNR. FoJ models each image patch with a parametric *M-junction*, a configuration of M wedges meeting at a freely located vertex, that captures local boundary geometry within a small neighborhood. Let $I : \Omega \to \mathbb{R}^K$ denote a K-channel image (*e.g.*, K=1 for grayscale), where $\Omega \subset \mathbb{R}^2$ is the image domain. The image I is divided into overlapping patches $\{I_i\}_{i=1}^N$ of size $R \times R$, with stride s in each dimension. For the ith patch, the junction parameters are

$$\boldsymbol{\theta}_i = (\phi_i, \boldsymbol{p}_i^{(0)}), \qquad \phi_i = (\phi_i^{(1)}, \dots, \phi_i^{(M)}), \tag{1}$$

where $\phi_i^{(k)}$ denotes the orientation of the kth boundary, and $\boldsymbol{p}_i^{(0)} = (x_i^{(0)}, y_i^{(0)})$ is the vertex position, which may lie inside or outside the patch. The number of wedges M determines the local boundary complexity: for example, when M=3, the model captures three intersecting lines that can form T- or Y-junctions, but can also model straight or bent edges or uniform regions when one or more lines coincide (see Figure 2(a)). Each wedge j in junction i is also endowed with a constant (optimizable) color value $c_i^{(j)} \in \mathbb{R}^K$. The junction parameters $\Theta = \{\theta_i\}$ and colors $C = \{c_i^{(j)}\}$ are estimated by minimizing a global objective that enforces both patch fidelity and inter-patch consistency; more details are provided in the appendix.

4 Method: 3D Field of Junctions

Our 3D FoJ is inspired by 2D Field of Junctions (FoJ) [35] but operates natively in 3D for volumetric signals. FoJ models image patches as junctions parameterized by intersecting lines and constant-color wedges. In three dimensions, we generalize these boundaries to planar surfaces that divide a 3D patch into constant-value regions. Like its 2D counterpart, our 3D FoJ uses a unified junction parameterization model to represent 3D patches with many possible shapes, including constant regions, straight and bent edges, and multi-surface junctions.



(a) 2D FoJ (M=3). **(b) 3D FoJ** (M=2 shown).

Fig. 2: Junction schematics for Field of Junctions (FoJ) in 2D (left) and our 3D FoJ (right). (a) In 2D, [35] models each patch as an M-junction: a vertex and M angles define M uniform-color wedges. Allowing the vertex to lie inside or outside the patch enables a unified representation of edges, corners, junctions, and homogeneous regions. **(b)** In 3D, we visualize the two-region case (M=2) with different configurations of intersecting planes inside a volumetric patch. We show M=2 for visual clarity; our experiments use M=3 regions.

4.1 Local Volumetric Model

Each volumetric patch $V_i : \Omega_i \to \mathbb{R}^K$ is modeled as a 3D region partitioned by three planes that intersect at a common vertex $\boldsymbol{v}_i^{(0)} \in \mathbb{R}^3$, which may lie inside or outside the 3D patch. Here, $\Omega_i \subset \mathbb{R}^3$ denotes the spatial domain of the 3D patch. The full 3D volume is divided into overlapping patches $\{V_i\}_{i=1}^N$, each of size $R \times R \times R$, with stride s in each dimension. Each voxel intensity is represented as a K-dimensional vector (*e.g.*, K=1 for grayscale data). The parameters of junction i are defined as

$$\boldsymbol{\gamma}_i = (\boldsymbol{\rho}_i^{(\ell)}, \boldsymbol{v}_i^{(0)}), \qquad \boldsymbol{\rho}_i^{(\ell)} = (\theta_i^{(\ell)}, \phi_i^{(\ell)}), \quad \ell = 1, 2, 3 \tag{2}$$

where each pair $(\theta_i^{(\ell)}, \phi_i^{(\ell)})$ specifies the polar and azimuthal angles of the ℓth plane normal, and $\boldsymbol{v}_i^{(0)} = (x_i^{(0)}, y_i^{(0)}, z_i^{(0)})$ denotes the vertex position at which the three planes intersect. Since the vertex may freely move inside or outside the 3D patch, the junction can represent a wide variety of 3D structures as illustrated in Figure 2b. The three planes divide each patch into regions (generalized 3D wedges), each of which is modeled with a constant parameter value (intensity, color, density, etc.) $c_i^{(j)} \in \mathbb{R}^K$.

4.2 Global Objective Function

We jointly optimize the parameters $\Gamma = (\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_N)$ and regional intensities $C = (c_1, \ldots, c_N)$ to minimize a global objective that enforces both local data fidelity and inter-patch coherence across the entire volume. Our volumetric extension of the 2D FoJ objective is given by:

$$\begin{aligned} \min_{\Gamma, C} \quad & \sum_{i,j} \iiint u_{\boldsymbol{\gamma}_i}^{(j)}(\boldsymbol{v}) \, \|V_i(\boldsymbol{v}) - c_i^{(j)}\|^2 \, d\boldsymbol{v} + \lambda_B \sum_i \iiint_{\Omega_i} [B_i(\boldsymbol{v}) - \widehat{B}(\boldsymbol{v})]^2 \, d\boldsymbol{v} \\ & + \lambda_C \sum_{i,j} \iiint u_{\boldsymbol{\gamma}_i}^{(j)}(\boldsymbol{v}) \, \|c_i^{(j)} - \widehat{V}(\boldsymbol{v})\|^2 \, d\boldsymbol{v}. \end{aligned} \tag{3}$$

Here, $u^{(j)}_{\boldsymbol{\gamma}_i}(\boldsymbol{v}) \in \{0,1\}$ denotes membership of a voxel $\boldsymbol{v} = (x,y,z)$ in the jth volumetric region of patch i, $V_i(\boldsymbol{v})$ represents the observed volume within patch i (with domain Ω_i), and $c_i^{(j)}$ is the constant value associated with region j of the junction model for patch i. The first term enforces local reconstruction accuracy of each patch using its piecewise-constant junction model. The second term, weighted by $\lambda_B \geq 0$, aligns local and global boundary estimates through the boundary fields $B_i(\boldsymbol{v})$ and $\widehat{B}(\boldsymbol{v})$, which will be defined shortly in the *Global Maps* paragraph. This term encourages continuity of surface structures across overlapping patches. The third term, weighted by $\lambda_C \geq 0$, promotes consistent regional appearance by comparing each patch's reconstructed intensity to the global field $\widehat{V}(\boldsymbol{v})$ obtained by averaging overlapping patch reconstructions.

Global Maps. The global volumetric color and boundary fields are defined by aggregating the contributions of all patches that contain a given voxel $\boldsymbol{v}$. The global color field $\widehat{V}(\boldsymbol{v})$ is given by

$$\widehat{V}(\boldsymbol{v}) = \frac{1}{|\mathcal{N}_{\boldsymbol{v}}|} \sum_{i \in \mathcal{N}_{\boldsymbol{v}}} \sum_{j=1}^{M} u^{(j)}_{\gamma_i}(\boldsymbol{v})\, c_i^{(j)}, \tag{4}$$

where $\mathcal{N}_{\boldsymbol{v}}$ denotes the set of all patches that include voxel $\boldsymbol{v}$, and $|\mathcal{N}_{\boldsymbol{v}}|$ is its cardinality, the number of patches that include voxel $\boldsymbol{v}$. This operation computes a weighted average of all region intensities overlapping at $\boldsymbol{v}$, yielding a smooth global volume reconstruction. Similarly, the relaxed global boundary field $\widehat{B}(\boldsymbol{v})$ averages local boundary predictions from all overlapping patches:

$$\widehat{B}(\boldsymbol{v}) = \frac{1}{|\mathcal{N}_{\boldsymbol{v}}|} \sum_{i \in \mathcal{N}_{\boldsymbol{v}}} B_i(\boldsymbol{v}), \tag{5}$$

where $B_i(\boldsymbol{v}) \in \{0,1\}$. This field encodes the presence or absence of a boundary plane at each voxel, effectively representing the local boundary consensus across overlapping patches. Although the binary formulation provides a clear geometric interpretation, it is not differentiable and therefore unsuitable for gradient-based optimization. To enable efficient and stable optimization of the junction parameters $\boldsymbol{\gamma}_i$, we replace the binary region indicators and boundary maps with differentiable, smooth counterparts, introduced in the following subsection.

4.3 Smooth Indicator Functions

To enable gradient-based optimization, the region indicator functions in Equation (3) must be differentiable. We therefore define soft indicator functions using signed distance values relative to the three separating planes of each 3D patch. For voxel $\boldsymbol{v} = (x,y,z)$, the signed distance to the ℓth plane is given by

$$d_\ell(\boldsymbol{v}) = \langle (x,y,z) - (x_i^{(0)}, y_i^{(0)}, z_i^{(0)}),\, \mathbf{n}_i^{(\ell)} \rangle, \tag{6}$$

where $\mathbf{n}_i^{(\ell)}$ is the normal vector of the ℓth plane, parameterized by its polar and azimuthal angles $(\theta_i^{(\ell)}, \phi_i^{(\ell)})$. Inspired by the 2D FoJ formulation [35], we use a Heaviside function as a differentiable approximation of the binary region assignment $H_\eta(d) = \frac{1}{2}\Big(1 + \frac{2}{\pi}\arctan\frac{d}{\eta}\Big)$, where $\eta > 0$ controls the transition sharpness across the plane boundaries. The differentiable volumetric region indicators are then defined as

$$\begin{aligned}
u^{(1)}_{\boldsymbol{\gamma}_i}(\boldsymbol{v}) &= H_\eta(d_1(\boldsymbol{v}))\,[1 - H_\eta(d_2(\boldsymbol{v}))]\,[1 - H_\eta(d_3(\boldsymbol{v}))],\\
u^{(2)}_{\boldsymbol{\gamma}_i}(\boldsymbol{v}) &= [1 - H_\eta(d_1(\boldsymbol{v}))]\,H_\eta(d_2(\boldsymbol{v}))\,[1 - H_\eta(d_3(\boldsymbol{v}))],\\
u^{(3)}_{\boldsymbol{\gamma}_i}(\boldsymbol{v}) &= [1 - H_\eta(d_1(\boldsymbol{v}))]\,[1 - H_\eta(d_2(\boldsymbol{v}))]\,H_\eta(d_3(\boldsymbol{v})),
\end{aligned} \tag{7}$$

for our parameterization with 3 regions per patch. By following a similar pattern and enumerating over all 8 binary arrangements with 3 planes, our 3D junction model can represent up to 8 distinct regions per 3D patch. These functions softly assign each voxel to one of these three regions based on its position relative to the planes, providing continuous gradients for optimization.

We note that partitioning a 3D patch into M disjoint regions is more geometrically complex than partitioning a 2D patch. Our 3D FoJ formulation based on three intersecting planes naturally divides a 3D patch into eight regions; we work with fewer (typically $M = 3$) regions by only assigning nonzero values to a subset of these eight regions. In particular, the region assignments in Equation (7) may leave some portion of the patch outside any of the $M = 3$ regions and thus with default value zero. Effectively we find that these zero-valued regions do not negatively impact reconstruction or denoising quality; we use $M = 3$ in our experiments to balance the expressivity of having more regions per junction with the increased computational and memory cost. Ablation studies over the choice of M are provided in Table 3.

Smooth Boundary Map. To compare and align local surfaces across overlapping patches, we use a differentiable boundary-strength field from the 2D FoJ formulation [35]:

$$B_i^{(\delta)}(\boldsymbol{v}) = \pi\,\delta\,H'_\delta\Big(\min_{i\in 1,2,3}\{\,|d_i(\boldsymbol{v})|\}\Big) \in [0,1]. \tag{8}$$

Here $H_\delta(d)$ is the regularized Heaviside function and $H'_\delta(d)$ is its derivative with respect to d. The input d to H'_δ is the distance of voxel $\boldsymbol{v}$ to the nearest separating plane; thus $B_i^{(\delta)}(\boldsymbol{v})$ attains high values near predicted surfaces and smoothly decays with distance. The parameter $\delta > 0$ controls the transition width: larger δ yields broader, softer boundaries; smaller δ produces sharper peaks. This soft boundary field serves as a differentiable surrogate for a binary boundary map, and is used in the boundary-consistency term of Equation (3) to encourage inter-patch alignment of boundary planes.

4.4 Optimization Procedure

Reconstructing the volumetric Field of Junctions involves minimizing a nonconvex objective over the 3D junction parameters $\boldsymbol{\gamma}_i = (\boldsymbol{\rho}_i, \boldsymbol{v}_i^{(0)})$. Similar to the 2D case [35], we perform this nonconvex optimization in two stages: an initialization stage that estimates local geometry for each patch independently, and a refinement stage that jointly optimizes all patches using differentiable indicator functions. A detailed step-by-step version of this optimization process is provided in the Appendix.

Initialization Step. The initialization stage provides a coarse yet reliable estimate of the plane orientations and their vertex positions within each patch. Each junction is initialized with the vertex $\boldsymbol{v}_i^{(0)}$ placed at the center of the patch, and the plane orientation parameters $\boldsymbol{\rho}_i^{(\ell)} = (\theta_i^{(\ell)}, \phi_i^{(\ell)})$ are optimized using a discrete coordinate-descent search. For each orientation pair (θ, ϕ), the data fidelity term from Equation (3) is evaluated over a uniform angular grid, and the minimizing direction is selected. After the optimal plane orientations are estimated, the vertex position $\boldsymbol{v}_i^{(0)} = (x_i^{(0)}, y_i^{(0)}, z_i^{(0)})$ is refined by scanning a small neighborhood around its current position to minimize the same data term. This coordinate-wise update is repeated cyclically across the vertex and all planes until convergence, defined by a fixed number of iterations. This procedure is robust to noise and computationally efficient, as all patches can be processed in parallel.

Refinement Step. After initialization, all patches are jointly refined using gradient-based optimization of the full objective function in Equation (3). The binary region indicators and boundary maps are replaced by their differentiable counterparts (defined in Section 4.3), enabling end-to-end optimization through the junction parameters $\boldsymbol{\gamma}_i$. We use first-order iterative optimization (e.g., Adam) to update $\theta_i^{(\ell)}$, $\phi_i^{(\ell)}$, and $\boldsymbol{v}_i^{(0)}$, while the region intensities $c_i^{(j)}$ are updated analytically at each iteration using the following closed-form solution:

$$c_i^{(j)} = \frac{\iiint u_{\boldsymbol{\gamma}_i}^{(j)}(\boldsymbol{v}) \, [V_i(\boldsymbol{v}) + \lambda_C \, \widehat{V}(\boldsymbol{v})] \, d\boldsymbol{v}}{(1 + \lambda_C) \iiint u_{\boldsymbol{\gamma}_i}^{(j)}(\boldsymbol{v}) \, d\boldsymbol{v}}. \tag{9}$$

During refinement, the regularization weights λ_B and λ_C are gradually increased from small initial values to their final targets, ensuring stable convergence and avoiding premature over-smoothing. This joint optimization progressively improves both the geometric accuracy of the surfaces and their spatial consistency across overlapping patches, yielding a coherent 3D Field of Junctions.

4.5 Solving Inverse Problems with 3D FoJ

Our 3D FoJ model can be used as a drop-in structural prior in any volumetric inverse problem; we showcase it in the context of low-dose sparse-view tomographic reconstruction. The reconstruction problem is formulated as

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}) + g(\boldsymbol{x}), \tag{10}$$

where $\boldsymbol{x} \in \mathbb{R}^{D\times H\times W\times K}$ denotes the volumetric image to be reconstructed. The data fidelity term $f(\boldsymbol{x})$ encourages consistency with the measured projections:

$$f(\boldsymbol{x}) = \tfrac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2, \tag{11}$$

where $\boldsymbol{A}$ is the system matrix (for tomography, the Radon transform) and $\boldsymbol{b}$ is the vector of projection measurements. The 3D FoJ regularization term

$$g(\boldsymbol{x}) = \min_{\Gamma, C} \|\boldsymbol{x} - R(\Gamma, C)\|^2 \tag{12}$$

encourages geometric consistency in voxel space, with $R(\Gamma, C)$ denoting the reconstructed volume parameterized by 3D FoJ parameters Γ and C. We solve Equation (10) via a proximal-gradient method:

$$\boldsymbol{x}^{(k+1)} = \mathrm{prox}_{\lambda g}\Big(\boldsymbol{x}^{(k)} - \lambda \nabla f(\boldsymbol{x}^{(k)})\Big), \tag{13}$$

which alternates between a gradient descent step on the data term and a proximal step defined by the FoJ prior. The scalar $\lambda > 0$ serves a dual role: it acts as the gradient descent step size on the data fidelity term $f(\boldsymbol{x})$, and simultaneously controls the strength of the 3D FoJ regularization through the proximal operator. As $\lambda \downarrow 0$, the update approaches a pure gradient descent step with negligible 3D FoJ influence, and the reconstruction converges toward the conventional least-squares solution. Conversely, as $\lambda \uparrow \infty$, the proximal step dominates and the solution is driven toward the 3D FoJ manifold, yielding reconstructions that closely follow the 3D FoJ structural prior $R(\Gamma, C)$. In practice, λ is tuned to balance data fidelity and 3D FoJ regularization. The two updates can be written explicitly as:

$$\boldsymbol{x}^{(k+\frac{1}{2})} = \boldsymbol{x}^{(k)} - \lambda\, \nabla f\big(\boldsymbol{x}^{(k)}\big). \tag{14}$$

$$\boldsymbol{x}^{(k+1)} = \arg\min_{\boldsymbol{x}} \Big[g(\boldsymbol{x}) + \tfrac{1}{2\lambda}\, \|\boldsymbol{x} - \boldsymbol{x}^{(k+\frac{1}{2})}\|^2\Big]. \tag{15}$$

The first half-iteration (Equation (14)) corresponds to a conventional gradient descent update enforcing data consistency, and the second half-iteration (Equation (15)) corresponds to a proximal update to encourage consistency with the denoised 3D FoJ representation. In practice, we approximate this proximal step efficiently by performing a single 3D FoJ update (one iteration of the initialization step and one iteration of the refinement step) per outer iteration, initialized from the intermediate volume. Empirically, we find that a single FoJ update at each proximal step is sufficient to guide the solution toward the FoJ manifold while keeping the overall reconstruction computationally tractable. This proximal formulation enables joint recovery of the voxel intensity field and the underlying junction geometry within each iteration, using noisy tomographic measurements rather than direct access to a noisy volume. We emphasize that, while we present results on a tomographic inverse problem, the same proximal gradient formulation would enable 3D FoJ to be used as a drop-in structural prior to regularize any volumetric inverse problem. We also note that, while the 3D FoJ optimization process is nonconvex, empirically we observed stable optimization across all experiments. Empirical update-norm convergence curves for the proximal reconstruction procedure are provided in the appendix.

5 Results

To illustrate the broad applicability of 3D FoJ as a structural prior for volume reconstruction and denoising, we perform experiments on three diverse volume imaging tasks that share the key feature of inherent measurement noise: low-dose X-ray computed tomography (CT), cryogenic electron tomography (cryo-ET), and point cloud denoising. Low-dose CT and cryo-ET face severe shot noise due to low X-ray and electron beam intensities, which are restricted to avoid radiative damage and sample degradation, respectively. Point clouds reconstructed from sensors such as lidar can be corrupted by outliers due to sensor noise or adverse weather. Our experiments frame X-ray CT as an inverse problem, where we use 3D FoJ inside proximal gradient descent to reconstruct a denoised volume from raw noisy projections. For cryo-ET and point cloud denoising, we start with a noisy volume and showcase 3D FoJ as a direct 3D denoiser.

5.1 Low-dose X-ray Computed Tomography (CT)

To evaluate the reconstruction performance of our 3D FoJ for limited-angle and low-dose 3D CT reconstruction, we conduct experiments on five synthetic volumetric datasets—*pepper, teapot, jaw, foot,* and *engine*—originally introduced in the R^2-Gaussian paper [43]. We reconstruct each object at resolution 256^3 from 20 non-uniform projection views, each with resolution 256^2. To emulate low-dose X-ray acquisition, we add Poisson-distributed noise to the synthetic projection data, where the simulated photon count per detector pixel determines the effective signal-to-noise ratio (SNR).

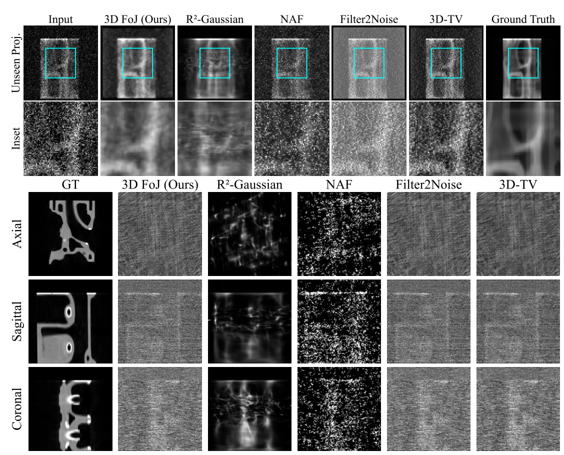


Fig. 3: Comparison of unseen projection views synthesized from reconstructed 3D volumes (top) and slice views of reconstructed 3D volumes (bottom) for the *engine* dataset under extremely low-SNR conditions (*P50*, severe shot noise with only 50 photons per detector pixel).

We simulate three photon-count levels, denoted as *P50*, *P100*, and *P1000*, corresponding respectively to increasing SNR regimes. Here, the notation *Pn* indicates an expected photon count of n per pixel (in air), with *P50* representing the lowest SNR (extreme noise) and *P1000* representing the highest SNR (but still noisy) condition. Additional qualitative comparisons of unseen projection views and reconstructed axial, sagittal, and coronal slice views across all datasets and noise levels further demonstrate the consistent structural fidelity and noise robustness of 3D FoJ.

We compare our 3D FoJ reconstructions against baselines that do not require paired clean/noisy 3D training data: an instance-optimized explicit representation, R^2-Gaussian [43]; an instance-optimized neural representation, Neural Attenuation Fields (NAF) [44]; a self-supervised denoiser trained only on noisy examples, Filter2Noise [31]; and a classical volumetric regularizer, 3D total variation (TV) regularization [4]. Fully supervised volumetric denoisers trained with matched clean/noisy CT data are outside our evaluation setting, as are large generative priors that require external 2D or patch-based training. Our goal is to compare against methods that can be applied at the full-volume scale under similar computational budgets and without an external training set of paired clean and noisy volumes.

Metric	Method	P50	P100	P1000
2D MS-SSIM	3D-TV	0.703	0.739	0.829
	Filter2Noise	0.663	0.689	0.695
	NAF	0.423	0.467	0.569
	R^2-Gaussian	0.574	0.577	0.652
	3D FoJ (ours)	0.754	0.788	0.838
3D PSNR	3D-TV	13.79	15.31	16.96
	Filter2Noise	11.61	15.13	14.64
	NAF	14.17	14.67	15.66
	R^2-Gaussian	15.29	15.29	15.87
	3D FoJ (ours)	15.88	16.14	16.72

Table 1: Average MS-SSIM on 2D projections (360 views) and 3D PSNR on volumes across all CT datasets (*engine, foot, jaw, pepper*, and *teapot*) for each noise level. The best method is highlighted for each noise level and metric.

We evaluate reconstruction fidelity using both volumetric and projection-based metrics. Specifically, we report the 3D peak signal-to-noise ratio (3D PSNR) computed on the reconstructed volumes and the multi-scale structural similarity (MS-SSIM) [36] measured on 360 2D projections (one per degree along a circular trajectory). Table 1 summarizes the average MS-SSIM and 3D PSNR across all datasets (*engine*, *foot*, *jaw*, *pepper*, and *teapot*) and noise levels. As shown, 3D FoJ achieves the strongest average structural fidelity across all noise levels, with particularly clear gains under low-SNR (low photon count) conditions. For volumetric PSNR, 3D FoJ performs best at P50 and P100 and remains competitive at P1000. Figure 3 provides qualitative comparisons of the reconstructed geometry under the low-SNR (*P50*) condition. The top row shows an unseen projection synthesized from each reconstructed 3D volume, which tests global consistency for a view not included in the input angles. The bottom row shows slice views of the reconstructed volumes, where 3D FoJ preserves cleaner surfaces, sharper edges, and more faithful structural details than all baselines. The full quantitative comparison across all datasets and noise levels, including both projection-based MS-SSIM and volumetric 3D PSNR metrics, is provided in Table 4 in the appendix.

5.2 Cryogenic Electron Tomography (cryo-ET)

To evaluate the denoising capability of our method on real-world electron microscopy data, we conducted experiments on four cryo-ET datasets: *centriole* from [17] and *mitochondria*, *vesicle*, and *VEEV* from [40]. We take as input the noisy $1024 \times 1024 \times 300$ volumes reconstructed using the `tilt` program in IMOD [17] as described in the SC-Net pipeline [40]. Noise in the input volumes is the result of both electron shot noise, because the electron beam intensity is limited to avoid sample degradation, and missing wedge artifacts, because cryo-ET projections are geometrically constrained to measure limited angles.

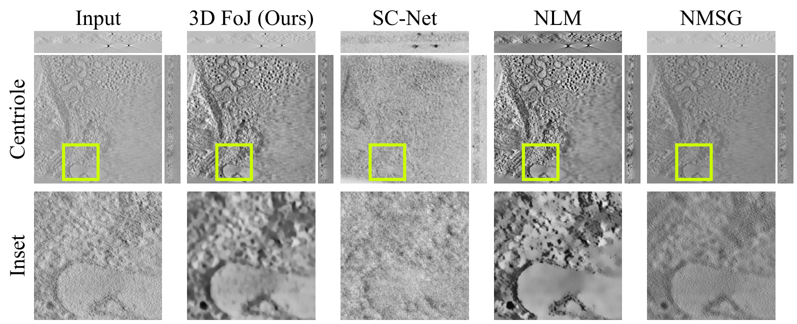
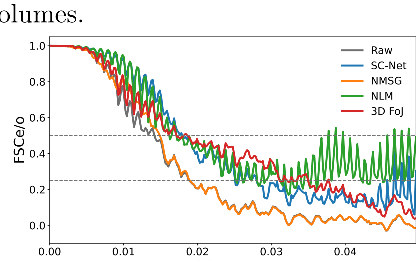


Fig. 4: Visual denoising results for the cryo-ET *centriole* volume, sliced in each dimension. 3D FoJ preserves structural details that are overly smoothed by other methods.

Since noiseless ground-truth volumes are unavailable for real cryo-ET data, we use Fourier Shell Correlation (FSC) [12] for quantitative evaluation, following standard practice in cryo-ET reconstruction. FSC measures the frequency-domain consistency between independently reconstructed subsets of the available projections. For the IMOD *centriole* dataset, we split projections into even and odd halves following SC-Net [40], reconstruct both halves in IMOD, denoise each half independently, and compute the half-map FSC $\rho_{\text{half}}(k)$ between the two denoised half-maps. We report the corrected even/odd FSC curves using the correction adopted from [40]: $\text{FSC}_{\text{e/o}}(k) = 2\rho_{\text{half}}(k)/(1 + \rho_{\text{half}}(k))$. Under this correction, the equivalent of the gold-standard 0.143 half-map FSC threshold is 0.25.

We compared our 3D FoJ method against SC-Net [40], Noise Modeling and Sparsity Guidance (NMSG) [39], and nonlocal means denoising (NLM) [5] as representative neural [39, 40] and classical [5] denoisers applicable to cryo-ET volumes.



	Resolution (Å) ↓	
Method	$\text{FSC}_{\text{e/o}}$ 0.5	$\text{FSC}_{\text{e/o}}$ 0.25
SC-Net	**55.60**	39.86
NMSG	67.10	52.76
NLM	61.73	39.75
3D FoJ	58.24	**32.14**

Fig. 5: Even/odd FSC evaluation on the IMOD *centriole* dataset.

As shown in Figure 5, 3D FoJ achieves the best $\text{FSC}_{\text{e/o}}$ resolution at the 0.25 threshold and remains competitive with SC-Net at the 0.5 threshold, while requiring no training data. Qualitative volume denoising results in Figure 4 show that 3D FoJ achieves the best visual balance of denoising, contrast enhancement, and detail preservation. Further qualitative results on additional real cryo-ET volumes are presented in the appendix, illustrating the competitive denoising performance of 3D FoJ relative to alternative approaches.

5.3 Point Cloud Denoising

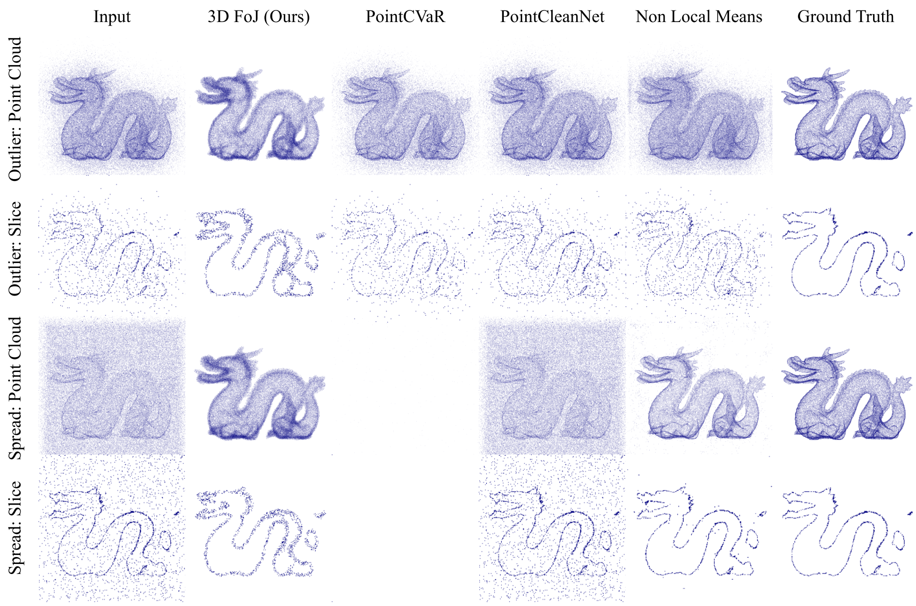


Fig. 6: Denoising results for the *dragon* point cloud with 60% *outlier* noise and 500k random *spread* points. Slices show that 3D FoJ effectively removes noise both inside and outside the object surface. At this extreme level of *spread* noise, PointCVaR rejects nearly all points as outliers.

We further evaluate our method on point cloud denoising tasks, using a total of 28 point clouds provided by PointCleanNet [24]. We consider two noise models: (i) outlier noise modeled on what might be expected from lidar sensor noise or miscalibration, in which noise points are concentrated near object surfaces, and (ii) synthetic real-world noise similar to what would be expected from a lidar sensor in rain or snow. We refer to the first noise model as *outlier* and the second as *spread*. Our *outlier* removal experiments use the same noise configurations as PointCleanNet [24], focusing on the four outlier ratios (10%, 30%, 60%, 90%) in their dataset. Our *spread* denoising experiments use simulated inclement-weather lidar noise modeled on a similar experiment from PointCleanNet. For each point cloud, we randomly add 40k, 100k, 200k, or 500k noisy points within an axis-aligned bounding box, defined by padding along each of the x, y, and z axes by 10 meters.

We compare our 3D FoJ method against PointCleanNet [24], PointCVaR [22], and nonlocal means (NLM) [5] as representative strong and diverse baselines for point cloud denoising. PointCleanNet uses a neural network for denoising, PointCVaR follows a statistical approach to reject outlier points, and nonlocal means averages structurally similar but disjoint 3D patches to reduce noise. Because 3D FoJ and NLM are configured for voxel rather than point cloud denoising, for these methods we convert the noisy input point cloud into a 256^3 voxel grid before denoising. After denoising, we convert the denoised voxel grids from these methods back into point clouds by placing points at the highest-intensity 100k voxel coordinates, to match the point-based denoisers (PointCleanNet and PointCVaR) which are configured to output 100k points. For fairness, all voxel-based methods use the same voxelization and de-voxelization protocol, with full details provided in the appendix.

We report Chamfer Distance (CD) [2,21] in Table 2 to quantify geometric consistency between denoised and ground truth point clouds. We find that 3D FoJ is a robust point cloud denoiser for both types of noise and across all noise levels, extracting robust shape structure even under low SNR. Visual results are shown in Figure 6. Additional quantitative evaluations under outlier noise and spread noise settings are reported in the appendix, further demonstrating the robustness and consistency of 3D FoJ across a wide range of corruption levels.

Task	Level	PointCVaR	PointCleanNet	NLM	3D FoJ (Ours)
Outlier	10%	2.15 ± 6.47	2.17 ± 6.44	5.02 ± 16.73	0.15 ± 0.40
	30%	6.54 ± 19.39	6.52 ± 19.25	7.82 ± 24.15	0.16 ± 0.42
	60%	12.93 ± 38.26	12.98 ± 38.19	13.41 ± 40.62	0.22 ± 0.59
	90%	19.64 ± 57.82	19.57 ± 57.56	18.35 ± 54.88	1.74 ± 5.68
Spread	40k	154.41 ± 396.20	62.33 ± 191.76	87.05 ± 279.53	0.16 ± 0.41
	100k	0.44 ± 0.66	108.50 ± 333.13	138.14 ± 438.95	0.17 ± 0.44
	200k	256.72 ± 661.88	144.94 ± 445.71	168.08 ± 527.55	0.18 ± 0.48
	500k	515.25 ± 1327.14	181.12 ± 557.22	102.84 ± 334.54	0.48 ± 1.79

Table 2: Point cloud denoising (Chamfer-L2), mean±std across shapes. Best per noise level is highlighted. PointCVaR is competitive at moderate spread noise but is less robust across noise levels.

5.4 Ablation Study

We perform an ablation study on the *engine* CT dataset at the lowest SNR level (P50), to assess the influence of patch size, stride, batch size, and number of regions on reconstruction quality and runtime. For this experiment, we first reconstruct the 3D volume from its sparse, noisy projections using the standard CGLS algorithm, and then fit 3D FoJ to the resulting volume under different parameter settings.

# regions	Batch size	Stride	Patch size	MS-SSIM	3D PSNR	Runtime
3	6	1	4	0.755	11.84	14.3 min
3	6	1	6	0.732	14.29	31.6 min
3	6	1	8	0.722	15.76	47.3 min
3	6	2	4	0.762	11.33	7.2 min
3	6	2	6	0.784	14.37	10.5 min
3	6	2	8	0.796	16.18	13.1 min
3	6	2	10	0.798	17.17	17.2 min
3	6	2	12	0.800	17.70	24.7 min
3	1	2	10	0.798	17.17	37.6 min
3	2	2	10	0.798	17.17	25.6 min
3	3	2	10	0.798	17.17	24.5 min
3	4	2	10	0.798	17.17	20.3 min
3	5	2	10	0.798	17.17	18.5 min
3	6	2	10	0.798	17.17	17.2 min
2	6	2	10	0.805	17.03	15.2 min
3	6	2	10	0.798	17.17	17.2 min
4	6	2	10	0.798	16.70	20.3 min
5	6	2	10	0.797	16.43	21.5 min
6	6	2	10	0.790	16.50	22.8 min
7	6	2	10	0.799	16.37	25.1 min
8	6	2	10	0.797	16.12	27.7 min

Table 3: Ablation study over 3D FoJ hyperparameters on the *engine* CT dataset under extreme shot noise (*P50*).

Results are reported in Table 3. Increasing the patch size generally improves reconstruction quality in low-SNR settings, at the cost of increased runtime. We expect that the optimal patch size is likely an increasing function of noise level, as larger patches can average out more noise. Using a stride of 2 achieves a favorable trade-off between reconstruction fidelity and efficiency; slightly higher fidelity is possible with a stride of 1, but at a substantial computational price.

The batch size for parallel processing primarily affects runtime, with no impact on reconstruction accuracy. Varying the number of regions per junction shows that using two or three regions provides the best balance of reconstruction quality and speed. It is possible that higher junction complexity could prove beneficial for particularly complex structures, or with additional steps of refinement optimization. Our default configuration uses a patch size of 10, stride of 2, batch size of 6, and $M = 3$ regions per junction. The complete set of ablation results, including additional configurations for initialization and refinement iterations, is provided in the appendix. Additional task-scale runtime and peak-memory measurements are provided in the appendix.

6 Discussion

Our experiments confirm that our proposed 3D Field of Junctions parameterization is an effective prior for 3D denoising and noisy volumetric inverse problems, capable of extracting sharp boundary structure in 3D even under extremely low SNR. We emphasize that 3D FoJ is fully explicit, with all parameters endowed with physical structural interpretation. It requires neither training data nor neural networks, can run on a single GPU or parallelize across several, and has fairly stable performance across hyperparameter settings and noise levels. It robustly extracts 3D structure across different types of noise, and works well in both direct volume denoising and indirect regularization of low-SNR volumetric inverse problems, outperforming the evaluated classical, untrained neural, and no-clean-target learned alternatives.

Our results are subject to three main limitations. First, we did not tune the various 3D FoJ hyperparameters (patch size, stride, number of regions M per patch, or the smoothness parameters η and δ) separately for each task. While this ensures our results reflect expected performance on new tasks, stronger performance is likely possible with additional hyperparameter tuning, especially for different noise levels and dataset resolutions. A second limitation is computational cost. For a moderate-size volume (256^3), fitting 3D FoJ runs in a few minutes on an NVIDIA A6000 GPU. However, memory usage increases with both patch and volume size, necessitating chunked or parallel processing, which our implementation supports, for large volumes. We provide task-scale runtime and peak-memory measurements for our CT, cryo-ET, and point-cloud denoising experiments in the appendix. Finally, 3D FoJ imposes a local piecewise-planar inductive bias: curved or smoothly varying structures are represented by locally planar pieces. This bias is useful for suppressing noise and preserving sharp interfaces, and is well matched to many man-made objects and membrane-dominated biological structures. However, for highly curved structures, performance depends on the patch size and the number of active regions per patch. The curved-tube study in the appendix shows that the default $M = 3$ setting preserves curved geometry when the FoJ patch size is local relative to curvature, while using all $M = 8$ sectors better handles large patches or high curvature, at higher memory and runtime cost.

References

1. Aali, A., Arvinte, M., Kumar, S., Tamir, J.I.: Solving inverse problems with score-based generative priors learned from noisy data. In: 2023 57th Asilomar Conference on Signals, Systems, and Computers. pp. 837–843 (2023). https://doi.org/10.1109/IEEECONF59524.2023.10477042
2. Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L.: Learning representations and generative models for 3d point clouds. In: International conference on machine learning. pp. 40–49. PMLR (2018)
3. Arnold, T.C., Freeman, C.W., Litt, B., Stein, J.M.: Low-field mri: clinical promise and challenges. Journal of Magnetic Resonance Imaging **57**(1), 25–44 (2023)
4. Barbero, A., Sra, S.: Modular proximal optimization for multidimensional total-variation regularization. Journal of Machine Learning Research **19**(56), 1–82 (2018), http://jmlr.org/papers/v19/13-538.html
5. Bruns, S., Stipp, S., Sørensen, H.: Looking for the signal: A guide to iterative noise and artefact removal in x-ray tomographic reconstructions of porous geomaterials. Advances in Water Resources **105**, 96–107 (2017)
6. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Multiscale modeling & simulation **4**(2), 490–530 (2005)
7. Candès, E.J., Romberg, J., Tao, T.: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on information theory **52**(2), 489–509 (2006)
8. Chung, H., Kim, J., Mccann, M.T., Klasky, M.L., Ye, J.C.: Diffusion posterior sampling for general noisy inverse problems. International Conference on Learning Representations (2023)
9. Chung, H., Lee, S., Ye, J.C.: Decomposed diffusion sampler for accelerating large-scale inverse problems. International Conference on Learning Representations (2024)
10. Chung, H., Ryu, D., McCann, M.T., Klasky, M.L., Ye, J.C.: Solving 3d inverse problems using pre-trained 2d diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22542–22551 (2023)
11. Coupé, P., Yger, P., Prima, S., Hellier, P., Kervrann, C., Barillot, C.: An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images. IEEE transactions on medical imaging **27**(4), 425–441 (2008)
12. Harauz, G., van Heel, M.: Exact filters for general geometry three dimensional reconstruction. Optik. **73**(4), 146–156 (1986)
13. Heckel, R., Hand, P.: Deep decoder: Concise image representations from untrained non-convolutional networks. International Conference on Learning Representations (2019)
14. Heckel, R., Soltanolkotabi, M.: Compressive sensing with un-trained neural networks: Gradient descent finds a smooth approximation. In: International conference on machine learning. pp. 4149–4158. PMLR (2020)
15. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
16. Jia, F., Wong, W.H., Zeng, T.: 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)
17. Kremer, J.R., Mastronarde, D.N., McIntosh, J.R.: Computer visualization of three-dimensional image data using imod. Journal of structural biology **116**(1), 71–76 (1996)

18. Lee, S., Chung, H., Park, M., Park, J., Ryu, W.S., Ye, J.C.: Improving 3d imaging with pre-trained perpendicular 2d diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10710–10720 (2023)
19. Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., Aila, T.: Noise2noise: Learning image restoration without clean data. International Conference on Machine Learning (2018)
20. Lepcha, D.C., Goyal, B., Dogra, A., Vaghela, K., Singh, A., Kumar, K.R., Bavirisetti, D.P.: Low-dose computed tomography image denoising using pixel level non-local self-similarity prior with non-local means for healthcare informatics. Scientific Reports **15**(1), 25095 (2025)
21. Levandowsky, M., Winter, D.: Distance between sets. Nature **234**(5323), 34–35 (1971)
22. Li, X., Lu, J., Ding, H., Sun, C., Zhou, J.T., Chee, Y.M.: Pointcvar: Risk-optimized outlier removal for robust 3d point cloud classification. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 21340–21348 (2024)
23. Pierazzo, N., Morel, J.M., Facciolo, G.: Multi-scale dct denoising. Image Processing On Line **7**, 288–308 (2017)
24. Rakotosaona, M.J., La Barbera, V., Guerrero, P., Mitra, N.J., Ovsjanikov, M.: Pointcleannet: Learning to denoise and remove outliers from dense point clouds. In: Computer graphics forum. vol. 39, pp. 185–203. Wiley Online Library (2020)
25. Saragadam, V., LeJeune, D., Tan, J., Balakrishnan, G., Veeraraghavan, A., Baraniuk, R.G.: Wire: Wavelet implicit neural representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18507–18516 (2023)
26. Sitzmann, V., Martel, J.N., Bergman, A.W., Lindell, D.B., Wetzstein, G.: Implicit neural representations with periodic activation functions. In: Proc. NeurIPS (2020)
27. Sivgin, I., Fridovich-Keil, S., Wetzstein, G., Pilanci, M.: Geometric algebra planes: Convex implicit neural volumes. In: International Conference on Machine Learning (2025)
28. Smith-Bindman, R., Chu, P.W., Firdaus, H.A., Stewart, C., Malekhedayat, M., Alber, S., Bolch, W.E., Mahendra, M., de González, A.B., Miglioretti, D.L.: Projected lifetime cancer risks from current computed tomography imaging. JAMA internal medicine **185**(6), 710–719 (2025)
29. Song, B., Duan, Z., Gao, Y., Shao, T.: Adaptive bm3d algorithm for image denoising using coefficient of variation. In: 2019 22th International Conference on Information Fusion (FUSION). pp. 1–8. IEEE (2019)
30. Song, B., Hu, J., Luo, Z., Fessler, J., Shen, L.: Diffusionblend: Learning 3D image prior through position-aware diffusion score blending for 3D computed tomography reconstruction. Advances in Neural Information Processing Systems **37**, 89584–89611 (2024)
31. Sun, Y., Schneider, L.S., Gu, M., Mei, S., Ye, C., Wagner, F., Bayer, S., Maier, A.: Filter2noise: Interpretable self-supervised single-image denoising for low-dose ct with attention-guided bilateral filtering. arXiv preprint arXiv:2504.13519 (2025)
32. Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. Advances in neural information processing systems **33**, 7537–7547 (2020)
33. Turk, M., Baumeister, W.: The promise and the challenges of cryo-electron tomography. FEBS letters **594**(20), 3243–3261 (2020)
34. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9446–9454 (2018)

35. Verbin, D., Zickler, T.: Field of junctions: Extracting boundary structure at low snr. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6869–6878 (2021)
36. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004). https://doi.org/10.1109/TIP.2003.819861
37. Xu, X., Chi, Y.: Provably robust score-based diffusion posterior sampling for plug-and-play image reconstruction. Advances in Neural Information Processing Systems **37**, 36148–36184 (2024)
38. Yang, G.J.: Median filters and their applications to image processing. Purdue University (1980)
39. Yang, Z., Zang, D., Li, H., Zhang, Z., Zhang, F., Han, R.: Self-supervised noise modeling and sparsity guided electron tomography volumetric image denoising. Ultramicroscopy **255**, 113860 (2024)
40. Yang, Z., Zhang, F., Han, R.: Self-supervised cryo-electron tomography volumetric image restoration from single noisy volume with sparsity constraint. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4056–4065 (2021)
41. Yu, G., Sapiro, G.: Dct image denoising: a simple and effective image denoising algorithm. Image Processing On Line **1**, 292–296 (2011)
42. Yu, R., Li, X., Bi, T.: Modelling and validation of lidar noise distribution in fog and rain. Measurement **229**, 114472 (2024)
43. Zha, R., Lin, T.J., Cai, Y., Cao, J., Zhang, Y., Li, H.: R^2-gaussian: Rectifying radiative gaussian splatting for tomographic reconstruction. In: Advances in Neural Information Processing Systems (NeurIPS) (2024)
44. Zha, R., Zhang, Y., Li, H.: Naf: Neural attenuation fields for sparse-view cbct reconstruction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 442–452. Springer (2022)
45. Zhu, W., Mousavi, S.M., Beroza, G.C.: Seismic signal denoising and decomposition using deep neural networks. IEEE Transactions on Geoscience and Remote Sensing **57**(11), 9476–9488 (2019)
46. Zhu, Y., Huang, C.: An improved median filtering algorithm for image noise reduction. Physics Procedia **25**, 609–616 (2012)
47. Zhu, Y., Zhang, K., Liang, J., Cao, J., Wen, B., Timofte, R., Gool, L.V.: Denoising diffusion models for plug-and-play image restoration. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (NTIRE) (2023)