D2SA: Dual-Stage Distribution and Slice Adaptation for Efficient Test-Time Adaptation in MRI Reconstruction

Lipei Zhang[1]*, **Rui Sun**[2,3]*, **Zhongying Deng**[1], **Yanqi Cheng**[1], **Carola-Bibiane Schönlieb**[1], **Angelica I Aviles-Rivero**[4]

[1] Department of Applied Mathematics and Theoretical Physics, University of Cambridge
[2] Shenzhen Future Network of Intelligence Institute
and Guangdong Provincial Key Laboratory of Future Networks of Intelligence,
The Chinese University of Hong Kong (Shenzhen)
[3] School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen)
[4] Yau Mathematical Sciences Center, Tsinghua University

Abstract

Variations in Magnetic resonance imaging (MRI) scanners and acquisition protocols cause distribution shifts that degrade reconstruction performance on unseen data. Test-time adaptation (TTA) offers a promising solution to address this discrepancies. However, previous single-shot TTA approaches are inefficient due to repeated training and suboptimal distributional models. Self-supervised learning methods may risk over-smoothing in scarce data scenarios. To address these challenges, we propose a novel Dual-Stage Distribution and Slice Adaptation (D2SA) via MRI implicit neural representation (MR-INR) to improve MRI reconstruction performance and efficiency, which features two stages. In the first stage, an MR-INR branch performs patient-wise distribution adaptation by learning shared representations across slices and modelling patient-specific shifts with mean and variance adjustments. In the second stage, single-slice adaptation refines the output from frozen convolutional layers with a learnable anisotropic diffusion module, preventing over-smoothing and reducing computation. Experiments across five MRI distribution shifts demonstrate that our method can integrate well with various self-supervised learning (SSL) framework, improving performance and accelerating convergence under diverse conditions.

1 Introduction

Magnetic resonance imaging (MRI) captures detailed tissue structures using k-space sampling. In clinical practice, MRI is often under-sampled to accelerate scan time and reduce patient burden. However, under-sampling results in an ill-posed inverse problem, making accurate MRI reconstruction challenging [23]. Traditional compressed sensing techniques attempt to address this through iterative reconstruction algorithms [4, 5, 3, 28], but these methods are computationally expensive and less accurate. Recent advances in deep learning have significantly improved both reconstruction speed and quality by learning direct mappings from raw data [18], such as unrolled networks [27], plug-and-play frameworks [1], and diffusion models [8].

Despite these advancements, deep learning models struggle with adapting to diverse clinical scenarios due to two primary challenges. Firstly, limited MRI data for model adaptation: MRI datasets are difficult to collect, making it challenging to generalise deep models without overfitting. Secondly,

*These authors contributed equally.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

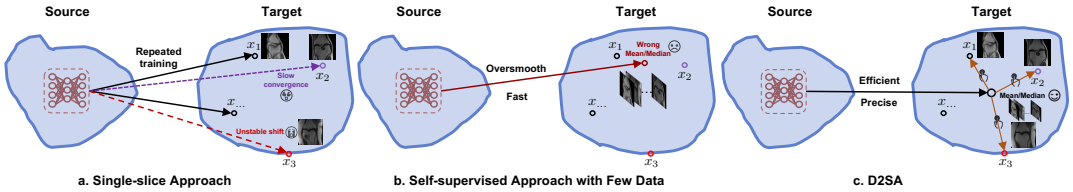

Figure 1: Illustration of TTA strategies for MRI reconstruction under distribution shifts. (a) Single-slice methods require repeated training and are often unstable.(b) Self-supervised approaches with limited data may oversmooth or converge to incorrect mean/median of target domain. (c) **D2SA** first performs efficient, patient-level adaptation to unknown target distributions (blue path), followed by optional slice-level refinement guided by requirement from clinician (orange path and 🧑‍⚕️).

distribution shifts between training and test data: In real-world deployment, MRI scans may be acquired under different conditions (e.g., scanner types, patient demographics), causing performance degradation due to mismatched data distributions between training and test sets [10, 11]. An ideal MRI reconstruction model should therefore balance three key goals for overcoming distribution shifts: 1) *Strong adaptation to new distributions* – maintaining high performance despite distribution shifts. 2) *Robustness to limited data* – preventing overfitting in data-scarce scenarios. 3) *Fast convergence* – minimising adaptation time at test time.

Most existing methods focus primarily on distribution generalisation but fail to optimise all three goals simultaneously. Test-time adaptation (TTA) techniques partially address this, i.e., they mitigate the distribution shift by updating models on the fly using only test data. Besides handling distribution shifts, batch-based TTA methods (e.g., Noiser2noise [12], FINE [38], SSDU [36]) further enforce self-supervised learning across multiple slices to facilitate fast convergence. However, this batch-wise approach may overfit shared features across slices while ignoring slice-specific variations, leading to over-smoothed reconstructions. Conversely, single-slice-based TTA methods [41, 34, 35] improve fine-grained adaptation but require repeated optimisation cycles, significantly increasing computational overhead. More recent diffusion-based models [2, 9] generate realistic slices for adaptation but are computationally expensive and prone to overfitting on smaller datasets.

To effectively balance all three goals, we propose Dual-Stage Distribution and Slice Adaptation (D2SA). D2SA leverages both patient-wise and slice-wise adaptation through a two-stage process. The first stage models single patient distribution using a small number of slices as prior knowledge. The second stage utilises this learned prior for fast adaptation to each slice, and further introduces an anisotropic diffusion (AD) module to enhance denoising [21, 7] while preventing over-smoothing the structural details. It thus achieves fast adaptation with high reconstruction quality. Both stages treat each MRI slice as a continuous function rather than a static matrix, drawing inspiration from Functa [13] and implicit neural representations (INRs)[25]. This function-based perspective allows us to interpret distribution shifts as small function-level variations, e.g., functions with different mean/variance variables in the feature space. Owing to the adaptive mean/variance, this function-centric approach can be efficiently adapted to new distributions without the need for extensive data for retraining. It also enables the plug-in of networks at test time, thus highly flexible. Our novel approach ensures fast convergence, robustness to limited data, and strong generalisation to new distributions, addressing a critical gap in MRI reconstruction research. Our contributions are:

- **Functional-Level Patient Adaptation.** We develop an INR-based strategy that learns a patient's distribution from a small number of slices, with the INRs trained to capture individualised mean and variance shifts for the second-stage fast adaptation.
- **Structural-Preserving Single-Slice Refinement.** After modelling patient-level shifts, the pre-trained INR network rapidly refines each slice. We introduce a learnable Anisotropic Diffusion (AD) module to maintain structural fidelity, reduce over-smoothing, and limit computation by freezing the main convolutional layers.
- **Extensive Validation.** We evaluate D2SA on five distribution shift scenarios, using both UNet [26] and a variational network [31]. Results demonstrate robust and efficient reconstruction across diverse clinical conditions.

2 Related Work

Test-Time Adaptation (TTA) in Medical Imaging. TTA tackles distribution shifts by adapting pre-trained models using unlabelled test data [22]. A key challenge is constructing supervision signals without ground truth, typically addressed via consistency regularisation or self-supervised losses. Consistency-based methods enforce stable predictions under perturbations. For instance, PINER [30] leverages implicit neural representations (INRs) to select resolution-consistent CT slices, while steerable diffusion models [2] ensure realistic reconstructions. Self-supervised approaches define pretext tasks such as contrastive learning [19] or rotation prediction [17]. DIP-TTT [11] applies self-supervision for slice-wise reconstruction under shifts, and Meta-TTT [34] incorporates meta-learning to improve generalisation. In contrast to computationally expensive TTA methods, we propose a dual-stage TTA framework that first performs patient-level adaptation to improve the efficiency and robustness of per-slice refinement.

Implicit Neural Representations (INRs). INRs encode data as continuous functions, enabling compact and flexible learning. Functa [14] embeds entire datasets as INRs for function-level learning, while DeepSDF [25] uses latent-conditioned autodecoders to model 3D shape fields. Biomedical INRs have been used to represent detailed structures like airway trees [40], allowing for effective batch optimisation. Among various designs [15], SIREN [29] remains a strong choice for high-frequency data due to its sine activation, and forms the basis of our patient-wise adaptation module.

3 Problem Setup

First, MRI reconstruction is an inverse problem where the goal is to recover $x^* \in \mathbb{C}^N$ from undersampled measurements $y \in \mathbb{C}^M$ with $M \ll N$: $y = A\,x^* + \epsilon$, where A is the measurement operator, and ϵ represents noise. In multi-coil MRI, the acquired measurements for each coil i follow:

$$y_i = M\,F\,S_i\,x^* + \epsilon, \quad i = 1, \ldots, n_c, \tag{1}$$

where S_i denotes the coil sensitivity map, F is the 2D Fourier transform. M is the undersampling mask which can be the 1D cartesian mask, or others [37]. The individual coil images $x_i = F^{-1}y_i$ are then combined via root-sum-of-squares to reconstruct x.

Reconstruction is framed as an optimisation problem:

$$\hat{x} = \arg\min_x \ \frac{1}{2}\|A\,x - y\|_2^2 + \lambda R(x), \tag{2}$$

where $R(x)$ encodes prior knowledge (e.g., wavelet ℓ_1, total variation, or CNN-based priors), and λ controls the balance between data fidelity and regularisation. However, standard reconstruction models assume a fixed distribution during testing, limiting their ability to generalise to new datasets or acquisition conditions.

Domain shifts from scanners, anatomy, or acquisition protocols degrade performance. Existing TTA methods can address this but they rely on repeated single-slice training [11] or self-supervised learning on large datasets [12, 38, 36], lacking stability in data-scarce scenarios. To address this, we introduce a D2SA that first learns patient-wise distributions explicitly for better initialisation, enabling more stable and efficient refinement in the second stage.

4 Proposed Method

To address domain shifts efficiently, we propose D2SA, a dual-stage test-time adaptation (TTA) framework that avoids large datasets and slow repeated single-slice training. As illustrated in Figure 2, D2SA consists of: (1) **Patient-wise Distribution Adaptation**, where MR-INR captures shared representations across slices and estimates variance (α) and mean (β) shifts; and (2) **Single-Slice Refinement (SST)**, which refines each slice using a learnable anisotropic diffusion (AD) module with frozen convolutional layers. This design enables efficient adaptation with improved initialization.

4.1 Functional-Level Patient Adaptation

In Figure 2, the MR-INR branch models the structure and distribution of patient-wise data. Inspired by Functa [13] and DeepSDF [25], which use INRs to encode data as continuous functions, our

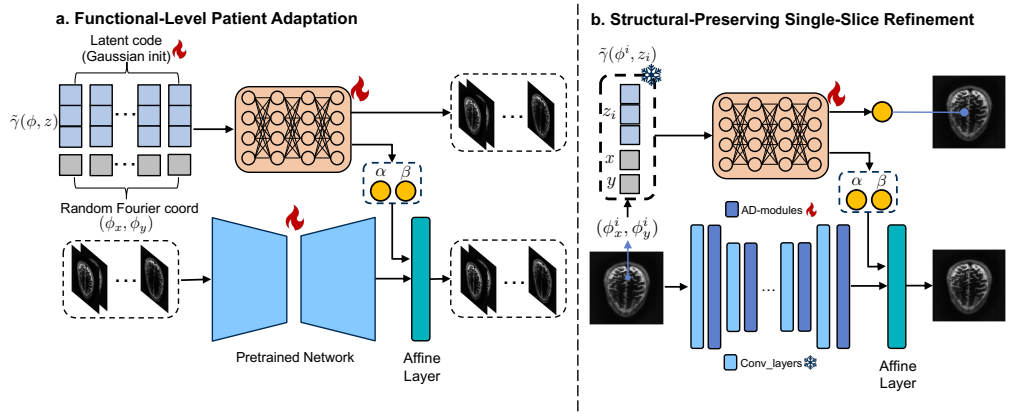

Figure 2: Overview of the proposed two-stage D2SA framework. (a) Functional-Level Patient Adaptation: An INR with a Gaussian-initialized latent code and random Fourier coordinates captures patient-level mean/variance shifts. The 🔥 indicates trainable modules, including the "pretrained" network and the affine layer. (b) Structural-Preserving Single-Slice Refinement (SST): The main convolutional layers and learned latent code are frozen ❄., while a learnable Anisotropic Diffusion (AD) module and the INR refine individual slices, preserving structural details and finalising outputs via the affine layer.

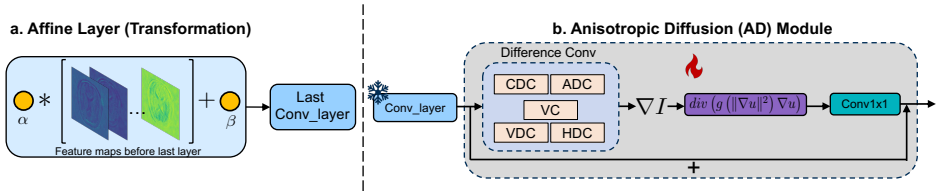

Figure 3: (a) Learnable affine transform scales feature maps by α and β before the final layer. (b) Learnable AD module 🔥 refines images while preserving structures, with frozen convolution ❄.

approach shifts from learning on discrete datasets to learning in function spaces. This enables efficient adaptation to new unknown distributions, better handling of few-shot scenarios, and improved patient-wise learning capabilities.

In Figure 2.a, each slice in this patient set can we assume the prior distribution over a 1D latent code z_i as zero-mean multivariate-Gaussian with a spherical covariance $\sigma^2 I$. In this work, σ is set to 0.01. The random Fourier coordinates will be calculated by geometric coordinates ϕ of each slice. The input $\tilde{\gamma}$ for MR-INR is from concatenation of z_i and ϕ.

$$\tilde{\gamma}(\phi, z_i) = [z_i, \cos(2\pi B\phi), \sin(2\pi B\phi)]\,, \tag{3}$$

where the transformation matrix B is sampled from a Gaussian distribution $\mathcal{N}(0, \omega^2)$.

After this step, we use the standard batch training protocol for all slices in each patient. In the MR-INR branch, the corresponding latent code and Fourier coordinates are modulated and passed through a SIREN [29] network f_θ architecture. The ability of SIREN and Fourier feature [32] to efficiently model target representation and stability has been shown in [15]. This MR-INR branch can be formed as:

$$\begin{aligned}
[\hat{x}, \alpha, \beta] &= f_\theta(\tilde{\gamma}(\phi^i, z)) = W_n(\Gamma_{n-1} \circ \Gamma_{n-2} \circ \cdots \circ \Gamma_0)(\tilde{\gamma}(\phi^i, z)) + b_n, \\
h^{(i+1)} &= \Gamma_i(h^{(i)}) = \sin(W_i h^{(i)} + b_i),
\end{aligned} \tag{4}$$

Here, $\Gamma_i : \mathbb{R}^{M_i} \to \mathbb{R}^{N_i}$ represents the i^{th} transformation layer. Each layer applies an affine transformation with weight matrix $W_i \in \mathbb{R}^{N_i \times M_i}$ and bias $b_i \in \mathbb{R}^{N_i}$, followed by a sine activation function. The final layer produces $[\hat{x}, \alpha, \beta]$ through three output heads. $\hat{x}$ with dimension $(B, 2, H, W)$, represents the predicted pixel intensity for MRI reconstruction. α variance shifts and β nonzero-mean shifts, with dimension (B, C, H, W), modulate feature maps before the final layer via an affine transformation, as shown in Figure 3.a, where C is the number of feature channels. This formulation

enables a shared base network to model common structures while adapting to patient-specific variations, ensuring a compact and efficient solution for TTA. Meanwhile, in the first stage, under-sampled MR images from the target domain are input into the network g_δ, initialised with source domain pre-trained weights, for TTA ($g_\delta \rightarrow g_{\delta+\Delta}$). Here, feature maps before last layer are extracted and adjusted via the affine transformation using α and β, as illustrated in Figure 3.a.

The predicted MR image from this branch is used to compute the self-supervised loss $\mathcal{L}_{\text{self}}$ such as Noiser2noise [12, 24], SSDU [36] and fidelity-based FINE [38]. The $\mathcal{L}_{\text{self}}$ combines with other two loss from MR-INR for joint optimisation. For MR-INR, we adopt a joint optimisation strategy similar to the auto-decoder framework [25], optimising both latent codes and network parameters. The optimisation for the first stage is formulated as:

$$\begin{aligned}\hat{\theta}, \hat{z}, \hat{\Delta} = \arg\min_{\theta,z,\Delta} \lambda_{\text{INR}} \sum_{(x_j,z_j,y_j)\in X} \mathcal{L}_1(Af(\tilde{\gamma}(\phi^j,z),\theta), y_j) \\ + \lambda_{\text{reg}} \frac{1}{\sigma^2}\|z\|_2^2 \quad + \lambda_{\text{self}} \sum_{(x_j)\in X} \mathcal{L}_{\text{self}}(g(x_j,\alpha,\beta,\delta+\Delta)).\end{aligned} \tag{5}$$

Here, $\mathcal{L}_1(Af(\tilde{\gamma}(\phi^j,z),\theta), y_j)$ ensures INR predictions aligning with MR signal consistency. The regularisation term $\frac{1}{\sigma^2}\|z\|_2^2$ constrains sparsity of the latent codes and prevents overfitting. The final term updates weights $\delta + \Delta$ to adapt to patient-specific variations using self-supervised loss. λ_{self}, λ_{INR} and λ_{reg} are used for balancing every loss contribution.

4.2 Structural-Preserving Single-Slice Refinement Training

To refine MRI reconstruction at the slice level, we design a new SST strategy for fine-grained adjustments. Unlike patient-wise adaptation, which learns shared representations across slices, this stage optimises each slice independently to capture localised variations, as shown in Figure 2.b. The latent codes are frozen to preserve the learned global prior information from patient-wise training. This prevents instability and avoids overfitting to slice-specific noise. Instead, the SIREN weights remain trainable, allowing the model to refine its implicit function for each slice. The affine modulation parameters α, β continue adjusting the final feature maps via scaling and shifting.

In the other branch, we adopt a similar DIP-based TTT strategy [11] for SST. This approach leverages CNNs' strong image priors for structural preservation and optimises a self-supervised loss $\mathcal{L}_{\text{self}}$ on under-sampled test measurements. To improve efficiency, we freeze all convolutional layers except the final one and transpose convolutions, reducing unnecessary updates and accelerating optimisation.

A key challenge in batch training is its tendency to learn mean or median representations, leading to over-smoothing that can obscure fine textures and edges. This is critical in MRI, where structural details must be preserved. To address this, we introduce an Anisotropic Diffusion (AD) module, inspired by its shape-preserving properties in image denoising [7, 6]. As shown in Figure 3.b, the AD module refines structural details while suppressing noise by integrating diffusion filtering into the adaptation process. Given an set of feature u, the AD equation is:

$$\left(\frac{\partial u}{\partial t}\right) = \text{div}\left(g(|\nabla u|)\nabla u\right); \quad g(|\nabla u|) = \frac{1}{1+\frac{|\nabla u|^2}{k^2}} \tag{6}$$

When the gradient magnitude is small ($|\nabla u| \rightarrow 0$), the diffusion coefficient g approaches 1, leading to isotropic smoothing similar to Gaussian filtering. Near object boundaries, where $|\nabla u| \rightarrow 1$, g approaches 0, preserving fine details. This allows AD to suppress noise effectively while keeping sharp edges, making it well-suited for edge-aware regularisation in reconstruction.

We enhance traditional convolutions by integrating difference-based operators [7] that explicitly encode gradient information ∇u. Five types of convolutions are introduced: Vanilla Convolution (VC), Central Difference Convolution (CDC), Angular Difference Convolution (ADC), Horizontal Difference Convolution (HDC), and Vertical Difference Convolution (VDC). These capture multiple directional gradients, incorporating concepts from Sobel, Prewitt, and Scharr filters directly into the convolution process [7]. The convolution operation is formulated as:

$$\nabla u = F_{\text{out}} = \text{DConv}(F_{\text{in}}) = \sum_{i=1}^{5} F_{\text{in}} * K_i = F_{\text{in}} * K_{\text{cvt}}, \tag{7}$$

where F_{in} and F_{out} represent input and output feature maps, respectively. Instead of separate convolutions, we merge all five kernels K_i into a single equivalent kernel K_{cvt} using a re-parameterisation technique. To improve efficiency, we reduce the number of output feature maps to 1/4 of the original channels, ensuring compact gradient extraction while minimising redundancy.

In calculation of AD equation (6), the computed ∇u is used to determine the diffusion coefficient g, while the divergence $\text{div}(\cdot)$ is approximated via a 2D Laplacian kernel, which is more efficient to preserve spatial information than standard finite difference methods [39]. The output of the first equation in (6) is restored to its original feature map dimensions using a 1×1 convolution. Setting the diffusion step size $\Delta t = 1$ in (6), the updated feature maps are:

$$u_{i+1} = u_i + \Delta t \cdot Conv_{1\times1}(\text{div}\,(g(|\nabla u_i|)\nabla u_i)). \tag{8}$$

In this stage, we optimise the weights in MR-INR and the original network with AD module. The final loss function for the second step is:

$$\hat{\theta}, \hat{\Delta} = \arg\min_{\theta,\Delta} \sum_{(x_j,y_j)\in X} \underbrace{\lambda_{\text{INR}}\mathcal{L}_1\big(Af(\tilde{\gamma}(\phi^j,\hat{z}),\theta), y_j\big)}_{\text{MR-INR consistency loss}} + \sum_{(y_j)\in X} \underbrace{\lambda_{\text{self}}\frac{|y_j - Ag(\mathbf{A}^\dagger \mathbf{y}_i,\delta+\Delta)|_1}{|y_j|_1}}_{\text{Self-sup loss}}. \tag{9}$$

The first term is for measurement consistency, ensures that the MR-INR branch reconstructs MRI images accurately. The second term, Self-Supervised loss, refines the prediction using measurement consistency. This formulation enables adaptive refinement while preserving prior knowledge learned from the first stage. λ_{self} and λ_{INR} are used for balancing every loss contribution.

4.3 Mathematical Analysis of Affine Adaptation

To motivate our use of affine transformations during test-time adaptation, we analyze a simplified setting under distribution shift. While our method uses nonlinear system, this linear case offers insights into how the learned parameters α and β operate under such shift. Consider the test distribution:

$$Q: \quad \mathbf{y} = \mathbf{x} + \mathbf{z}, \quad \mathbf{x} = \mathbf{U}\mathbf{c} + \mu_Q, \quad \mathbf{c} \sim \mathcal{N}(0, I), \quad \mathbf{z} \sim \mathcal{N}(0, s^2\mathbf{I}). \tag{10}$$

Here, $\mathbf{U} \in \mathbb{R}^{n\times d}$ is an orthonormal basis of the signal subspace, and μ_Q encodes the mean shift. Our goal is to estimate $\mathbf{x}$ under this shift. The optimal TTA estimator and self-supervised loss are next:

Proposition 1. *An affine estimator of the form $\hat{\mathbf{x}} = \alpha\mathbf{U}\mathbf{U}^T\mathbf{y} + \beta$ minimizes the following self-supervised loss:*

$$L_{SS}(\alpha,\beta) = \mathbb{E}_Q\left[\left\|\mathbf{y} - \alpha\mathbf{U}\mathbf{U}^T\mathbf{y} - \beta\right\|_2^2\right] + \frac{2\alpha d}{n-d}\mathbb{E}_Q\left[\left\|(\mathbf{I} - \mathbf{U}\mathbf{U}^T)\mathbf{y}\right\|_2^2\right]. \tag{11}$$

Theorem 1. *Minimizing $L_{SS}(\alpha,\beta)$ yields optimal parameters by solving the first-order conditions. The gradients are:*

$$\frac{\partial L_{SS}}{\partial \beta} = 2(\beta - \mu_Q), \quad \frac{\partial L_{SS}}{\partial \alpha} = -2d(1-\alpha) + 2\alpha d s^2.$$

Solving these gives the optimal solutions $\alpha^* = \frac{1}{1+s^2}$ and $\beta^* = \mu_Q$.

These parameters decouple the effects of noise and mean shift: α^* corrects variance, and β^* aligns the mean. They are learned by minimizing the self-supervised loss under the test distribution. This analysis provides a clear intuition for our design: although the full model is nonlinear, we apply the linear affine adaptation in feature space of MR-INR. More detailed proof and derivations are in the Appendix. Next, our empirical results further confirm the robustness of this effective TTA approach.

5 Experimental Settings

5.1 Datasets and Experimental Settings

We evaluate on multi-coil MRI data from **fastMRI** [37] (knee, brain) and **Stanford** [16]. Each experiment defines a source distribution $\mathcal{S}$ and target distribution $\mathcal{T}$, measuring performance via

SSIM, PSNR, and LPIPS. We consider two baselines: (1) **U-Net** [26]: 8 layers, 64 channels, trained with Adam [20] at learning rate 10^{-5}; (2) **VarNet** [31]: 12 cascades, 18 channels, trained with Adam at 10^{-4}. All other training settings follow [11], using combination of supervised and self-supervised losses. We simulate $4\times$ undersampling with 1D random Cartesian masks and 8% auto-calibration lines, estimating sensitivity maps via ESPiRiT [33]. We examine five domain shifts: *anatomy*, *dataset*, *modality*, *acceleration*, and *sampling*, evaluating both out-of-distribution ($\mathcal{S} \rightarrow \mathcal{T}$) and in-distribution performance (see Appendix).

Anatomy Shift. Following in [11], U-Net and VarNet are trained on fastMRI knee data as the source domain ($\mathcal{S}$) and evaluated on fastMRI AXT2 brain data as the target domain ($\mathcal{T}$). For TTA evaluation, we randomly select 10 subjects, resulting in 110 AXT2 brain slices subsampled at $4\times$.
Dataset Shift. Following [11], we train both models on Stanford knee data ($\mathcal{S}$) and evaluate on fastMRI knee data ($\mathcal{T}$). We sample 20 patients from fastMRI, yielding 400 knee slices under the same $4\times$ subsampling ratio for TTA evaluation.
Modality Shift. U-Net and VarNet are trained on fastMRI AXT2 brain slices ($\mathcal{S}$) and tested on AXT1PRE slices ($\mathcal{T}$), following the setup in [11]. We randomly select 10 patients, yielding 110 AXT1PRE brain slices with $4\times$ subsampling for TTA.
Acceleration Shift. Models are trained on fastMRI knee measurements acquired with $2\times$ acceleration ($\mathcal{S}$) and tested on the same set of knee slices with $4\times$ acceleration ($\mathcal{T}$), as in [11]. We evaluate TTA on 400 slices sampled from 20 patients.
Sampling Shift. The uniform sampling presents more coherent artifacts that are more challenging to handle. We also train on fastMRI AXT2 brain data subsampled using a random 1D Cartesian mask at $4\times$ acceleration ($\mathcal{S}$)[11], and evaluate on the same AXT2 slices sampled with a uniform 1D mask at the same acceleration rate ($\mathcal{T}$). For TTA, we randomly select 10 patients, totaling 110 slices.

Additional details of stage-1 and stage-2 training procedures are in the Appendix. **Appendix** is provided as a **separate** file in supplementary materials.

5.2 Compared Methods

We compare D2SA with four representative test-time adaptation (TTA) baselines. **DIP-TTT** [11] performs single-slice adaptation using Deep Image Prior (DIP). **FINE** [38] is a batch-level TTA approach based on fidelity constraints. **Noiser2noise (NR2N)** [12, 24] and **SSDU** [36] are self-supervised methods that operate in a patient-wise manner.

DIP-TTT follows its original setting, while FINE, NR2N, and SSDU are trained using the same configuration as the first stage of our method. To assess the benefit of MR-INR, we integrate it into FINE, NR2N, and SSDU for patient-wise adaptation. All resulting pretrained models—with and without MR-INR—are then used in the second-stage single-slice refinement under the same setup as our stage 2. Models without MR-INR adopt only self-supervised loss, similar to DIP-TTT.

We also include **ZS-SSL** [35], an augmentation-based self-supervised single-slice TTA method. Our preliminary results show that it is compatible only with unrolled networks such as VarNet, and fails to generalise to end-to-end U-Net architectures. Detailed comparisons are provided in the Appendix. All experiments were run on a single NVIDIA RTX 3090 GPU. For timing, Stage 1 inference time is measured as the total duration of 25 fixed training epochs. Stage 2 (SST) time is computed as the sum of per-slice training durations until early stopping, using the same validation-based strategy as in [11]. The final reported time combines both stages.

6 Results & Discussion

Main results. Tables 1 and Figure 4 summarize the average SSIM, PSNR, LPIPS, and adaptation time for U-Net and VarNet under five domain shifts: *Anatomy*, *Dataset*, *Modality*, *Acceleration*, and *Sampling*. Across most settings in Table 1, +MR-INR+SST achieves consistent improvements over baselines. For example, in the acceleration shift, FINE+MR-INR (VarNet) improves SSIM from 0.696 to 0.791 and PSNR from 21.39 to 25.30. While MR-INR introduces a modest runtime overhead. +MRI-INR+SST performances rival or exceed DIP-TTT in multiple cases (e.g., SSDU on anatomy shift, NR2N on dataset shift), while significantly reducing adaptation time (e.g., 17.1 vs. 52.5 mins/patient in anatomy shift). These results demonstrate the strong synergy between patient-wise MR-INR adaptation and single-slice SST refinement.

Method (VarNet)	Anatomy Shift ($\mathcal{S}$: Knee, $\mathcal{T}$: Brain)	Dataset Shift ($\mathcal{S}$: Stanford, $\mathcal{T}$: fastMRI)	Modality Shift ($\mathcal{S}$: AXT2, $\mathcal{T}$: AXT1PRE)	Acceleration Shift ($\mathcal{S}$: 2x, $\mathcal{T}$: 4x)	Sampling Shift ($\mathcal{S}$: Random, $\mathcal{T}$: Uniform)
Zero-filling	0.737/24.50/0.327/-	0.747/24.33/0.359/-	0.747/25.71/0.350/-	0.754/23.37/0.396/-	0.766/26.28/0.338/-
Non-TTA	0.799/23.16/0.371/-	0.706/22.35/0.365/-	0.796/23.54/0.379/-	0.761/23.04/0.372/-	0.111/16.00/0.594/-
DIP-TTT	0.878/27.67/0.312/52.5	0.798/28.02/0.292/41.8	0.867/28.33/0.337/71.6	0.815/28.25/0.285/137.2	0.771/27.65/0.254/38.9
FINE	0.820/24.01/0.343/3.9	0.789/26.26/0.311/6.6	0.821/26.18/0.369/3.5	0.696/21.39/0.342/6.2	0.669/21.18/0.365/3.7
FINE+MR-INR	0.862/26.45/0.328/4.7	0.795/26.44/0.306/6.9	0.830/26.58/0.369/4.4	0.791/25.30/0.310/7.5	0.789/25.10/0.339/4.1
FINE+SST	0.862/27.57/0.311/53.5	0.794/27.72/0.294/20.3	0.857/28.08/0.345/79.8	0.823/28.17/0.288/63.8	0.748/25.18/0.276/48.3
FINE+MR-INR+SST	0.882/27.68/0.311/17.1	0.808/28.72/0.286/18.2	0.867/28.32/0.337/21.8	0.829/28.64/0.276/44.5	0.824/28.49/0.232/22.1
NR2N	0.827/23.95/0.334/4.9	0.798/26.59/0.299/6.8	0.827/25.60/0.368/4.1	0.718/20.97/0.327/6.6	0.661/21.15/0.379/4.3
NR2N+MR-INR	0.868/26.41/0.321/5.1	0.806/26.95/0.294/7.1	0.833/26.44/0.369/4.7	0.806/25.42/0.291/7.6	0.786/25.24/0.366/4.5
NR2N+SST	0.883/27.72/0.307/63.9	0.798/27.78/0.293/25.3	0.860/28.17/0.341/80.2	0.822/28.09/0.291/69.2	0.771/26.46/0.276/57.1
NR2N+MR-INR+SST	0.884/27.81/0.306/18.7	0.812/28.76/0.281/20.4	0.869/28.36/0.336/20.3	0.826/28.89/0.273/43.1	0.822/28.40/0.241/25.2
SSDU	0.738/20.87/0.375/5.1	0.737/20.43/0.349/7.2	0.746/22.59/0.391/4.4	0.556/16.93/0.421/7.4	0.483/17.53/0.415/4.0
SSDU+MR-INR	0.821/23.25/0.350/5.3	0.764/21.67/0.339/7.5	0.796/24.09/0.390/4.8	0.728/19.57/0.358/7.9	0.554/18.88/0.409/5.3
SSDU+SST	0.879/27.65/0.310/68.3	0.789/26.79/0.299/24.9	0.857/28.07/0.343/93.4	0.803/28.06/0.293/134.2	0.736/24.84/0.288/50.5
SSDU+MR-INR+SST	0.882/27.66/0.307/18.5	0.808/28.16/0.290/21.4	0.863/28.09/0.342/29.3	0.826/28.62/0.286/45.2	0.800/28.21/0.254/25.8

Table 1: Performance comparison of VarNet methods under different domain shifts. Each cell presents ((**SSIM** ↑ / **PSNR** ↑ / **LPIPS** ↓ / **Time (mins/patient)** ↓). The family of proposed methods incorporates a self-supervised learning framework, combining MR-INR-based patient-wise adaptation with single-slice refinement using pre-trained patient-wise models.

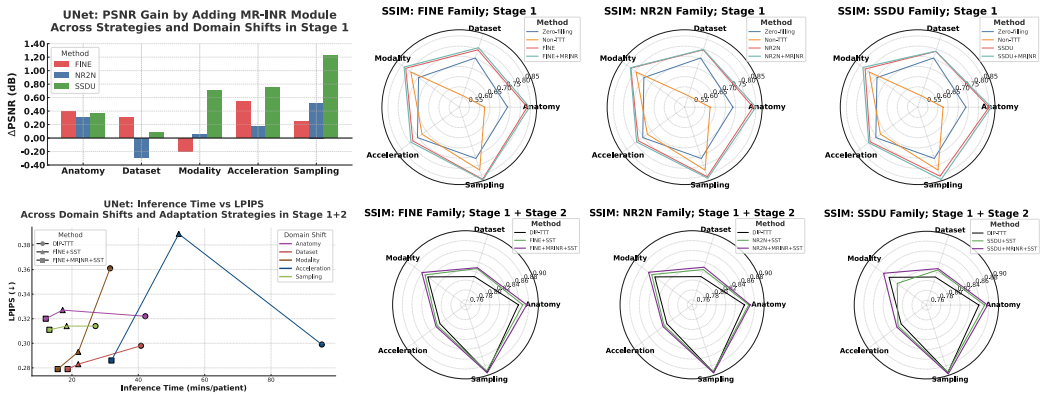

Figure 4: Performance Analysis of U-Net under Different Domain Shifts. Top-left: PSNR gain from adding MR-INR across FINE, NR2N, and SSDU in stage 1, with the largest gain in modality shift in most strategies. Bottom-left: LPIPS vs. inference time trade-off showing that +MR-INR+SST achieves higher quality with reduced time comsumption compared to DIP-TTT and normal +SST. Right: SSIM across five domain shifts for each SSL family, visualized patient-level (Stage 1) and after slice refinement (Stage 1 + Stage 2). MR-INR consistently improves performance across domains, and combining it with SST further enhances SSIM.

Figure 4 further illustrates the effectiveness of MR-INR. The top-left plot shows PSNR consistently increases across most TTA families, with the largest gain in the sampling shift. The bottom-left trade-off curve shows MR-INR+SST achieves lower LPIPS with lower cost than DIP-TTT. Radar plots confirm SSIM gains from Stage 1 (MR-INR) and further improvements when combined with Stage 2 (SST). Quantitative results of U-Net results and performance analysis of VarNet are also provided in the Appendix 8.

Additional findings on experiments show increased undersampling or new anatomies, modalities and datasets show that non-TTA often outperforms zero-filling, especially for U-Net. However, under large mask shifts, unrolled models like VarNet suffer more severe degradation without adaptation, underscoring the necessity of TTA in such cases.

Qualitative Results. Figure 5 present visual comparisons of reconstructed images for the FINE-based UNet methods in the anatomy shift. More results of other distribution shifts and VarnNet are provided in the Appendix. Self-supervised methods without MR-INR (e.g., FINE [38]) may risk over-smoothing when confronted with limited data, as highlighted in the error maps. While FINE+SST improves over FINE by incorporating single-slice adaptation, it lacks the AD module, leading to over-smoothing and loss of structural details. Our proposed approach, which integrates MR-INR with SST and AD, effectively balances adaptation and detail preservation, reducing hallucinations and enhancing reconstruction quality.

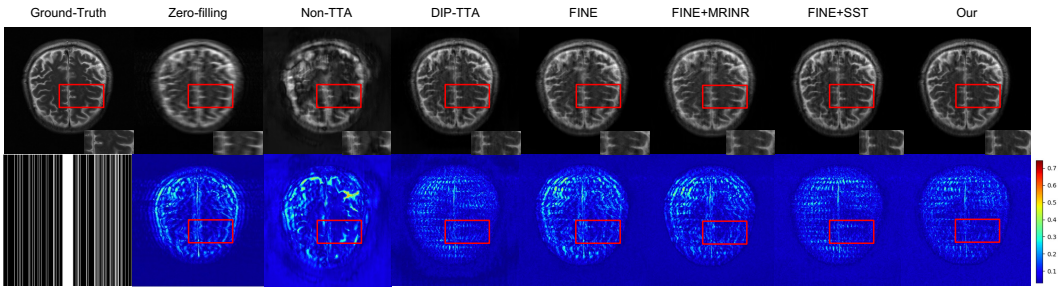

Figure 5: Comparison of different frameworks in UNet under anatomy shift (Knee to Brain) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI. The proposed method (far right) achieves the lowest residuals, indicating improved reconstruction accuracy.

Ablation Study. We conduct a comprehensive ablation to disentangle the contributions of MR-INR, SST, and AD to adaptation performance under anatomy shift (Table 2 and Table 3). For both U-Net and VarNet backbones, introducing MR-INR on top of FINE yields clear PSNR improvements (e.g., +0.5 dB on U-Net), demonstrating that the lightweight latent code provides effective patient-specific adaptation with negligible parameter increase. Further integrating SST significantly boosts performance but comes with increased inference time, highlighting its complementary role in capturing slice-level variations. Importantly, incorporating AD alongside frozen CNN achieves the best trade-off between performance and efficiency (27.71 dB / 12.1 min on U-Net; 27.68 dB / 17.1 min on VarNet), outperforming both purely patient-wise or slice-wise training. This suggests that AD effectively compensates for the lack of full fine-tuning, particularly in constrained adaptation settings, and enhances the robustness of MR-INR+SST pipelines. Interestingly, while AD brings limited improvement when used with a fully trainable CNN, it yields notable gains when the CNN are frozen. This highlights AD's effectiveness in constrained settings, where its adaptive capacity compensates for the lack of end-to-end fine-tuning. This effect is further supported by the results in the last two rows of the table.

We conduct a sensitivity analysis on the AD step size (Table 4). We observe that reconstruction quality is stable across a wide range of step sizes, with the best PSNR/SSIM obtained at 1.0 and only marginal degradation at smaller values. Meanwhile, LPIPS improves slightly as the step size decreases, indicating a tunable trade-off between fidelity and perceptual similarity. This robustness suggests that AD is insensitive to moderate hyperparameter variations, which is desirable for test-time deployment. Similarly, Figure 6 shows that adding directional and adaptive priors steadily improves PSNR and SSIM, further validating the effectiveness of the adaptive components.

Additional results on in-domain adaptation, statistical analysis, sensitivity analysis, and more visualizations are provided in Appendix 8.

7 Conclusion and Limitation

We presented D2SA framework, a test-time adaptation framework that improves MRI reconstruction under distribution shifts. D2SA combines patient-wise MR-INR for modeling mean/variance shifts and single-slice refinement via a learnable AD module. This dual-stage design enhances generalisation, preserves structural fidelity, and accelerates convergence. Extensive experiments across five domain shifts demonstrate that D2SA consistently outperforms prior TTA approaches in both quality and efficiency. Ablation studies further validate the contributions of MR-INR, AD, and frozen layers in balancing performance and runtime. While D2SA reduces adaptation time and improves generalisation, several limitations remain. First, current evaluations are limited to publicly available datasets; further validation on real-world clinical undersampled MRI is necessary. Second, the framework operates on 2D slices independently—extending it to exploit full 3D spatial correlations is a natural next step. Finally, integrating stronger priors (e.g., diffusion models) and developing online or incremental learning strategies could further enhance adaptability and prevent forgetting when adapting to continuous patient streams.

Ablation (UNet)	PSNR ↑ / Params ↓ / Time ↓
Patient-wise training	
FINE	25.98 / 31.02 / **4.9**
+MR-INR (❄ latent code)	26.37 / 31.29 / 5.3
+MR-INR (🔥 latent code)	**26.48** / **31.29** / 5.5
Single-slice training without stage 1	
+SST (❄ cnn + 🔥 AD)	27.31 / 17.62 / 18.7
Single-slice training with stage 1	
+SST (🔥 cnn)	27.22 / 31.02 / 17.2
+SST (❄ cnn + 🔥 AD)	27.37 / 17.62 / 22.6
+MR-INR+SST (🔥 cnn)	27.65 / 31.29 / 23.65
+MR-INR+SST (🔥 cnn+🔥 AD)	27.54 / 46.13 / 15.7
+MR-INR❄+SST(❄cnn+🔥AD)	27.41 / 17.62 / 14.5
+MR-INR+SST(❄ cnn)	27.38 / **3.05** / 17.5
+MR-INR+SST(❄cnn+🔥AD)(Ours)	**27.71** / 17.89 / **12.1**

Table 2: Ablation study on MR-INR and AD under anatomy shift (U-Net). Each row reports PSNR ↑, parameter count (Millions) ↓, and inference time (min/patient) ↓. The latent codes only have 1408 parameters in this shift. Stage 1 compares MR-INR variants; Stage 2 evaluates SST with and without frozen MR-INR and AD.

Ablation (VarNet)	PSNR ↑ / Params ↓ / Time ↓
Patient-wise training	
FINE	24.01 / 29.45 / **3.9**
+MR-INR (❄ latent code)	26.37 / 29.69 / 5.3
+MR-INR (🔥 latent code)	**26.45** / **29.69** / 4.7
Single-slice training without stage 1	
+SST (❄ cnn + 🔥 AD)	27.50 / 16.74 / 56.8
Single-slice training with stage 1	
+SST (🔥 cnn)	27.57 / 29.45 / 53.6
+SST (❄ cnn + 🔥 AD)	27.58 / 16.74 / 53.3
+MR-INR+SST (🔥 cnn)	27.68 / 29.69 / 20.9
+MR-INR+SST (🔥 cnn+🔥 AD)	27.61 / 43.79 / 34.0
+MR-INR❄+SST(❄cnn+🔥AD)	27.63 / 16.74 / 27.8
+MR-INR+SST(❄ cnn)	27.45 / **2.88** / 21.6
+MR-INR+SST(❄cnn+🔥AD)(Ours)	**27.68** / 16.98 / **17.1**

Table 3: Ablation study on MR-INR and AD under anatomy shift (VarNet). Each row reports PSNR ↑, parameter count (Millions) ↓, and inference time (min/patient) ↓. The latent codes only have 1408 parameters in this shift. Stage 1 compares MR-INR variants; Stage 2 evaluates SST with and without frozen MR-INR and AD.

AD Step Size	PSNR ↑	SSIM ↑	LPIPS ↓
1.0	27.71	0.876	0.320
0.1	27.38	0.874	0.322
0.01	27.36	0.874	0.323

Table 4: Sensitivity analysis of AD step size. Reconstruction quality remains stable across step sizes, with the best PSNR/SSIM at 1.0 and slightly improved LPIPS for smaller values, indicating a tunable fidelity–perception trade-off and robustness to hyperparameter variations.

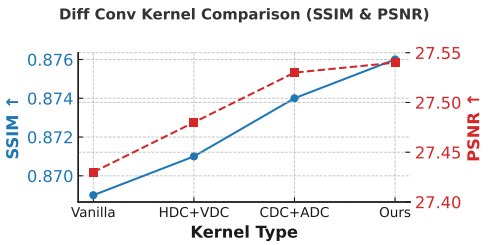

Figure 6: Ablation study on different kernel designs in difference convolution in terms of SSIM ↑ and PSNR ↑. Progressive improvements are observed from Vanilla to HD+VD, CD+AD, and ours, demonstrating the effectiveness of direction-aware and adaptive designs.

8 Acknowledgement

LZ gratefully acknowledges the financial aid award from NeurIPS and the travel support from DAMTP and Queens' College. RS acknowledges the support from Future Network of Intelligence Institute, The Chinese University of Hong Kong (Shenzhen). ZD acknowledges the support from Wellcome Trust 221633/Z/20/Z and the funding from the Cambridge Centre for Data-Driven Discovery and Accelerate Program for Scientific Discovery, made possible by a donation from Schmidt Sciences. YC is funded by an AstraZeneca studentship and a Google studentship. CBS acknowledges support from the Philip Leverhulme Prize, the Royal Society Wolfson Fellowship, the EPSRC advanced career fellowship EP/V029428/1, EPSRC grants EP/S026045/1 and EP/T003553/1, EP/N014588/1, EP/T017961/1, the Wellcome Innovator Awards 215733/Z/19/Z and 221633/Z/20/Z, the European Union Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant agreement No.777826 NoMADS, the Cantab Capital Institute for the Mathematics of Information and the Alan Turing Institute. AIAR gratefully acknowledges the support from Yau Mathematical Sciences Center, Tsinghua University.

References

[1] Rizwan Ahmad, Charles A Bouman, Gregery T Buzzard, Stanley Chan, Sizhuo Liu, Edward T Reehorst, and Philip Schniter. Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery. *IEEE signal processing magazine*, 37(1):105–116, 2020. 1

[2] Riccardo Barbano, Alexander Denker, Hyungjin Chung, Tae Hoon Roh, Simon Arridge, Peter Maass, Bangti Jin, and Jong Chul Ye. Steerable conditional diffusion for out-of-distribution adaptation in medical image reconstruction. *IEEE Transactions on Medical Imaging*, 2025. 2, 3

[3] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009. 1

[4] Kai Tobias Block, Martin Uecker, and Jens Frahm. Undersampled radial mri with multiple coils. iterative image reconstruction using a total variation constraint. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 57(6):1086–1098, 2007. 1

[5] Emmanuel J Candès and Michael B Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008. 1

[6] Tianxiang Chen, Zhentao Tan, Tao Gong, Qi Chu, Bin Liu, and Nenghai Yu. Feature preservation and shape cues assist infrared small target detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2024. 5

[7] Zixuan Chen, Zewei He, and Zhe-Ming Lu. Dea-net: Single image dehazing based on detail-enhanced convolution and content-guided attention. *IEEE Transactions on Image Processing*, 33:1002–1015, 2024. 2, 5

[8] Hyungjin Chung and Jong Chul Ye. Score-based diffusion models for accelerated mri. *Medical image analysis*, 80:102479, 2022. 1

[9] Hyungjin Chung and Jong Chul Ye. Deep diffusion image prior for efficient ood adaptation in 3d inverse problems. In *European Conference on Computer Vision*, pages 432–455. Springer, 2024. 2

[10] Mohammad Zalbagi Darestani, Akshay S Chaudhari, and Reinhard Heckel. Measuring robustness in deep learning based compressive sensing. In *International Conference on Machine Learning*, pages 2433–2444. PMLR, 2021. 2

[11] Mohammad Zalbagi Darestani, Jiayu Liu, and Reinhard Heckel. Test-time training can close the natural distribution shift performance gap in deep learning based compressed sensing. In *International Conference on Machine Learning*, pages 4754–4776. PMLR, 2022. 2, 3, 5, 7, 22, 24

[12] Arjun D Desai, Batu M Ozturkler, Christopher M Sandino, Robert Boutin, Marc Willis, Shreyas Vasanawala, Brian A Hargreaves, Christopher Ré, John M Pauly, and Akshay S Chaudhari. Noise2recon: Enabling snr-robust mri reconstruction with semi-supervised and self-supervised learning. *Magnetic Resonance in Medicine*, 90(5):2052–2070, 2023. 2, 3, 5, 7

[13] Emilien Dupont, Hyunjik Kim, S. M. Ali Eslami, Danilo Jimenez Rezende, and Dan Rosenbaum. From data to functa: Your data point is a function and you can treat it like one. In *39th International Conference on Machine Learning (ICML)*, 2022. 2, 3

[14] Emilien Dupont, Hyunjik Kim, SM Eslami, Danilo Rezende, and Dan Rosenbaum. From data to functa: Your data point is a function and you can treat it like one. *arXiv preprint arXiv:2201.12204*, 2022. 3

[15] Amer Essakine, Yanqi Cheng, Chun-Wun Cheng, Lipei Zhang, Zhongying Deng, Lei Zhu, Carola-Bibiane Schönlieb, and Angelica I Aviles-Rivero. Where do we stand with implicit neural representations? a technical and performance survey. *Transactions on Machine Learning Research*, 2025. Survey Certification. 3, 4

[16] CUBE FSE'XL, PD PD, and FAT FAT. Creation of fully sampled mr data repository for compressed sensing of the knee. In *SMRT 22nd Annual Meeting, Salt Lake City, Utah, USA*. Citeseer, 2013. 6, 16

[17] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018. 3

[18] Kerstin Hammernik, Teresa Klatzer, Erich Kobler, Michael P Recht, Daniel K Sodickson, Thomas Pock, and Florian Knoll. Learning a variational network for reconstruction of accelerated mri data. *Magnetic resonance in medicine*, 79(6):3055–3071, 2018. 1

[19] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 3

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[21] Karl Krissian and Santiago Aja-Fernández. Noise-driven anisotropic diffusion filtering of mri. *IEEE transactions on image processing*, 18(10):2265–2274, 2009. 2

[22] Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, 2025. 3

[23] Michael Lustig, David Donoho, and John M Pauly. Sparse mri: The application of compressed sensing for rapid mr imaging. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 58(6):1182–1195, 2007. 1

[24] Nick Moran, Dan Schmidt, Yu Zhong, and Patrick Coady. Noisier2Noise: Learning to Denoise From Unpaired Noisy Data . In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12061–12069, Los Alamitos, CA, USA, June 2020. IEEE Computer Society. 5, 7

[25] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 2, 3, 5

[26] Olaf Ronneberger, Philipp Fischer, and Thomas 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*, pages 234–241. Springer, 2015. 2, 7

[27] Jo Schlemper, Jose Caballero, Joseph V Hajnal, Anthony N Price, and Daniel Rueckert. A deep cascade of convolutional neural networks for dynamic mr image reconstruction. *IEEE transactions on Medical Imaging*, 37(2):491–503, 2017. 1

[28] Zakhar Shumaylov, Jeremy Budd, Subhadip Mukherjee, and Carola Bibiane Schonlieb. Weakly convex regularisers for inverse problems: Convergence of critical points and primal-dual optimisation. In *Forty-first International Conference on Machine Learning*, page 6574, 2024. 1

[29] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in neural information processing systems*, 33:7462–7473, 2020. 3, 4, 22

[30] Bowen Song, Liyue Shen, and Lei Xing. Piner: Prior-informed implicit neural representation learning for test-time adaptation in sparse-view ct reconstruction. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1928–1938, 2023. 3

[31] Anuroop Sriram, Jure Zbontar, Tullie Murrell, Aaron Defazio, C Lawrence Zitnick, Nafissa Yakubova, Florian Knoll, and Patricia Johnson. End-to-end variational networks for accelerated mri reconstruction. In *Medical image computing and computer assisted intervention–MICCAI 2020: 23rd international conference, Lima, Peru, October 4–8, 2020, proceedings, part II 23*, pages 64–73. Springer, 2020. 2, 7

[32] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020. 4

[33] Martin Uecker, Peng Lai, Mark J Murphy, Patrick Virtue, Michael Elad, John M Pauly, Shreyas S Vasanawala, and Michael Lustig. Espirit—an eigenvalue approach to autocalibrating parallel mri: where sense meets grappa. *Magnetic resonance in medicine*, 71(3):990–1001, 2014. 7

[34] Zhiwen Wang, Zexin Lu, Tao Wang, Ziyuan Yang, Hui Yu, Zhongxian Wang, Yinyu Chen, Jingfeng Lu, and Yi Zhang. Test-time adaptation via orthogonal meta-learning for medical imaging. *IEEE Transactions on Radiation and Plasma Medical Sciences*, 2024. 2, 3

[35] Burhaneddin Yaman, Seyed Amir Hossein Hosseini, and Mehmet Akcakaya. Zero-shot self-supervised learning for MRI reconstruction. In *International Conference on Learning Representations*, 2022. 2, 7, 24

[36] Burhaneddin Yaman, Seyed Amir Hossein Hosseini, Steen Moeller, Jutta Ellermann, Kâmil Uğurbil, and Mehmet Akçakaya. Self-supervised learning of physics-guided reconstruction neural networks without fully sampled reference data. *Magnetic resonance in medicine*, 84(6):3172–3191, 2020. 2, 3, 5, 7

[37] Jure Zbontar, Florian Knoll, Anuroop Sriram, Tullie Murrell, Zhengnan Huang, Matthew J Muckley, Aaron Defazio, Ruben Stern, Patricia Johnson, Mary Bruno, et al. fastmri: An open dataset and benchmarks for accelerated mri. *arXiv preprint arXiv:1811.08839*, 2018. 3, 6, 16

[38] Jinwei Zhang, Zhe Liu, Shun Zhang, Hang Zhang, Pascal Spincemaille, Thanh D Nguyen, Mert R Sabuncu, and Yi Wang. Fidelity imposed network edit (fine) for solving ill-posed image reconstruction. *Neuroimage*, 211:116579, 2020. 2, 3, 5, 7, 8

[39] Lipei Zhang, Yanqi Cheng, Lihao Liu, Carola-Bibiane Schönlieb, and Angelica I Aviles-Rivero. Biophysics informed pathological regularisation for brain tumour segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 3–13. Springer, 2024. 6

[40] Minghui Zhang, Hanxiao Zhang, Xin You, Guang-Zhong Yang, and Yun Gu. Implicit representation embraces challenging attributes of pulmonary airway tree structures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 546–556. Springer, 2024. 3

[41] Yutian Zhao, Tianjing Zhang, and Hui Ji. Test-time model adaptation for image reconstruction using self-supervised adaptive layers. In *European Conference on Computer Vision*, pages 111–128. Springer, 2024. 2

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstracts and introduction claims to demonstrate the capabilities of our new method with five simulated domain shifts. Theoretical analysis and experimental results for the main claim are described in Section 4 and Section 6

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations related to the absence of validation on real-world clinically undersampled MR data and the lack of 3D exploration are discussed in Section 7.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theoretical assumptions and detailed proofs are provided in Section 4 ("Mathematical Analysis of Affine Adaptation") and the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Implementation details for our method and baselines are provided in Section 5 and the Appendix. Demo code for a representative experiment is included as supplementary material. A public GitHub repository will be released upon publication.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper includes demo code in the supplementary material with clear instructions to reproduce a representative experiment. Full implementation and training scripts for all experiments will be made publicly available upon publication. All used datasets are publicly available on [37, 16].

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies all training and testing details, including data splits, optimizer types, learning rates, number of epochs, and other key hyperparameters. The hyperparameter choices and dataset-specific configurations is provided in Section 5 and further detailed in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports the standard error of the mean for all evaluated settings under domain shifts, as shown in the Appendix. This provides a rigorous measure of uncertainty across test samples and supports the statistical validity of the results.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The main paper specifies that all experiments were conducted on a single NVIDIA RTX 3090 GPU. Inference time per patient is reported for each method in the Table 1, Table 2, Table 3, Figure 4 and Figure 7, reflecting computational efficiency. Additional implementation and runtime details are provided in the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The work is done in appropriation with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses potential positive societal impacts, such as improving MRI accessibility through faster and more robust reconstruction under five distribution shifts. We can not see any negative societal impacts arising from the proposed approach.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We properly cite all the test-time adaptation baselines and datasets we used in our paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Implementation details of our method are provided in the main paper and Appendix. A demo script is included in the supplementary material, and full code with additional documentation will be released upon publication.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This work does not involve crowdsourcing or research with human subjects. All experiments are conducted on publicly available medical imaging datasets, which are fully de-identified and released under appropriate data use agreements.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This study does not involve human participants or personally identifiable information. All data used are from publicly available, fully anonymized medical imaging datasets, which do not require IRB approval.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: No large language models (LLMs) were used in the development of the core methods or experiments. LLMs were only used for writing support and did not influence the scientific content of the paper.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

D2SA: Dual-Stage Distribution and Slice Adaptation for Efficient Test-Time Adaptation in MRI Reconstruction- Appendix

Contents

A Training Details

This section outlines the key configurations, optimisation strategies, and architectural choices employed during training.

Stage 1: Functional-Level Patient Adaptation. We train the MR-INR-based model using a batch size of 2. Two Adam optimizers are used: one with a learning rate of 10^{-4} for the MR-INR weights and the base network, and the other with a learning rate of 10^{-3} for the learnable latent code. The training runs for 25 epochs, with convergence typically observed after 20 epochs. The 1D latent code (size 1×128) is initialized using a zero-mean multivariate Gaussian distribution with standard deviation $\sigma = 0.01$.

For the SIREN architecture [29], we use a 4-layer MLP with 256 hidden units per layer and follow the weight initialization method from the original paper. In U-Net, the loss coefficients in this stage are set to: $\lambda_{\text{self}} = 1$, $\lambda_{\text{INR}} = 1$, and $\lambda_{\text{reg}} = 1\text{e}^{-4}$. For VarNet, we use $\lambda_{\text{self}} = 1$, $\lambda_{\text{INR}} = 1\text{e}^{-3}$, and $\lambda_{\text{reg}} = 1\text{e}^{-4}$.

Stage 2: Single-Slice Refinement. In this stage, we refine each slice using a learnable Anisotropic Diffusion (AD) module while keeping the original convolutional layers frozen. We use the Adam optimizer with a learning rate of 10^{-4} and train for up to 1000 steps.

Following the self-validation strategy in [11], we reserve 5% of the k-space signals for validation. If the validation error does not decrease in fixed iterations, the refinement process is terminated early. For early stopping, we apply a sliding window of size 30 to monitor the moving average of the validation error for methods including FINE, NR2N, and SSDU (with and without MR-INR). DIP-TTT uses a sliding window size of 100, as defined in the original repository [11]. The loss coefficients for U-Net are $\lambda_{\text{self}} = 1$, $\lambda_{\text{INR}} = 1$, and for VarNet, they are $\lambda_{\text{self}} = 1$, $\lambda_{\text{INR}} = 1\text{e}^{-3}$.

Mask Setup. First, a general under-sampling mask is set by using 1D Cartesian masks with an acceleration rate of $\times 4$ and 8% auto-calibrating lines.

- For *Anatomy*, *Dataset*, and *Modality* shifts, the same $\times 4$ random 1D Cartesian mask is applied to both source and target domains. - For the *Acceleration* shift, the source model is trained on MR signals under-sampled at $\times 2$ using the same 8% auto-calibration strategy, while the target domain uses a $\times 4$ mask with the same random seed to simulate the shift. - For the *Sampling* shift, the source domain uses a random $\times 4$ mask, and the target domain uses a uniform $\times 4$ mask, both with 8% calibration lines. - Specifically, for in-distribution testing under sampling shift, we apply different random seeds to generate the masks while keeping the sampling strategy ($\times 4$, random) unchanged.

All mask generation and TTA implementation are provided in the demo script.

B Supplementary Quantitative Results in OOD Shift

Performance comparison of UNet methods under different domain shifts. Table 5 reports comprehensive and direct quantitative comparisons of UNet-based reconstruction methods under five distinct types of domain shift: anatomy, dataset, modality, acceleration, and sampling. Our proposed two-stage strategy combines patient-level adaptation (MR-INR) and slice-level refinement (SST with AD module). It consistently achieves top or near-top results across SSIM, PSNR, and LPIPS. Meanwhile, the inference time remains practical and competitive. Notably, the two-stage models yield especially strong improvements under more challenging shifts such as dataset and modality, where domain discrepancies are typically larger. These gains demonstrate the benefit of globally shared representations learned during patient-level adaptation, which are then effectively refined with localized slice-level refinemennt training. Furthermore, compared to strong baselines like DIP-TTT and one-stage TTA methods (e.g., +SST), our models exhibit improved stability (lower LPIPS) and higher sample-level fidelity (SSIM/PSNR), highlighting the robustness and generalization capacity of our hierarchical test-time learning approach under OOD scenarios.

Visual Analysis of VarNet Performance Across Domain Shifts. Figure 7 provides a detailed visualization of VarNet's performance under various domain shifts based on the Table of VarNet in main paper. The bar chart (top-left) highlights PSNR gains achieved by incorporating MR-INR in Stage 1 across FINE, NR2N, and SSDU training strategies. Acceleration and sampling shifts benefit the most, reflecting MR-INR's ability to capture cross-slice anatomical consistency in challenging

Method (UNet)	Anatomy Shift ($\mathcal{S}$: Knee, $\mathcal{T}$: Brain)	Dataset Shift ($\mathcal{S}$: Stanford, $\mathcal{T}$: fastMRI)	Modality Shift ($\mathcal{S}$: AXT2, $\mathcal{T}$: AXT1PRE)	Acceleration Shift ($\mathcal{S}$: 2x, $\mathcal{T}$: 4x)	Sampling Shift ($\mathcal{S}$: Random, $\mathcal{T}$: Uniform)
Zero-filling	0.737/24.50/0.327/-	0.754/24.33/0.359/-	0.747/25.7/0.350/-	0.754/23.371/0.396/-	0.765/26.28/0.338/-
Non-TTA	0.625/21.77/0.458/-	0.559/21.87/0.454/-	0.794/27.18/0.391/-	0.726/23.37/0.396/-	0.825/26.97/0.376/-
DIP-TTT	0.859/27.05/0.322/42.1	0.810/28.08/0.298/40.8	0.846/27.61/0.361/31.5	0.815/27.93/0.299/95.3	0.894/28.98/0.314/27.1
FINE	0.834/25.98/0.351/4.9	0.796/26.54/0.319/6.4	0.825/26.71/0.377/5.6	0.782/25.75/0.333/6.6	0.872/27.99/0.334/5.1
FINE+MR-INR	0.845/26.37/0.346/5.5	0.807/26.84/0.314/6.6	0.835/26.51/0.373/6.0	0.793/26.29/0.326/7.0	0.876/28.24/0.333/5.2
FINE+SST	0.868/27.22/0.327/17.2	0.827/28.16/0.283/21.9	0.853/27.72/0.293/21.9	0.822/28.07/0.389/52.2	0.891/29.02/0.314/18.4
FINE+MR-INR+SST	0.876/27.71/0.320/12.1	0.829/28.34/0.279/18.7	0.861/27.93/0.279/15.7	0.825/28.54/0.286/31.9	0.895/29.05/0.311/13.2
NR2N	0.836/25.80/0.353/5.2	0.796/26.71/0.316/6.9	0.826/26.59/0.383/6.7	0.781/26.20/0.335/6.9	0.859/26.91/0.347/5.6
NR2N+MR-INR	0.849/26.11/0.346/5.7	0.798/26.42/0.317/7.4	0.829/26.64/0.380/7.3	0.791/26.37/0.332/7.5	0.867/27.43/0.343/5.7
NR2N+SST	0.868/27.38/0.323/21.7	0.825/28.23/0.284/22.5	0.854/27.69/0.284/22.6	0.822/28.07/0.291/52.7	0.892/29.08/0.313/19.3
NR2N+MR-INR+SST	0.871/27.32/0.323/12.2	0.830/28.43/0.279/16.4	0.862/27.98/0.279/14.5	0.825/28.86/0.287/31.7	0.895/29.12/0.311/12.2
SSDU	0.851/24.82/0.353/5.4	0.788/22.37/0.344/7.8	0.819/24.27/0.339/7.1	0.789/23.03/0.346/7.3	0.856/24.88/0.349/6.5
SSDU+MR-INR	0.861/25.18/0.348/5.6	0.789/22.45/0.339/8.0	0.832/24.97/0.385/7.4	0.797/23.78/0.344/7.7	0.873/26.11/0.347/6.6
SSDU+SST	0.871/25.17/0.349/25.3	0.825/28.35/0.284/30.6	0.854/27.71/0.287/25.7	0.823/28.07/0.293/139.8	0.893/29.02/0.315/28.1
SSDU+MR-INR+SST	0.877/27.46/0.322/11.5	0.828/28.36/0.287/18.9	0.860/28.04/0.287/17.4	0.826/28.62/0.286/44.2	0.897/29.04/0.310/14.6

Table 5: Performance comparison of UNet methods under different domain shifts. Each cell presents (**SSIM ↑ / PSNR ↑ / LPIPS ↓ / Time (mins/patient)** ↓). The proposed methods combine MR-INR-based patient-wise adaptation and single-slice refinement.

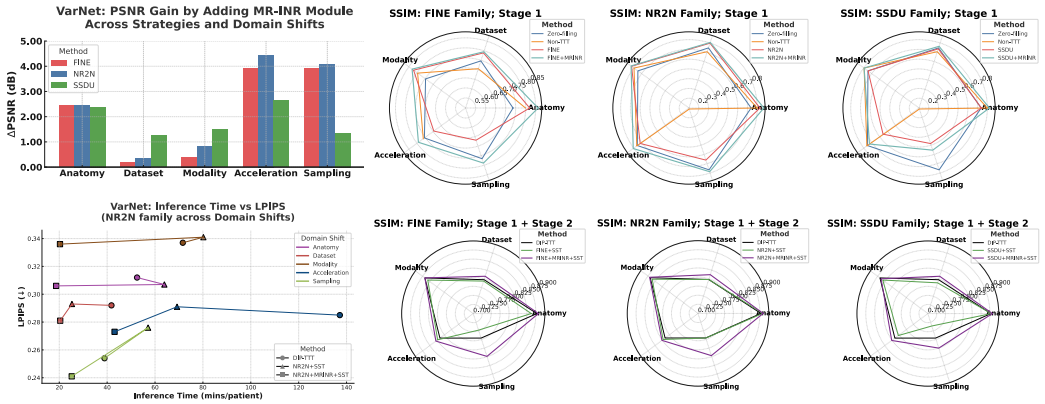

Figure 7: Performance Analysis of VarNet under Different Domain Shifts. Top-left: PSNR gain from adding MR-INR across FINE, NR2N, and SSDU in stage 1, with the largest gain in modality shift in most strategies. Bottom-left: LPIPS vs. inference time trade-off showing that +MR-INR+SST achieves higher quality with reduced time comsumption compared to DIP-TTT and normal +SST. Right: SSIM across five domain shifts for each SSL family, visualized patient-level (Stage 1) and after slice refinement (Stage 1 + Stage 2). MR-INR consistently improves performance across domains, and combining it with SST further enhances SSIM.

contexts. The LPIPS vs. inference time plot (bottom-left) illustrates that MR-INR+SST strikes a favorable balance between reconstruction quality and computational efficiency, outperforming DIP-TTT and conventional SST in both metrics. Radar plots (right) further validate our hierarchical design: Stage 1 improves SSIM through patient-level adaptation, while Stage 2 refinement with SST delivers additional performance boosts. These results reflect trends observed in Figure of performance analysis on UNet and confirm that our framework generalizes effectively to VarNet, enhancing robustness and generalization under all OOD shifts.

Distributional Analysis. To further assess the robustness of our method, Figure 8 presents violin plots of adjusted SSIM, PSNR, and LPIPS metrics under two representative domain shifts: *acceleration* (top row, Knee dataset with UNet) and *sampling* (bottom row, Brain dataset with VarNet). μ and σ denote the mean and standard deviation of each metric across slices per subject. Adjusted values are computed as $\mu - 0.5\sigma$ per subject, considering both and emphasizing performance stability across slices per patient. Notably, SSDU+MR-INR+SST consistently outperforms both DIP-TTT and SSDU+SST across all metrics. Improvements are statistically significant in most cases, particularly for PSNR and LPIPS ($p < 0.01$, Wilcoxon signed-rank test), highlighting our framework's ability to deliver high-fidelity and stable reconstructions. These results reaffirm that the two-stage strategy

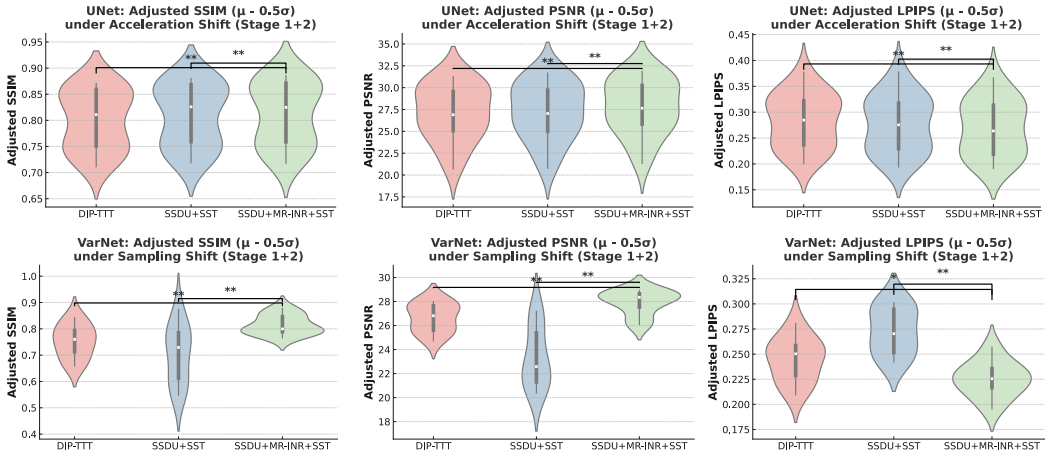

Figure 8: Comparison of model performance under domain shifts in **acceleration** (top) and **sampling** (bottom). Violin plots show the per-subject distributions of adjusted SSIM (↑), adjusted PSNR (↑), and adjusted LPIPS (↓), where each adjusted metric is computed as $\mu - 0.5\sigma$, with μ and σ denoting the mean and standard deviation of each metric across slices per subject. Three test-time strategies are evaluated: DIP-TTT, SSDU+SST, and SSDU+MR-INR+SST. The top row presents results under acceleration shift on the **Knee dataset** using UNet, while the bottom row corresponds to sampling shift on the **Brain dataset** using VarNet. Mean values are marked within each violin. Statistical significance of SSDU+MR-INR+SST compared to baseline methods (DIP-TTT and SSDU+SST) is indicated by * ($p < 0.05$) or ** ($p < 0.01$), based on Wilcoxon signed-rank tests.

(+MR-INR+SST with AD module) offers superior stability under diverse and challenging OOD shifts.

C Comparison to ZS-SSL

Recent progress in TTA MRI reconstruction has introduced zero-shot paradigms such as ZS-SSL [35], which train directly on undersampled measurements from a single subject. While effective in few data cases, ZS-SSL reconstructs each slice independently, partitioning and augmenting available k-space into training, loss, and validation sets. This design can limit the model's ability to capture shared anatomical representation.

Another representative method, DIP-TTT [11], improves upon the original Deep Image Prior by incorporating early stopping to stabilize optimization during test-time training. However, similar to ZS-SSL, it operates slice-by-slice and lacks modeling of inter-slice spatial dependencies. While DIP-TTT has been shown to outperform ZS-SSL under anatomy shifts, broader comparisons across other domain shifts are missing.

In contrast, our proposed approach adopts a two-stage test-time adaptation strategy: (1) MR-INR performs subject-level adaptation by leveraging all patient scans to capture shared anatomical structure and slice-wise relationships, and (2) SST refines each slice independently via AD module. This hierarchical modeling allows us to exploit both global context and local detail refinement during inference.

Table 6 presents quantitative results under five domain shifts: anatomy, dataset, modality, acceleration, and sampling. ZS-SSL shows competitive performance under anatomy and modality shifts, but it does not achieve surpassing on all metrics and spends more time to converge. It lags behind our methods in all other settings. Our MR-INR+SST framework consistently achieves state-of-the-art results across metrics and shifts. Furthermore, it significantly reduces inference time (e.g., 17–25 min vs. 100+ min for ZS-SSL), highlighting its practical applicability. More details about the the comparison on visualisation are in the last section-Supplementary Visualisations.

These findings suggest that the hierarchical design of our two-stage adaptation is better suited to handling diverse and challenging distribution shifts than methods relying solely on single-slice

reconstruction. By explicitly modeling cross-slice dependencies in Stage 1 and adapting locally in Stage 2, our method achieves strong generalization and efficiency in real-world deployment scenarios.

Method	**Anatomy Shift** ($\mathcal{S}$: Knee, $\mathcal{T}$: Brain)	**Dataset Shift** ($\mathcal{S}$: Stanford, $\mathcal{T}$: fastMRI)	**Modality Shift** ($\mathcal{S}$: AXT2, $\mathcal{T}$: AXT1PRE)	**Acceleration Shift** ($\mathcal{S}$: 2x, $\mathcal{T}$: 4x)	**Sampling Shift** ($\mathcal{S}$: Random, $\mathcal{T}$: Uniform)
Zero-filling	0.737/24.50/0.327/-	0.747/24.33/0.359/-	0.747/25.71/0.350/-	0.754/23.37/0.396/-	0.766/26.28/0.338/-
Non-TTT	0.799/23.16/0.371/-	0.706/22.35/0.365/-	0.796/23.54/0.379/-	0.761/23.04/0.372/-	0.111/16.00/0.594/-
DIP-TTT	0.878/27.67/0.312/52.5	0.798/28.02/0.292/41.8	0.867/28.33/0.337/71.6	0.815/28.25/0.285/137.2	0.771/27.65/0.254/38.9
ZS-SSL	0.884/27.07/0.314/99.5	0.745/21.93/0.365/135.4	0.860/28.31/0.338/103.3	0.751/21.37/0.381/167.4	0.734/24.30/0.306/52.2
FINE+SST	0.862/27.57/0.311/53.5	0.794/27.72/0.294/20.3	0.857/28.08/0.345/79.8	0.823/28.17/0.288/63.8	0.748/25.18/0.276/48.3
NR2N+SST	0.883/27.72/0.307/63.9	0.798/27.78/0.293/25.3	0.860/28.17/0.341/80.2	0.822/28.09/0.291/69.2	0.771/26.46/0.276/57.1
SSDU+SST	0.879/27.65/0.310/68.3	0.789/26.79/0.299/24.9	0.857/28.07/0.343/93.4	0.803/28.06/0.293/134.2	0.736/24.84/0.288/50.5
FINE+MRINR+SST	0.882/27.68/0.311/17.1	0.808/28.72/0.286/18.2	0.867/28.32/0.337/21.8	0.829/28.64/0.276/44.5	0.824/28.49/0.232/22.1
NR2N+MRINR+SST	0.884/27.81/0.306/18.7	0.812/28.76/0.281/20.4	0.869/28.36/0.336/20.3	0.826/28.89/0.273/43.1	0.822/28.40/0.241/25.2
SSDU+MRINR+SST	0.882/27.66/0.307/18.5	0.808/28.16/0.290/21.4	0.863/28.09/0.342/29.3	0.826/28.62/0.286/45.2	0.800/28.21/0.254/25.8

Table 6: Cross-domain evaluation under five domain shifts: anatomy, dataset, modality, acceleration, and sampling. Each cell shows the model performance in format **SSIM** (↑) **/ PSNR** (↑) **/ LPIPS** (↓) **/ Time (mins/patient)** (↓). Rows shaded in light red highlight our proposed MR-INR+SST methods, which achieve consistent improvements across shifts with notably reduced inference time.

D Other Ablation Studies and Sensitivity Analysis

Choice of INRs To clarify our choice of SIREN as the INR backbone in MR-INR, we conducted an additional ablation comparing different INR backbones under both patient-wise (FINE+MR-INR) and single-slice (FINE+MR-INR+SST) training. Results are summarized below:

Backbone (Stage 1)	**SSIM↑ / PSNR↑ / LPIPS↓**	**Backbone (Stage 1+2)**	**SSIM↑ / PSNR↑ / LPIPS↓**
WIRE	0.839 / 25.78 / 0.354	Finer	0.873 / 27.40 / 0.323
Finer	0.845 / 26.05 / 0.350	WIRE	0.872 / 27.44 / 0.321
RPE + MLP	0.845 / 25.76 / 0.349	RPE + MLP	0.864 / 27.22 / 0.326
SIREN only	0.845 / 26.16 / 0.349	SIREN only	0.874 / 27.51 / 0.322
Ours (PE+SIREN)	**0.845 / 26.37 / 0.346**	**Ours (PE+SIREN)**	**0.876 / 27.71 / 0.320**

Table 7: Sensitivity of INR backbone choice under patient-wise (Stage 1) and single-slice (Stage 1+2) Refinement.

Our design combining random Fourier positional encoding with SIREN consistently yields the best or near-best performance, especially in perceptual quality (LPIPS). While Finer offers frequency adaptability, its more complex architecture increases overhead without clear performance gains; WIRE, though efficient, suffers from unstable convergence and limited representation in medical shift settings. Compared to INCODE, our SIREN-based MR-INR avoids the need for pretraining, supporting our lightweight, plug-and-play adaptation objective.

Loss Weight Ablation We further evaluate the sensitivity of the framework to different loss weight configurations for Stage 1 (λ_{Self}, λ_{INR}, λ_{Reg}) and Stage 2 (λ_{Self}, λ_{INR}) (Table 8). Overall performance remains stable across a wide range of λ values, demonstrating the robustness of the training objectives. However, setting λ_{INR} too low (e.g., 0.1) leads to notable degradation, especially in Stage 1 and LPIPS in Stage 2. This aligns with the key role of MR-INR in capturing patient-level distribution shifts, where a sufficient INR loss weight is critical for effective adaptation.

Stage	**λ Settings**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**
Stage 1	1 / 1 / 1e-4	24.97	0.832	0.385
	1 / 0.1 / 1e-4	23.85	0.823	0.386
	1 / 1 / 0.1	24.00	0.822	0.388
Stage 2	1 / 1	28.04	0.860	0.287
	1 / 0.1	28.00	0.861	0.356
	1 / 0.01	27.99	0.861	0.350

Table 8: Ablation of loss weight combinations. Performance is stable across a wide range of λ values, but mild degradation when λ_{INR} is too small.

Hyper-parameter Sensitivity. We evaluate the sensitivity of MR-INR (Stage 1) and SST+AD (Stage 2) to key hyperparameters, including latent code size, INR depth, and initialization scale (Table 9). The performance across PSNR, SSIM, and LPIPS remains remarkably stable, with less than 0.15 dB variation in PSNR and negligible LPIPS differences. This robustness indicates that the method is largely insensitive to moderate hyperparameter changes, making it well suited for deployment without extensive tuning.

TTA Stage	Hyperparameter	SSIM ↑	PSNR ↑	LPIPS ↓
Stage 1	Latent Code Size (64 / 128 / 256)	0.849 / 0.845 / 0.850	26.39 / 26.37 / 26.47	0.345 / 0.346 / 0.335
	INR Depth (3 / 4 / 5)	0.842 / 0.845 / 0.848	26.34 / 26.37 / 26.44	0.345 / 0.346 / 0.346
	Init Std (0.1 / 0.01 / 0.001)	0.845 / 0.845 / 0.847	26.50 / 26.37 / 26.37	0.346 / 0.346 / 0.345
Stage 2	Latent Code Size (64 / 128 / 256)	0.876 / 0.876 / 0.877	27.53 / 27.71 / 27.53	0.323 / 0.320 / 0.320
	INR Depth (3 / 4 / 5)	0.874 / 0.876 / 0.877	27.54 / 27.71 / 27.56	0.321 / 0.320 / 0.319
	Init Std (0.1 / 0.01 / 0.001)	0.876 / 0.876 / 0.875	27.53 / 27.71 / 27.52	0.321 / 0.320 / 0.322

Table 9: Hyperparameter sensitivity of MR-INR (Stage 1) and SST+AD (Stage 2). The method remains stable under variations in latent code size, network depth, and initialization, with PSNR fluctuations < 0.15 dB and negligible LPIPS differences.

E Quantitative Results in Same Domain Shift

In-domain Generalization Analysis. Table 10 reports the performance of UNet-based reconstruction methods evaluated under same-domain (in-domain) settings across five shifts: *Brain*, *fastMRI*, *AXT1PRE*, *2x*, and *Random*. As expected, most methods perform better under in-domain testing compared to out-of-distribution (OOD) settings, with consistently higher SSIM and PSNR and lower LPIPS scores. Among the baselines, DIP-TTT demonstrates strong performance, especially in settings like Brain and AXT1PRE. However, it comes at a significantly higher inference cost, as reflected in its per-patient runtime.

Notably, our proposed two-stage pipeline (MR-INR+SST) achieves the best or second-best scores across almost all shifts, outperforming both FINE+SST and DIP-TTT in both fidelity and efficiency. For instance, in the sampling in same domain shift, SSDU+MR-INR+SST improves SSIM and PSNR while maintaining reduced LPIPS and halving the inference time compared to DIP-TTT. Furthermore, combining patient-level modeling (MR-INR) with slice-level SST refinement results in consistent performance gains over single-stage variants, demonstrating the benefit of hierarchical adaptation even in-domain.

This table highlights the robustness and generality of our framework: even when domain shifts are minimal in some cases, dual-stage adaptation continues to yield meaningful improvements in both reconstruction quality and computational efficiency.

Method (UNet)	Brain → Brain	fastMRI → fastMRI	AXT1PRE → AXT1PRE	2x → 2x	Random → Random
Zero-filling	0.737/24.50/0.359/-	0.754/24.33/0.359/-	0.747/25.70/0.350/-	0.846/26.52/0.226/-	0.811/26.32/0.387
Non-TTT	0.822/26.50/0.358/-	0.559/21.88/0.454/-	0.799/26.08/0.395/-	0.149/15.74/0.580/-	0.764/25.87/0.331
DIP-TTT	0.876/27.45/0.323/46.1	0.806/28.43/0.281/62.9	0.858/27.87/0.354/15.1	0.834/28.63/0.207/95.4	0.870/28.18/0.342/22.3
FINE	0.847/26.39/0.345/4.6	0.799/26.88/0.309/6.9	0.837/26.88/0.368/4.5	0.846/26.07/0.275/7.1	0.853/27.32/0.357/4.7
FINE+MRINR	0.852/26.66/0.343/13.5	0.805/27.08/0.312/7.2	0.839/27.28/0.366/4.7	0.878/28.40/0.229/7.4	0.856/27.44/0.356/4.9
FINE+SST	0.874/27.36/0.327/17.0	0.824/28.14/0.285/33.2	0.860/27.82/0.285/11.3	0.893/30.02/0.675/59.1	0.864/28.27/0.343/14.3
FINE+MRINR+SST	0.878/27.59/0.320/13.5	0.825/28.44/0.287/29.1	0.862/28.06/0.280/9.3	0.901/30.15/0.195/39.3	0.874/28.29/0.335/13.2
NR2N	0.850/26.11/0.347/4.8	0.799/26.92/0.307/7.3	0.835/26.88/0.374/4.7	0.836/25.09/0.286/7.2	0.850/27.40/0.362/4.8
NR2N+MRINR	0.857/26.18/0.345/5.0	0.803/26.99/0.298/7.6	0.836/26.95/0.378/5.1	0.870/25.56/0.242/7.6	0.854/27.35/0.358/5.0
NR2N+SST	0.875/27.49/0.323/20.7	0.825/28.20/0.283/33.5	0.865/27.88/0.283/12.6	0.892/30.03/0.207/60.0	0.864/28.27/0.343/18.1
NR2N+MRINR+SST	0.876/27.46/0.322/13.1	0.826/28.35/0.281/30.4	0.866/28.16/0.281/11.2	0.900/30.07/0.196/39.3	0.872/28.26/0.336/12.3
SSDU	0.861/25.16/0.347/5.0	0.794/22.64/0.332/7.5	0.848/25.40/0.375/5.2	0.804/20.73/0.311/7.4	0.836/24.30/0.374/5.0
SSDU+MRINR	0.865/25.36/0.323/5.3	0.8018/22.71/0.335/7.7	0.826/24.06/0.335/5.6	0.833/21.41/0.279/7.7	0.853/25.53/0.370/5.1
SSDU+SST	0.876/27.39/0.323/24.1	0.823/28.13/0.291/42.7	0.860/27.58/0.291/13.4	0.741/24.71/0.432/90.2	0.869/28.24/0.343/23.2
SSDU+MRINR+SST	0.879/27.57/0.322/11.5	0.825/28.17/0.285/35.4	0.863/28.14/0.285/11.2	0.901/30.25/0.192/44.2	0.877/28.30/0.333/14.4

Table 10: Performance comparison of **UNet** methods under in-domain evaluation. Each cell reports **SSIM** (↑) **/ PSNR** (↑) **/ LPIPS** (↓) **/ Time (mins/patient)** (↓). Shaded rows indicate methods that include the proposed MR-INR-based adaptation (MRINR) and further enhancement via slice-wise SST refinement (MRINR+SST).

In-domain Evaluation on VarNet. Table 11 presents a comprehensive comparison of VarNet-based reconstruction methods evaluated under in-domain conditions (including DIP-TTT and ZS-SSL). As expected, most methods demonstrate improved performance. Our proposed two-stage adaptation pipeline, incorporating MR-INR and SST, achieves strong results across most datasets in terms of SSIM, PSNR, and LPIPS.

Notably, our method (e.g., +MRINR+SST) attains the best SSIM, PSNR or LPIPS in four out of five shifts. An exception occurs in the AXT1PRE → AXT1PRE setting, where baseline+SST achieves slightly better metrics. In contrast, our dual-stage approach imposes a strong global anatomical prior, which promotes per-slice refinement in this homogeneous setting. A stronger global anatomical prior from MR-INR may slightly limit the flexibility of slice-level adaptation in this shift.

Despite this, our method still delivers highly competitive results with significantly less inference time compared to DIP-TTT. For instance, NR2N+MRINR+SST achieves a comparable PSNR of 28.27 in the T1 case within 18.4 minutes/patient, versus 25.8 minutes for DIP-TTT. This supports the claim that our framework achieves better trade-offs between quality and efficiency, making it favorable for real-world applications requiring fast implementation without sacrificing reconstruction accuracy.

Method (VarNet)	Brain → Brain	fastMRI → fastMRI	AXT1PRE → AXT1PRE	2x → 2x	Random → Random
Zero-filling	0.754/24.33/0.359/-	0.747/24.33/0.359/-	0.747/25.70/0.350/-	0.846/26.52/0.226/-	0.764/25.88/0.331/-
Non-TTT	0.845/24.60/0.305/-	0.706/23.12/0.331/-	0.838/22.48/0.354/-	0.149/15.74/0.580/-	0.111/15.98/0.593/-
DIP-TTT	0.875/27.29/0.315/102.4	0.815/28.28/0.282/57.1	0.869/28.33/0.331/25.8	0.840/29.26/0.196/120.4	0.683/24.84/0.320/32.3
ZS-SSL	0.885/27.57/0.313/124.4	0.754/22.22/0.379/100.2	0.870/27.93/0.334/85.4	0.742/21.94/0.302/167.4	0.634/21.34/0.410/49.6
FINE	0.854/26.49/0.328/3.9	0.801/26.55/0.300/7.2	0.862/27.70/0.335/3.5	0.816/24.17/0.273/7.4	0.646/20.62/0.388/4.3
FINE+MRINR	0.857/26.66/0.325/4.7	0.804/26.80/0.297/7.8	0.855/27.46/0.343/4.2	0.857/26.62/0.225/7.8	0.767/24.45/0.351/4.4
FINE+SST	0.877/27.62/0.310/91.9	0.809/27.87/0.289/30.4	0.873/28.41/0.329/26.2	0.832/29.07/0.204/63.8	0.697/24.06/0.327/47.1
FINE+MRINR+SST	0.884/27.81/0.306/47.6	0.825/28.34/0.278/23.5	0.862/28.16/0.337/19.1	0.860/29.30/0.198/44.3	0.787/27.49/0.261/18.3
NR2N	0.868/26.82/0.321/4.4	0.815/26.95/0.285/7.8	0.871/27.43/0.332/4.3	0.805/23.37/0.285/7.6	0.640/20.76/0.388/4.5
NR2N+MRINR	0.868/26.97/0.319/5.1	0.808/26.48/0.290/8.1	0.859/27.14/0.340/4.9	0.835/25.71/0.248/8.0	0.768/24.56/0.348/4.6
NR2N+SST	0.880/27.69/0.307/110.3	0.812/27.77/0.288/33.7	0.878/28.53/0.323/26.8	0.838/29.05/0.199/59.3	0.696/24.13/0.328/48.4
NR2N+MRINR+SST	0.885/27.81/0.306/49.7	0.826/28.42/0.279/22.9	0.867/28.27/0.331/18.4	0.844/29.24/0.191/45.7	0.787/27.51/0.261/20.4
SSDU	0.838/24.69/0.344/4.7	0.686/19.12/0.370/8.2	0.825/22.81/0.376/4.8	0.597/17.44/0.366/8.1	0.521/17.72/0.421/5.3
SSDU+MRINR	0.839/24.89/0.342/5.1	0.713/19.72/0.350/8.7	0.809/22.18/0.369/5.3	0.695/19.02/0.318/8.5	0.583/19.00/0.401/5.4
SSDU+SST	0.881/27.57/0.310/127.2	0.802/27.77/0.289/40.7	0.871/28.31/0.331/30.4	0.819/26.07/0.243/70.5	0.677/23.02/0.340/41.3
SSDU+MRINR+SST	0.887/27.57/0.310/50.4	0.815/28.29/0.285/39.2	0.867/28.20/0.333/24.7	0.826/28.62/0.286/50.1	0.768/27.05/0.282/25.9

Table 11: Performance comparison of **VarNet** methods under in-domain evaluation settings. Each cell reports **SSIM** (↑) **/ PSNR** (↑) **/ LPIPS** (↓) **/ Time (mins/patient)** (↓). Shaded rows indicate methods that include the proposed MR-INR-based adaptation (MRINR) and further enhancement via slice-wise SST refinement (MRINR+SST).

F Mathematical Analysis for Proposed Method

F.1 Test Distribution and Problem Formulation

We consider a setting where the observed signal $\mathbf{y}$ is a corrupted version of the underlying clean signal $\mathbf{x}$, which follows the test distribution:

$$Q : \mathbf{y} = \mathbf{x} + \mathbf{z}, \quad \mathbf{x} = \mathbf{U}\mathbf{c} + \mu_Q, \quad \mathbf{c} \sim \mathcal{N}(0, I), \quad \mathbf{z} \sim \mathcal{N}(0, Is^2). \tag{12}$$

Here, $\mathbf{U} \in \mathbb{R}^{n \times d}$ is an orthonormal basis for the signal subspace, and μ_Q represents the mean shift in the test distribution. Our goal is to estimate $\mathbf{x}$ under this distribution shift.

F.2 Self-Supervised Adaptation by Affine Transformation

To address the distribution shift from P to Q, we introduce an adaptation mechanism that accounts for both variance and mean shifts. The optimal estimator for $\mathbf{x}$ under test-time training (TTT) is:

$$\hat{\mathbf{x}} = \alpha \mathbf{U}\mathbf{U}^T\mathbf{y} + \beta, \tag{13}$$

where α accounts for variance shifts, and β corrects for mean shifts.

The self-supervised loss function is defined as:

$$L_{SS}(\alpha, \beta, \mathbf{U}, \mathbf{y}) = \mathbb{E}_Q \left[\|\mathbf{y} - \alpha \mathbf{U}\mathbf{U}^T\mathbf{y} - \beta\|_2^2 \right] + \frac{2\alpha d}{n-d} \mathbb{E}_Q \left[\|(\mathbf{I} - \mathbf{U}\mathbf{U}^T)\mathbf{y}\|_2^2 \right]. \tag{14}$$

Expanding the First Term of L_{SS}:

$$\begin{aligned} \mathbb{E}_Q \left[\|\mathbf{y} - \alpha \mathbf{U}\mathbf{U}^T\mathbf{y} - \beta\|_2^2 \right] = \mathbb{E}_Q \big[& \mathbf{y}^T\mathbf{y} - 2\alpha \mathbf{y}^T\mathbf{U}\mathbf{U}^T\mathbf{y} - 2\beta^T\mathbf{y} \\ & + \alpha^2 \mathbf{y}^T\mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{U}^T\mathbf{y} + 2\alpha\beta^T\mathbf{U}\mathbf{U}^T\mathbf{y} + \beta^T\beta \big]. \end{aligned} \tag{15}$$

Taking expectation:

$$\begin{aligned} & \mathbb{E}_Q \left[\mathbf{y}^T\mathbf{y} \right] - 2\alpha \mathbb{E}_Q \left[\mathbf{y}^T\mathbf{U}\mathbf{U}^T\mathbf{y} \right] - 2\mathbb{E}_Q \left[\beta^T\mathbf{y} \right] \\ & \quad + \alpha^2 \mathbb{E}_Q \left[\mathbf{y}^T\mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{U}^T\mathbf{y} \right] + 2\alpha \mathbb{E}_Q \left[\beta^T\mathbf{U}\mathbf{U}^T\mathbf{y} \right] + \mathbb{E}_Q \left[\beta^T\beta \right]. \end{aligned} \tag{16}$$

Compute Individual Expectations and using expectation properties:

$$\mathbb{E}_Q[\mathbf{y}^T\mathbf{y}] = \text{tr}(\mathbb{E}_Q[\mathbf{y}\mathbf{y}^T]). \tag{17}$$

Since:

$$\mathbb{E}_Q[\mathbf{y}\mathbf{y}^T] = \mathbf{U}\mathbf{U}^T + s^2 I + \mu_Q \mu_Q^T, \tag{18}$$

$$\begin{aligned} \mathbb{E}_Q[\mathbf{y}^T\mathbf{y}] &= \text{tr}(\mathbf{U}\mathbf{U}^T) + s^2\text{tr}(I) + \text{tr}(\mu_Q\mu_Q^T), & (19) \\ &= d + s^2 n + \|\mu_Q\|^2. & (20) \end{aligned}$$

For the second expectation:

$$\mathbb{E}_Q[\mathbf{y}^T\mathbf{U}\mathbf{U}^T\mathbf{y}] = \text{tr}(\mathbf{U}\mathbf{U}^T\mathbb{E}_Q[\mathbf{y}\mathbf{y}^T]). \tag{21}$$

Substituting $\mathbb{E}_Q[\mathbf{y}\mathbf{y}^T]$:

$$\begin{aligned} \mathbb{E}_Q[\mathbf{y}^T\mathbf{U}\mathbf{U}^T\mathbf{y}] &= \text{tr}(\mathbf{U}\mathbf{U}^T(\mathbf{U}\mathbf{U}^T + s^2 I + \mu_Q\mu_Q^T)), & (22) \\ &= \text{tr}(\mathbf{U}\mathbf{U}^T) + s^2\text{tr}(\mathbf{U}\mathbf{U}^T) + \text{tr}(\mathbf{U}\mathbf{U}^T\mu_Q\mu_Q^T), & (23) \\ &= d + s^2 d + \text{tr}(\mathbf{U}\mathbf{U}^T\mu_Q\mu_Q^T). & (24) \end{aligned}$$

$$L_{SS}(\alpha, \beta) = (d + s^2 n + \|\mu_Q\|^2) - 2\alpha(d + s^2 d + \text{tr}(\mathbf{U}\mathbf{U}^T\mu_Q\mu_Q^T)) - 2\beta^T\mu_Q + \alpha^2 d + 2\alpha d\beta^T\mu_Q + \|\beta\|^2. \tag{25}$$

Combine each component, we can get

$$L_{SS}(\alpha, \beta) = d + s^2 n + \|\mu_Q\|^2 - 2\alpha(d + s^2 d + \|\mathbf{U}^T\mu_Q\|^2) + \alpha^2 d + 2\alpha d\beta^T\mu_Q - 2\beta^T\mu_Q + \|\beta\|^2 \tag{26}$$

Final Simplified Expression for this term

$$L_{SS}(\alpha, \beta) = s^2 n + (1-\alpha)^2 d + (\alpha^2 - 2\alpha)s^2 d + \|\beta - \mu_Q\|^2 \tag{27}$$

Expending second term:

$$\frac{2\alpha d}{n-d} \mathbb{E}_Q \left[\|(\mathbf{I} - \mathbf{U}\mathbf{U}^T)\mathbf{y}\|_2^2 \right]. \tag{28}$$

First, expanding the squared norm:

$$\|(\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mathbf{y}\|_2^2 = \left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mathbf{y}\right)^T \left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mathbf{y}\right). \tag{29}$$

Since $(\mathbf{I}-\mathbf{U}\mathbf{U}^T)$ is symmetric:

$$= \mathbf{y}^T(\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mathbf{y}. \tag{30}$$

Taking expectation:

$$\mathbb{E}_Q\left[\mathbf{y}^T(\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mathbf{y}\right] = \text{tr}\left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mathbb{E}_Q[\mathbf{y}\mathbf{y}^T]\right). \tag{31}$$

Using the expectation property:

$$\mathbb{E}_Q[\mathbf{y}\mathbf{y}^T] = \mathbf{U}\mathbf{U}^T + s^2 I + \mu_Q\mu_Q^T. \tag{32}$$

Substituting:

$$= \text{tr}\left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)\left(\mathbf{U}\mathbf{U}^T + s^2 I + \mu_Q\mu_Q^T\right)\right). \tag{33}$$

Expanding the trace:

$$= \text{tr}\left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)s^2 I + (\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mu_Q\mu_Q^T\right). \tag{34}$$

Since $(\mathbf{I}-\mathbf{U}\mathbf{U}^T)$ removes the $\mathbf{U}\mathbf{U}^T$ component:

$$= s^2(n-d) + \text{tr}\left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mu_Q\mu_Q^T\right). \tag{35}$$

Thus, the second term simplifies to:

$$\frac{2\alpha d}{n-d}\left[s^2(n-d) + \text{tr}\left((\mathbf{I}-\mathbf{U}\mathbf{U}^T)\mu_Q\mu_Q^T\right)\right]. \tag{36}$$

Assuming μ_Q is entirely inside the subspace spanned by $\mathbf{U}$, the projection term vanishes, giving:

$$\frac{2\alpha d}{n-d}s^2(n-d). \tag{37}$$

Last, we take final simplification Now, simplifying the terms:

$$L_{SS}(\alpha,\beta) = s^2 n + (1-\alpha)^2 d + (\alpha^2 - 2\alpha)s^2 d + \|\beta-\mu_Q\|^2 + 2\alpha d s^2. \tag{38}$$

Combining the $s^2 d$ terms:

$$(\alpha^2-2\alpha)s^2 d + 2\alpha s^2 d = \alpha^2 s^2 d - 2\alpha s^2 d + 2\alpha s^2 d = \alpha^2 s^2 d. \tag{39}$$

Thus, the final loss function simplifies to:

$$L_{SS}(\alpha,\beta) = s^2 n + (1-\alpha)^2 d + \alpha^2 s^2 d + \|\beta-\mu_Q\|^2. \tag{40}$$

Finally, we compute the derivatives

For derivative with Respect to α

$$\frac{\partial L_{SS}}{\partial \alpha} = -2d(1-\alpha) + 2\alpha d s^2. \tag{41}$$

Setting this to zero and solving for α^*:

$$\alpha^* = \frac{1}{1+s^2}. \tag{42}$$

For derivative with respect to β

$$\frac{\partial L_{SS}}{\partial \beta} = 2(\beta - \mu_Q). \tag{43}$$

Setting this to zero and solving for β^*:

$$\beta^* = \mu_Q. \tag{44}$$

In conclusion,

1. α^* dynamically adjusts for noise variance shifts.
2. β^* corrects for mean shifts, making adaptation robust in OOD settings.

G Supplementary Visualisations

While the main paper presents qualitative comparisons on UNet (FINE) under the anatomy shift, this appendix includes additional visualisations across the remaining domain shifts. We provide side-by-side comparisons of reconstruction results for our method and competing approaches, including DIP-TTT and ZS-SSL. Notably, ZS-SSL results are visualised under the VarNet backbone as originally proposed with data consistency block. Our visualisations offer a more comprehensive view of cross-domain performance across architectures and adaptation strategies.

G.1 UNet

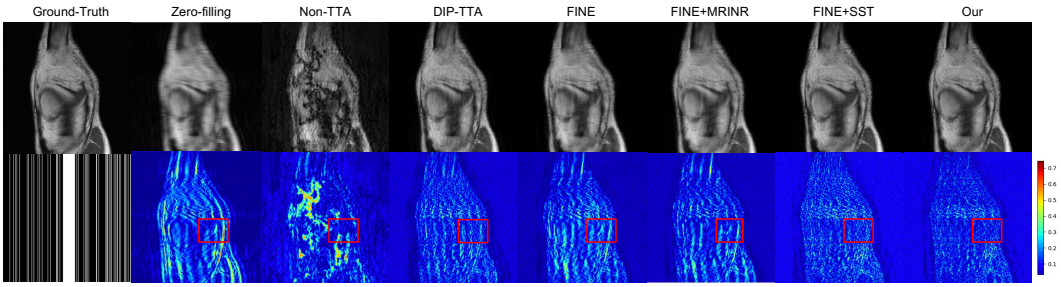

Figure 9: Comparison of different frameworks in UNet under dataset shift (Stanford to fastMRI) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

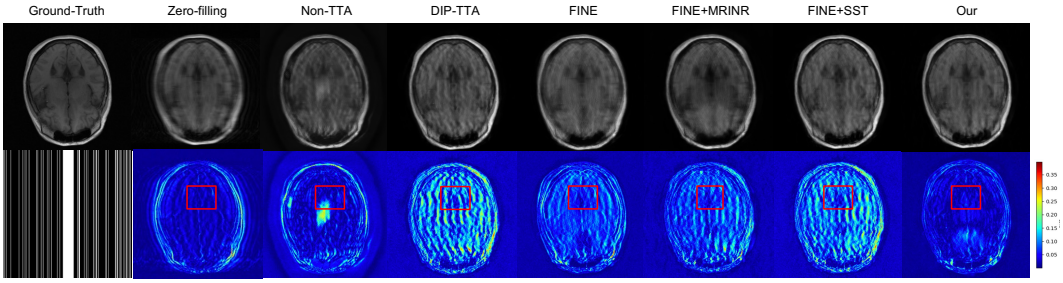

Figure 10: Comparison of different frameworks in UNet under modality shift (AXT2 to AXT1PRE) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

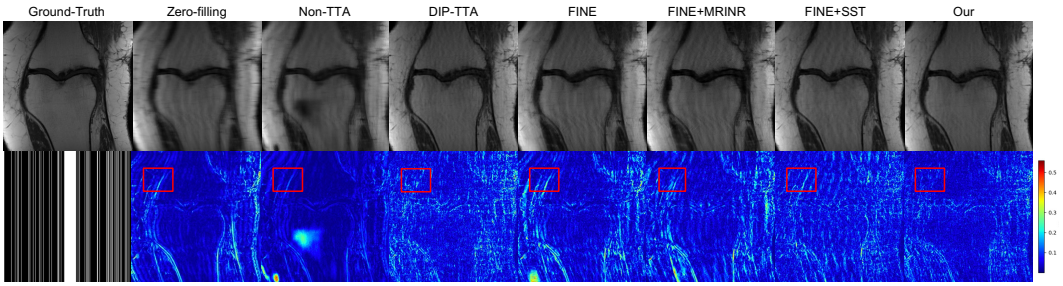

Figure 11: Comparison of different frameworks in UNet under acceleration shift (2X to 4X) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

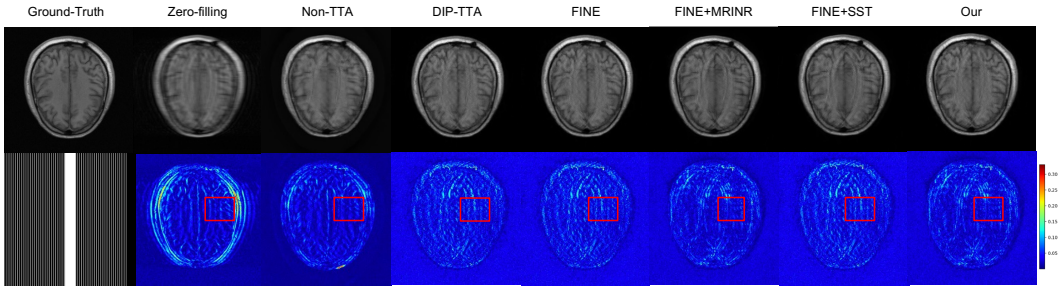

Figure 12: Comparison of different frameworks in UNet under sampling shift (random to uniform) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

G.2 VarNet

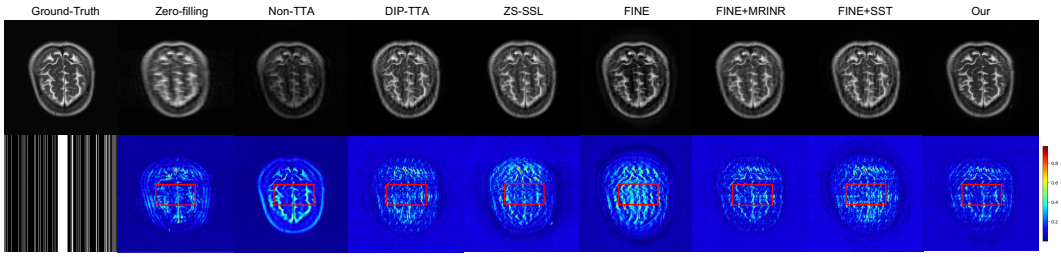

Figure 13: Comparison of different frameworks in VarNet under anatomy shift (Knee to Brain) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

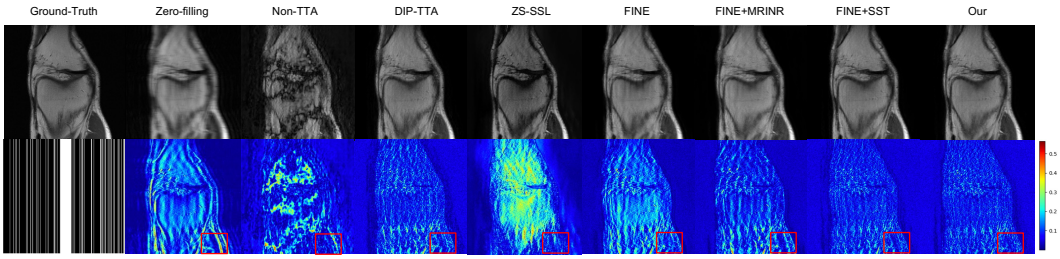

Figure 14: Comparison of different frameworks in VarNet under dataset shift (Stanford to fastMRI) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI. The proposed method (far right) achieves the lowest residuals, indicating improved reconstruction accuracy

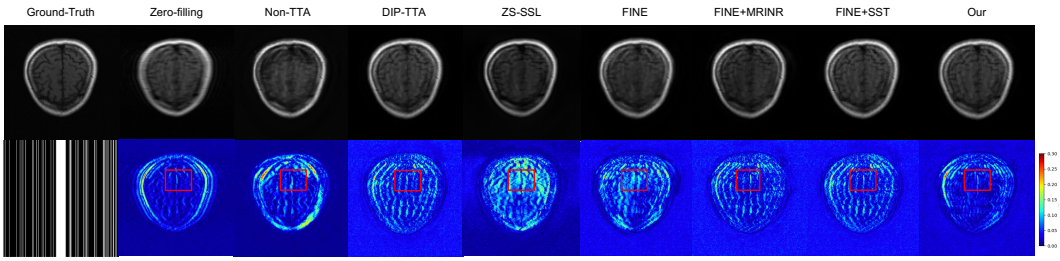

Figure 15: Comparison of different frameworks in VarNet under modality shift (AXT2 to AXT1PRE) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

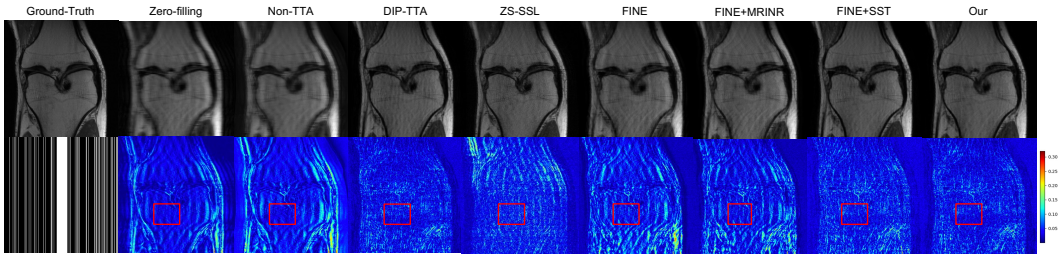

Figure 16: Comparison of different frameworks in Varnet under acceleration shift (2X to 4X) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.

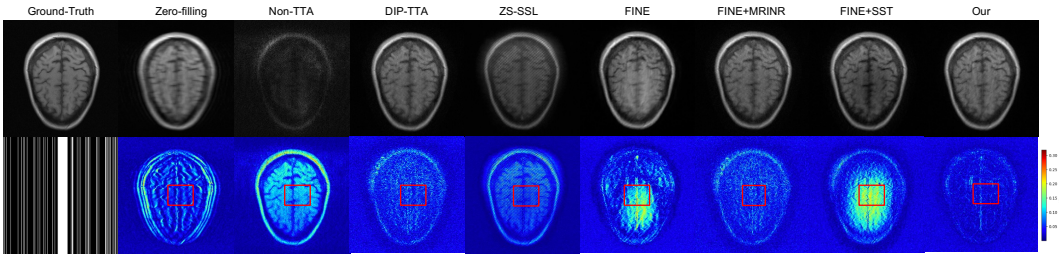

Figure 17: Comparison of different frameworks in VarNet under sampling shift (random to uniform) using the FINE method. The first row shows reconstructed MRI images, while the second row presents residual maps between reconstructions and full-sampled MRI.