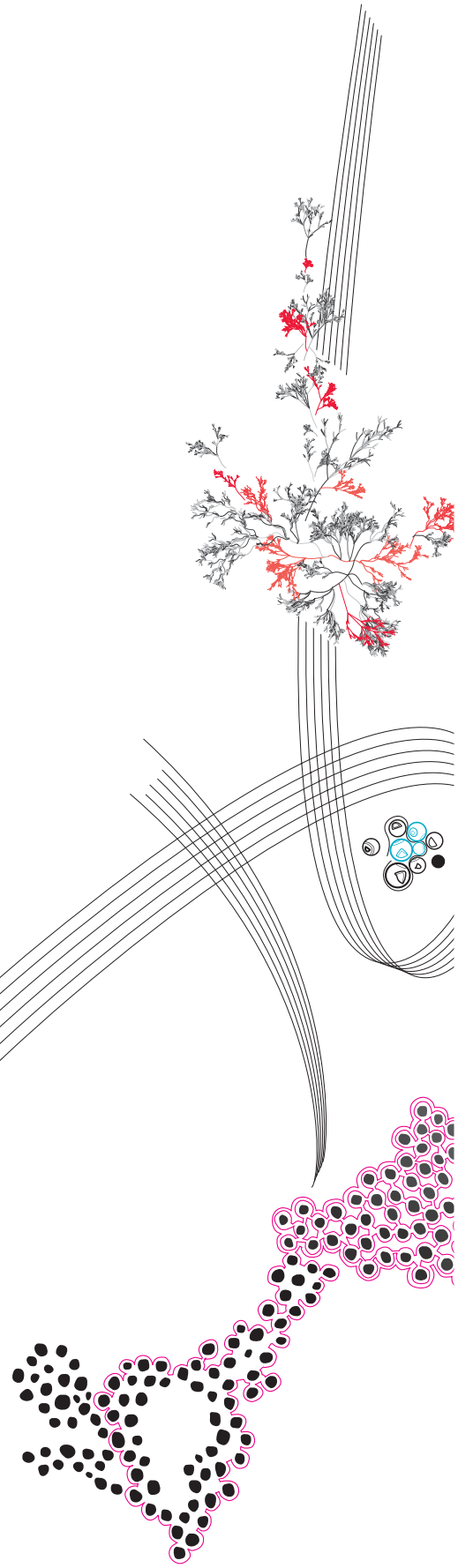


MSc Thesis Applied Mathematics

Optimal Transport Regularization of Implicit Neural Representations for Dynamic MRI Reconstruction

Thiadrik Robert George Blom

Graduation Committee:
Prof. dr. C. Brune
Dr. ir. J. Goseling
MSc. S.C. Dummer

June, 2025

Department of Applied Mathematics
Faculty of Electrical Engineering,
Mathematics and Computer Science

UNIVERSITY OF TWENTE.

Acknowledgments

I want to express my sincere gratitude to my supervisors, Prof. Dr. C. Brune and MSc. S.C. Dummer, for their continuous support, insightful guidance, and valuable feedback throughout this project. S.C. Dummer, thank you for always making the time to help me, for your clear explanations, and for pointing me toward new directions when I got stuck. I also wish to thank Prof. Dr. C. Brune for his time, expertise, and feedback during this thesis. Furthermore, I am grateful to the other member of my graduation committee, Dr. Jasper Goseling, for taking the time to read and evaluate my work. I would also like to thank the Mathematics Department of the University of Twente for their support and help during the project.

Finally, I am deeply grateful to my parents, family, and friends for their unwavering support, genuine interest, and for providing welcome distractions at times when I needed them most.

Abstract

The reconstruction of undersampled spatially and temporally undersampled dynamic inverse problems remains a challenge due to the trade-off between spatial and temporal resolution. Specifically, this occurs in MRI, where rapid anatomical motion and physiological changes demand high temporal fidelity, creating a trade-off between high spatial and temporal fidelity. An emerging direction in accelerated MRI reconstruction is the integration of deep learning techniques. Compared to more iterative methods like compressed sensing, deep learning methods offer improved reconstruction quality and also enable the possibility of real-time imaging. From the deep learning architectures, unsupervised learning strategies using neural networks have been proposed, as they do not require large datasets. Among these approaches, implicit neural representations (INRs) offer a strong framework by modeling data as continuous functions that map spatial and temporal coordinates to signal values. However, most existing approaches rely on positional encoding of the input coordinates, latent motion codes, and deformation networks to enhance temporal consistency in the presence of spatial undersampling. In this work, we incorporate an Optimal Transport-based regularization strategy into an implicit neural representation (INR) framework to enable higher spatiotemporal undersampling for physically plausible transitions between frames. Specifically, we incorporate temporal prior into the model through optimal transport (OT) regularization. We introduce two types of regularization: one based on the Wasserstein distance between consecutive frames and another based on a barycenter formulation. We demonstrate that both regularizers promote temporally coherent reconstructions and improve performance under high spatiotemporal subsampling rates.

Keywords: Dynamic MRI, Implicit Neural Networks, Optimal Transport, Inverse Problems, Spatiotemporal Regularization

Contents

1 Introduction

Magnetic resonance imaging (MRI) is a non-invasive, high-resolution imaging modality that uses the principles of nuclear magnetic resonance. In clinical practice, MRI is widely used for the detection and characterization of a broad range of conditions such as brain tumor growth [1], atrophy [2], and neurodegenerative disorders like Alzheimer [3]. MRI suffers from long acquisition times due to rapid signal decay [4] and hardware limits [5]. As a consequence, MRI scans are slow, discomfort patients (e.g., vulnerable groups such as pediatric, elderly, or claustrophobic patients), and have limited capacity. [6] Furthermore, the long scan times are susceptible to motion artifacts created by physiological processes such as respiration and cardiac activity, as well as involuntary patient movement. [7]

These issues have driven efforts to accelerate MRI scan times. Examples include the use of multiple receiver coils [8], increasing magnetic field strength [9], and improving hardware. [10] An alternative approach, in addition to the earlier hardware enhancements, is the MRI reconstruction of undersampled data, which is the focus of this research. The reconstruction of the image is an inverse problem, as the measured data consists of sampled Fourier coefficients rather than the image itself.

In efforts to reduce scan time, compressed sensing (CS) has become an essential technique over the past two decades. [11, 12] CS exploits the fact that redundant structures in MRI images become sparse in an appropriate transform domain, enabling accurate reconstruction from fewer measurements. Redundancy in the spatial domain is often achieved through methods such as Total Variation (TV) [13], or wavelet transforms [14]. In dynamic MRI, sparsity is exploited both spatially and temporally. Temporal redundancy can be captured using Fourier transforms for periodic motion, while Principal Component Analysis (PCA) for more irregular dynamics. Methods like k-t SPARSE [15] and k-t FOCUSS ()[16] combine spatial and temporal transforms to enhance reconstruction from undersampled data.

A special regularization strategy for spatiotemporal reconstruction, and not limited to MRI, is Optimal Transport (OT). OT models the transformation of probability densities by using a transport map, minimizing the total cost under conditions such as spatial continuity and mass conservation constrain. This makes it a compelling prior for nonlinear, non-rigid time series across various modalities, including cardiac MRI, dynamic computed tomography (CT), and others. For example, earlier work in [17] jointly estimates both the image sequence and a motion field regularized by dynamic OT, minimizing total kinetic energy over time to encourage temporally consistent and physically plausible reconstructions. Subsequent approaches include a template matching framework in [18] that quantifies frame differences by transport cost, and [19], which extends this approach by using adjacent frames as reference templates. In all these approaches, the emphasis lies on mitigating the impact of spatial undersampling within each frame. Temporal sampling between frames remains intact.

While these methods are applied without machine learning techniques, an emerging direction in accelerated MRI reconstruction is the integration of deep learning techniques. Compared to compressed sensing, deep learning methods offer improved reconstruction quality and also enable the possibility of real-time imaging [20]. However, deep learning learning models have poor generalization outside the trained dataset. [21]. Additionally, acquiring a large amount of high-quality data can be challenging due to the high costs and lengthy duration of scans [22]. Lastly, deep learning models may generate hallucinated

structures, which can be difficult to identify using conventional evaluation metrics [21].

To overcome the challenges associated with extensive training datasets, recent work has explored untrained neural networks that are directly fitted to individual patient measurements. The network is used as an implicit prior by capturing structural properties of the data through its architectural bias. A notable example is Deep Image Prior (DIP) [23], which uses a randomly initialized Convolutional Neural Network (CNN). The approach is extended to dynamic MRI to capture temporal consistency across frames.[24] However, CNNs rely on discrete grid representations, which limit their ability to model the continuous nature of dynamic MRI. Implicit Neural Representations (INRs) address this limitation by learning continuous mappings from spatial and temporal coordinates to image intensities, enabling memory-efficient storage and flexible sampling [25].

INRs have been applied to MRI reconstruction in several ways. In [26], Neural Implicit k-space (NIK) was introduced, learning a mapping from k-space coordinates (k_x, k_y, t) to complex k-space values. Within this category, a further distinction can be made between methods that rely on binned data and those that operate directly on raw, unbinned k-space. Binned data refers to dynamic MRI data temporally grouped into discrete time frames. This is used in [27]. The other group [26, 28] uses raw k-space to fit the INR directly to the original measurements. The work [29, 28] is a further extension of [26], utilizing additional regularization techniques such as total variation and nuclear norms.

The second option is to utilize the network to predict the image itself, and then use a Non-uniform Fast Fourier transform (NUFFT) to transform the prediction back to the measurement space [30, 31, 32, 30, 33]. These methods commonly employ compressed sensing regularization, typically applied in the spatial domain, to promote sparsity and improve reconstruction.

The previously mentioned INR methods above impose some form of temporal regularization through the positional encoding of input coordinates, latent-motion codes, and deformation networks. Nonetheless, these approaches do not guarantee physically meaningful or consistent motion patterns. By contrast, the Optimal Transport (OT) formulation enforces continuity and mass conservation, yielding geodesic interpolations even with irregularly sampled spatial coordinates or missing time points. Therefore, we extend an INR with an OT prior: the INR addresses spatial undersampling, and OT provides a mathematical temporal constraint that aligns consecutive frames, producing reconstructions that are possibly both sharper and more consistent than existing deep learning approaches.

1.1 Contributions

In this thesis, we develop a dynamic MRI reconstruction framework based on implicit neural representations (INRs). The INR is used on spatial and spatial-temporal undersampled k-space data. The focus is on improving temporal coherence across frames. To achieve this, we explore two different temporal priors based on Optimal Transport (OT) methods. The temporal priors are based on Optimal Transport (OT) methods.

The first is the total Wasserstein distance between consecutive frames in the image domain. The second prior utilizes Wasserstein barycenters of the missing frames and measures the L2 difference relative to the INR prediction. Our objective is to evaluate whether the combination of INRs with an OT prior can lead to improved temporal consistency and overall reconstruction quality in undersampled dynamic MRI data.

1.2 Thesis outline

This thesis begins in Chapter 2 with the theoretical background. It introduces inverse problems and describes the MRI acquisition process. This is followed by an overview of implicit neural representations (INRs), including SIREN architectures, and a formal introduction to Optimal Transport (OT), with a focus on the Wasserstein distance and barycenters.

Chapter 3 presents the proposed model, which combines coordinate-based neural networks with loss functions inspired by optimal transport. It details the architecture of the implicit neural representation (INR), the motivation for incorporating optimal transport, the inference procedure, and the implementation aspects of the method.

Chapter 4 evaluates the method through experiments on synthetic and real MRI datasets. The results are discussed in Chapter 5, and Chapter 6 concludes the thesis with a summary of the main findings and directions for future work.

2 Theoretical Background

In this section, we will review the main components of MRI reconstruction. Since MRI reconstruction is fundamentally an inverse problem, we start by presenting its general mathematical formulation. This is followed by a brief overview of the MRI acquisition process and its inverse formulation. We then discuss the non-machine learning baseline used for comparison. Next, we describe the deep learning model used for reconstruction. Finally, we introduce the concept of Optimal Transport.

2.1 Inverse problems

An inverse problem refers to the task of estimating an unknown signal or image u from the measurements f^δ, where the data is generated via a (possibly non-linear) forward operator A and contaminated by noise ϵ. This process is described by the equation:

$$f^\delta = A(u) + \epsilon, \tag{1}$$

where f^δ is the noisy data measurements, A models the underlying physical measurement: $A : U \to F$ between the Banach spaces U and F, with their respective norm $\|\cdot\|_U$ and $\|\cdot\|_F$. The term ϵ represents the combined measurement error and is assumed to satisfy: $||\epsilon|| \leq \delta$.

Finding the solution becomes challenging when a small perturbation in the data f^δ, like noise, results in significant variations in the solution u. The problem is then referred to as ill-posed. A problem is considered ill-posed when one or more of the conditions for a well-posed problem are not satisfied. According to Hadamard, a problem is well-posed if it satisfies the following three properties:

Definition 2.1 (Well-posedness inverse problems)**.** An inverse problem is called well-posed if the following three conditions hold:

1. **Existence:** For every admissible measurement f there exists at least one solution $u \in U$ such that $A(u) = f$.
2. **Uniqueness:** For every measurement f the solution is unique.
3. **Stability (continuous dependence):** There exists a constant $C > 0$ such that for all data pairs $f_1, f_2 \in F$ with corresponding solutions $u_1, u_2 \in U$,

$$\|u_1 - u_2\|_U \ \leq \ C \, \|f_1 - f_2\|_F.$$

A popular method for solving the problem is to use a variational formulation. The variational problem minimizes two different parts. The first part, data fidelity $\mathcal{D}$, which enforces consistency between estimated measurements ($A(u)$) and the measured data f. For additive Gaussian noise, the following data consistency term is typically used: $\mathcal{D}(u, f) = \frac{1}{2}\|u - f\|_2^2$ is used. This choice corresponds to the negative log-likelihood of a Gaussian noise model. It leads to a statistically unbiased estimator under the assumption that the noise is zero-mean and independent. The second part is the regularization $R(u)$ and imposes prior knowledge or desired properties on the solution. The final variational formulation is as follows:

$$\hat{u} = \min_{u \in U} D(u, f) + \lambda R(u) \tag{2}$$

2.2 MRI

2.2.1 Signal formation and spatial encoding

This section describes the magnetization dynamics during a short and specific moment of signal acquisition in MRI. We refer to *Medical Image Reconstruction: A Conceptual Tutorial* for a more detailed explanation of how this state is physically prepared.

We consider a two-dimensional plane in which each location $\vec{r} = (x, y)$ contains hydrogen nuclei. Each hydrogen nucleus consists of a single proton with a quantum property called "spin." In a magnetic field, these spins behave like tiny rotating magnetic dipoles. The combined effect of many such spins in a small region gives rise to a measurable magnetization vector. At each point, we assume the presence of a transverse magnetization vector that rotates in the plane along a circular path. This vector represents the combined effect of many spins in phase at that location. Initially, we assume that the transverse magnetization vectors at all positions share the same phase and rotate at the same frequency, resulting in a coherent rotating field across the entire plane. The direction of this vector evolves, while its magnitude remains constant throughout the acquisition. We denote this rotating vector field by a complex-valued function:

$$u(\vec{r}) = u_x(\vec{r}) + iu_y(\vec{r}),$$

where the real and imaginary parts correspond to the x- and y-components of the transverse magnetization.

We apply gradient fields that slightly modify the magnetic field across space to encode spatial information into the measured signal. A gradient field is a magnetic field whose strength varies linearly with position in a given direction. For example, a gradient in the y-direction with gradient amplitude G_y creates a magnetic field

$$B(y) = B_0 + G_y y,$$

where B_0 is the main static magnetic field of the MRI scanner, and G_y is the strength of the applied gradient along the y-axis. Spins at different y-positions thus experience slightly different magnetic field strengths, resulting in position-dependent precession frequencies. Similarly, a gradient in the x-direction with gradient amplitude G_x creates

$$B(x) = B_0 + G_x x,$$

encoding spatial information along the x-axis. By applying gradients G_x and G_y during the acquisition, we can distinguish spatial locations based on the unique frequency and phase evolution of the spins, enabling the reconstruction of two-dimensional images.

The first step is to apply a magnetic field gradient in the y-direction for a short time. This changes the local magnetic field strength as a function of y, which temporarily shifts each spin's frequency. Although the frequency returns to its original value afterward, each position y has accumulated a different phase. This leads to a position-dependent modulation of the vector field:

$$u(\vec{r}) \to u(\vec{r}) \cdot \exp\left(-i \int_0^T \gamma G_y y \, dt\right) = u(\vec{r}) \cdot \exp(-i\gamma G_y T y).$$

After this, a magnetic field gradient in the x-direction is applied during the signal acquisition period. This gradient modifies the rotation frequency of the magnetization depending on the x-position. During the acquisition period, the gradient field induces a new frequency that remains constant at each spatial location. The following exponential modulation of the magnetization can describe this frequency:

$$u(\vec{r}) \to u(\vec{r}) \cdot \exp\left(-i \int_T^t \gamma G_x x \, dt'\right) = u(\vec{r}) \to u(\vec{r}) \cdot \exp\left(\gamma G_x (t - T)\right)$$

Combining both steps, the transverse magnetization at time t and position $\vec{r} = (x, y)$ is given by:

$$m_{xy}(t, \vec{r}) = u(\vec{r}) \cdot \exp\left(-i \left[\gamma G_y T y + \gamma G_x (t - T)\right]\right).$$

The MRI receiver coil does not measure each position separately. Instead, it captures the global magnetic field of all transverse magnetization vectors combined. This collective field rotates over time, and according to Faraday's law of induction, any time-varying magnetic field passing through a conducting loop induces an electrical voltage. The receiver coil in MRI is such a loop placed around object that measures the total induced voltage. The received signal can be written as

$$s(t) = \int_{\mathbb{R}^2} m_{xy}(t, \vec{r}) \, d\vec{r} = \int_{\mathbb{R}^2} u(\vec{r}) \cdot \exp\left(-i \left[\gamma G_y T y + \gamma G_x (t - T) x\right]\right) d\vec{r},$$

which represents the two-dimensional Fourier transform of the transverse magnetization $u(\vec{r})$, evaluated at spatial frequencies $k_x(t)$ and k_y.

We now define the spatial frequency coordinates:

$$k_y = \frac{\gamma}{2\pi} G_y T, \qquad k_x(t) = \frac{\gamma}{2\pi} G_x (t - T),$$

where k_x and k_y represent the positions in Fourier space sampled by the MRI scanner at time t. These values form the coordinate pair

$$\vec{k}(t) = (k_x(t), k_y),$$

which traces out a line in k-space as time progresses during the readout.

By repeating this measurement for different values of the y-gradient G_y, we obtain multiple lines in the spatial frequency domain, known as k-space. Each measurement corresponds to a different fixed k_y-value, while the variation in time during acquisition traces a line in the k_x-direction. Once enough lines are collected, the image can be reconstructed using an inverse Fourier transform. The resulting image typically corresponds to the magnitude $|u(\vec{r})|$, which reflects the local signal intensity in clinical MRI scans. Figure 1 is a fully sampled k-space displayed with the corresponding Fourier transform.

2.2.2 MRI as an inverse problem

In static MRI, the goal is to reconstruct a complex-valued image $u \in \mathbb{C}^m$ from a subset of its Fourier coefficients $f^\delta \in \mathbb{C}^n$, where $n \ll m$. The measurement process is modeled by a forward operator $A \in \mathbb{C}^{n \times m}$, which typically consists of a subsampled discrete Fourier transform, possibly modulated by coil sensitivities. The measurement model is given by:

$$f^\delta = A(u) + \epsilon, \tag{3}$$

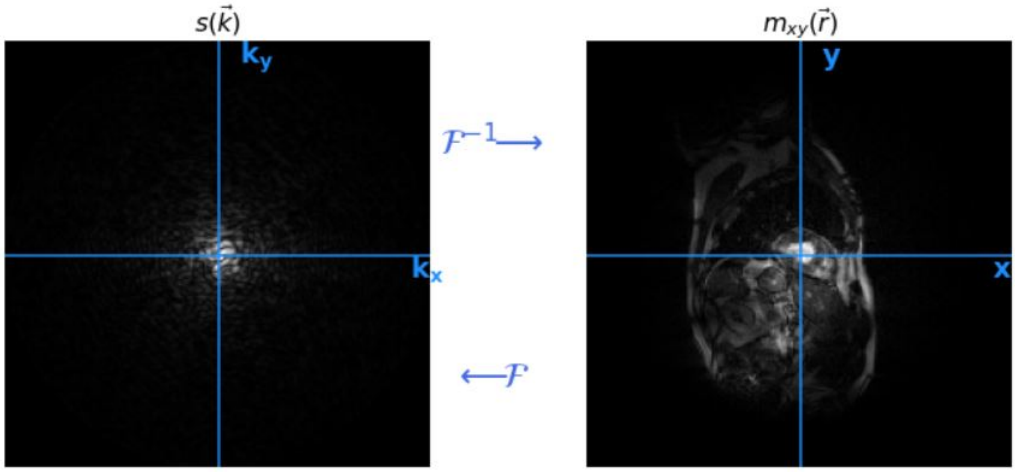


Figure 1: Left: The fully sampled k-space data, representing spatial frequency information acquired during MRI. Right: The corresponding image-space representation obtained by applying an inverse Fourier transform.

where $\epsilon \sim \mathcal{N}_{\mathbb{C}}(0, \sigma^2 I)$ represents additive complex Gaussian noise [4]. The inverse problem involves reconstructing u from the noisy and incomplete data f, which is ill-posed due to undersampling.

Dynamic MRI In dynamic MRI the goal is to reconstruct a time-varying image $u : \Omega \times [0, T] \to \mathbb{C}$. Each time frame may be sampled differently, resulting in a time-dependent sampling mask $S^t : \Sigma \to \{0, 1\}$, where $\Sigma \subset \mathbb{R}^2$ is the k-space domain. Let $u(\cdot, t)$ denote the image at time t, F be the non-uniform Fast Fourier Transform ($\mathcal{F}$), then the forward model becomes:

$$y_t = A_t(u(\cdot, t)) + \epsilon_t = \mathcal{F}(S^t u(\cdot, t)) + \epsilon_t. \tag{4}$$

Due to the trade-off between temporal and spatial resolution in dynamic MRI, k-space data are often grouped into temporal bins of duration Δt. This binning increases the effective sampling density per frame, enabling reconstruction with adequate spatial resolution. Without binning, the undersampling per frame would be too severe.

2.3 Traditional Reconstruction: IGRASP

For comparison with our proposed method, we use the iterative Golden-angle Radial Sparse Parallel (IGRASP) approach, as it has been successfully translated into clinical use and evaluated on a large scale for clinical feasibility, yielding convincing results for patients with regular breathing activity. [34]

The IGRSAP is an iterative reconstruction method that combines the classic parallel imaging (PI) approach with compressed sensing (CS). Here PI estimates the spatial sensitivity profile of the data, improving the noise reduction in the classical iterative approach [35]. Then compressed sensing is used in the temporal direction to promote sparsity. The original study by Feng et al. [36] demonstrated that this joint approach significantly outperforms either PI or CS alone, particularly in highly undersampled radial acquisitions.

The reconstruction problem is formulated as a variational problem:

$$\hat{x} = \arg\min_x \|\mathcal{F}Su - f\|_2^2 + \lambda \|T(u)\|_1 , \tag{5}$$

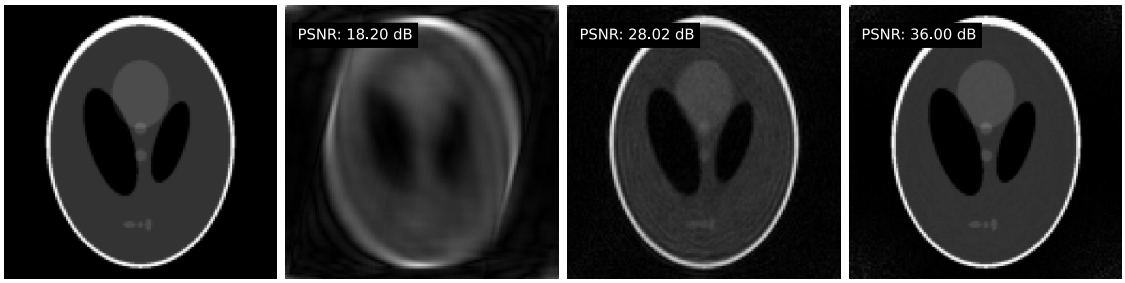
Where u is the image time series in x–y–t space, f are the acquired multi coil k-space data, S are the coil sensitivity maps, $\mathcal{F}$ denotes the non-uniform FFT operator (NUFFT) defined on the radial trajectory, and $T(u)$ applies temporal total variation as a sparsity constrain promoting gradual change between frames. The parameter λ controls the tradeoff between data consistency and temporal regularity. The temporal total variation operator $T(u)$ is defined as:

$$\|T(u)\|_1 = \sum_{x,y,t} |u(x, y, t+1) - u(x, y, t)| \,. \tag{6}$$

Following the original iGRASP implementation, the regularization weight is set to $\lambda = 0.05 \cdot M_0$, where M_0 refers to the maximum magnitude of the initial image reconstructed from the undersampled data. However, a slight difference lies in how the weights are updated, as the original paper employs a nonlinear conjugate gradient (CG) method. In contrast, we use the standard conjugate gradient method due to the support provided by the TensorFlow-MRI library.

2.4 Implicit neural networks

Implicit Neural Representation (INR) is a neural network representing a signal. Rather than storing discrete samples, an INR maps a measured continuous input (e.g., a coordinate, angle, time, or combination) to a signal value, enabling efficient representation. [25] For this, a multilayer perceptrons (MLP) are used. However, the classic use of the ReLU activation function has a spectral bias towards learning lower frequencies, resulting in a blurry image, as shown in 2b. To address this limitation, two common approaches have emerged: the use of alternative activation functions and the application of positional encoding. These strategies will be discussed after first introducing the standard multi-layer perceptron (MLP). Lastly, the specific model architecture used in this work (SIREN) is presented.



(a) Target (b) MLP-ReLU (c) ReLU + Positional encoding (d) SIREN

Figure 2: Reconstruction example of a Shepp-Logan phantom using the approaches described in model section. Each network was optimized for 5000 iterations. The displayed PSNR values reflect the reconstruction quality achieved after training.

2.4.1 Multi-layer perceptron

Multi-layer perceptrons are a fundamental building block of modern machine learning. Inspired by the functioning of biological neurons, these models are used to represent complex, nonlinear functions. Their versatility has led to widespread use in domains such as

denoising of X-ray images corrupted with additive white Gaussian noise [37] or PET Image Reconstruction [38].

An MLP consists of three types of layers: an input layer, one or more hidden layers, and an output layer. The input layer receives features or dimensions of the input data, and its size corresponds to the input dimensionality. Each neuron in a hidden layer receives input from all neurons in the previous layer and passes its output to the next layer. The number of hidden layers and the number of neurons per layer are hyperparameters determined during the model design phase. The output layer produces the final prediction, the dimensionality of which depends on the specific task. [39]

For a layer t with n_t neurons, the output for the i-th neuron in layer t is computed as:

$$a_i^t = \sigma\left(z_i^t\right), \quad z_i^t = \sum_{j=1}^{n_{t-1}} w_{ij}^t \, a_j^{t-1} + b_i^t, \tag{7}$$

where:

- a_j^{t-1} is the activation (output) of neuron j from the previous layer $t-1$
- w_{ij}^t is the weight of the connection from neuron j to neuron i
- b_i^t is the bias term for neuron i
- z_i^t is the pre-activation (linear combination)
- $\sigma(\cdot)$ is a nonlinear activation function

This neuron-wise computation can be expressed more compactly by viewing the entire network as a composition of nonlinear functions. For an n-layer network, we define:

$$F(\mathbf{x}) = (\varphi_n \circ \varphi_{n-1} \circ \cdots \circ \varphi_1)(\mathbf{x}), \quad \varphi_i(\mathbf{x}) = \sigma(\mathbf{W}^i\mathbf{x} + \mathbf{b}_i). \tag{8}$$

Here, $\mathbf{a}^{(i-1)} \in \mathbb{R}^{w_{i-1}}$ denotes the input vector to layer i, $\mathbf{W}^{(i)} \in \mathbb{R}^{w_i \times w_{i-1}}$ is the weight matrix, and $\mathbf{b}^{(i)} \in \mathbb{R}^{w_i}$ is the corresponding bias vector. The output of the layer is denoted by $\mathbf{a}^{(i)} \in \mathbb{R}^{w_i}$.

Regarding the choice of the activation function $\sigma()$, the linear mapping $\sigma(x) = x$ is avoided because it results in an affine transformation, limiting the representation of the network to linear functions. Hence, nonlinear activation functions are essential for approximating complex relations. Common choices for activation functions includes sigmoid and ReLU [39], with sine [25] and Gaussian activations [40] being particularly relevant for implicit neural representations. The functions are visualized in figure 3.

For the final layer, a different activation function is often applied depending on the task. The output $\mathbf{o} \in \mathbb{R}^m$ of the neural network can be computed as:

$$\mathbf{o} = \sigma_o\left(\mathbf{W}_o\,\mathbf{a}^{(n)} + \mathbf{b}_o\right), \tag{9}$$

Where:

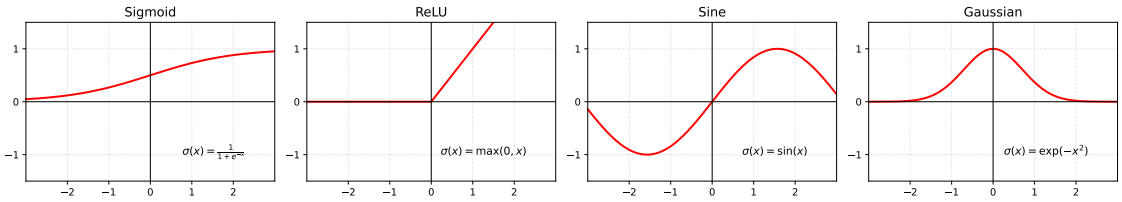


Figure 3: Common activation functions used in neural networks. From left to right: Sigmoid, ReLU, sine, and Gaussian. Each function introduces nonlinearity differently, influencing the network's ability to approximate complex functions.

- $\mathbf{a}^{(n)} \in \mathbb{R}^{w_n}$ is the output of the last hidden layer, where w_n denotes the number of neurons in that layer
- $\mathbf{W}_o \in \mathbb{R}^{m \times w_n}$ is the weight matrix connecting the last hidden layer to the output layer
- $\mathbf{b}_o \in \mathbb{R}^m$ is the bias vector for the output layer
- σ_o is the activation function applied to the output layer.

When we deal with a regression task, one might take the identity function as $\sigma(x) = x$. The network can produce continuous valued outputs without constraining them to a specific range, often needed for pixel intensities predictions.

2.4.2 INRs and spectral bias

The MLP networks tend to learn primarily the low-frequency attributes of the training data.[41] The phenomenon is known as spectral basis, which causes blurry reconstructions, particularly when high-frequency spatial details are underrepresented [41]. Two different strategies are often used to address the spectral bias introduced by the network: alternative activation functions or Fourier feature mapping. We further discuss the two approaches below.

Fourier feature mapping The new network with Fourier mapping $F_\theta^{'}$ is a composition of the encoding function γ and MLP network F_θ. The new formulation becomes: $F'_\theta(x) = (F_\theta \circ \gamma)(\mathbf{x})$. The most basic form of Fourier feature mapping wraps input coordinates around the circle.

$$\gamma(\mathbf{x}) = [\sin(2\pi\mathbf{x}), \cos(2\pi\mathbf{x})]^T \tag{10}$$

An extension is to introduce a Gaussian matrix B:

$$\gamma(\mathbf{x}) = [\sin(2\pi\mathbf{B}\mathbf{x}), \cos(2\pi\mathbf{B}\mathbf{x})]^T, \tag{11}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the input coordinate, and $\mathbf{B} \in \mathbb{R}^{m \times d}$ is sampled from $\mathcal{N}(0, \sigma^2)$ and σ^2 is chosen for each task and dataset with a hyperparameter sweep. The size of input coordinate increases to $2m \cdot d$.

An alternative to the random Gaussian matrix is to construct the encoding matrix $\mathbf{B}$ as a diagonal matrix with log-linearly spaced frequency values:

$$\mathbf{B} = \text{diag}(\sigma_0, \sigma_1, \ldots, \sigma_{m-1}), \quad \text{with} \quad \sigma_j = \sigma \cdot 2^{j/m},$$

where σ is a base frequency scale chosen through hyperparameter tuning. This formulation encodes each input dimension independently and is often referred to as positional encoding [42]. It is commonly used in applications such as NeRF [43], where the fixed frequency structure provides a strong prior over the type of signal expected.

All three approaches show an improvement over no mapping at all; however, experiments indicate that the Gaussian matrix produces the best performance. [42]

A more recent approach is the use of multi-scale hash encoding [44], which has shown impressive results in real-time geometry representation. However, the paper is focused on dense, supervised signals without complex forward models. Since our dynamic MRI reconstruction problem involves undersampled measurements and a non-trivial forward model, we do not look at hash encoding in this thesis.

Periodic Activation Functions This approach for modeling high-frequency signals utilizes periodic activation functions. This idea, introduced in SIREN [25], substitutes the commonly used ReLu activation with sine activations for every layer:

$$\Phi(x) = (\varphi_n \circ \cdots \circ \varphi_1)(x), \quad \varphi_i(x) = \sin(\omega(\mathbf{W}_i x + \mathbf{b}_i)), \tag{12}$$

Where ω is a hyperparameter. While this approach further enhances high-frequency learning, it also enables modeling capabilities that positional encoding + ReLU alone cannot provide, such as the ability to represent non-zero higher-order derivatives. In contrast, ReLU networks, which are piecewise linear, have derivatives of order two and higher that are always zero.

Alternative activation functions have also been proposed, such as Gaussian $(\exp(-(sx)^2)$ [45], Gabor wavelet $(\exp(-(sx)^+ i\omega x))$ [40], hosc $(\tanh(\beta \sin()))$ [46], sinc $(\frac{sin(\omega x)}{x})$ [47], where s, ω, β are hyperparameter and x the input. We adopt the SIREN network approach.

The performance of a siren can be seen in Figure 2. Here, it can be visually seen and by the performance metric that the siren network outperforms the positional one. Therefore, we use a SIREN network for the neural implicit representations in this thesis.

SIREN is an MLP perceptron with space-time coordinates (x,y,t) as input to predict the signal intensity. The activation is $\sin(\omega(\mathbf{W}^i \mathbf{x} + \mathbf{b}_i))$, where ω is the hyperparameter balancing the convergence and expressiveness of the network. We use $\omega = 30$ following the original paper [25].

Siren requires a modified initialization scheme to ensure stability during training. The weights of the first layer are sampled from a uniform distribution given by:

$$W_{ij}^{(1)} \sim \mathcal{U}\left(-\frac{1}{n}, \frac{1}{n}\right),$$

where n is the number of hidden neurons of the previous layer. For the rest of the layers is, the following distribution is used:

$$W_{ij}^{(l)} \sim \mathcal{U}\left(-\sqrt{\frac{6}{n}} \cdot \frac{1}{\omega}, \sqrt{\frac{6}{n}} \cdot \frac{1}{\omega}\right), \quad \text{for } l > 1,$$

2.5 Optimal transport

Optimal Transport quantifies the minimal amount of work required to transform one probability measure into another, given a specified cost function. Before formulating the optimal transport problem, we first introduce key definitions from measure theory to define the function spaces in which the optimization problem is posed. Optimal transport has two formulations: Monge's formulation and Kantorovich's formulation. Kantorovich's formulation can be seen as a relaxation of Monge's original problem. While Monge seeks a deterministic transport map that moves mass directly from source to target, Kantorovich allows for probabilistic couplings between the two measures, which makes the problem convex and always admits a solution under mild conditions.[48] In this thesis, we start with Monge's formulation, followed by Kantorovich's formulation. For the calculation of the Kantorovich formulation, an approximation of the exact solution is used. For discrete probability measures supported on n points, solving the exact optimal transport problem requires $\mathcal{O}(n^3 log(n))$ time.[48] To address this, [49] proposed adding an entropic regularization term to the Kantorovich objective, leading to a strictly convex and smooth optimization problem that can be solved efficiently via the Sinkhorn algorithm.[49]

2.5.1 Measure theory

Measure theory formalizes the notion of size for abstract sets, enabling flexible definitions of length across different contexts. The key concepts used here follow [50], to which we refer for more detailed exposition.

Definition 2.2 (power set)**.** Let X be a finite set. The power set of X, denoted by $P(X)$, is the set of all subsets of Ω, including the empty set $\emptyset$ and X itself.

Definition 2.3 (σ-algebra)**.** Let X be a non–empty set. A *σ-algebra* subset on X, denoted as $\Sigma \subseteq P(X)$ where $P()$ is the powerset, only if the properties of an algebra set hold (i-iii) and in addition property (iv):

(i) full and empty set: $\emptyset, X \in \Sigma$

(ii) closed under finite intersections $E_1, E_2 \in \Sigma$ then $E_1 \cap E_2 \in \Sigma$

(iii) closed under complement $A \in \Sigma \implies X \setminus A \in \Sigma$

(iv) closed under countable unions $\{A_n\}_{n=1}^{\infty} \subseteq \Sigma \implies \bigcup_{n=1}^{\infty} A_n \in \Sigma$

Definition 2.4 (Measurable space)**.** Given a set X and itself σ-algebra set ($\Sigma \subset X$). Then the tuple (X, Σ) is called the measurable space, where the elements of Σ are called the measurable set.

Definition 2.5 (Borel σ-algebra)**.** If X carries a topology τ, the *Borel σ-algebra* on X, denoted $\mathcal{B}(X)$, is the smallest σ-algebra that contains all open sets:

$$\mathcal{B}(X) \;=\; \sigma(\tau) \;=\; \bigcap\big\{\Sigma \subseteq P(X) \;:\; \Sigma \text{ is a } \sigma\text{-algebra and } \tau \subseteq \Sigma\big\}.$$

The measurable space $(X, \mathcal{B}(X))$ is called a Borel space, and the sets in $\mathcal{B}(X)$ are called *Borel sets*.

Definition 2.6 (Measure)**.** Let (X, Σ) be a measurable space. A map $\mu\colon \Sigma \to [0, \infty]$ is a measure if

1. $\mu(\varnothing) = 0$;
2. $\mu(A) \geq 0$ for every $A \in \Sigma$;
3. σ-additivity: For any pairwise-disjoint sequence $\{A_n\}_{n=1}^{\infty} \subseteq \Sigma$,

$$\mu\Big(\bigcup_{n=1}^{\infty} A_n\Big) = \sum_{n=1}^{\infty} \mu(A_n).$$

Now we can finally define the object where optimal transport operates, called the probability space. The space given as tuple triple (X, Σ, μ), where μ is the measure on (X, Σ) with the extra condition $\mu(X) = 1$.

2.5.2 From Monge formulation to Kantorovich relaxation

The contents of this section draw from the theory in [50, 48]. The last definition we need is the formal definition of a mapping from the distribution called the push-forward measure. Let $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ be borel measurable spaces, and let $T : \Omega_1 \to \Omega_2$ be a measurable map. Given a probability measure μ on Ω_1, we define the push-forward measure as follows:

Definition 2.7 (Push-forward operator)**.** Let μ be a probability measure on $(\Omega_1, \mathcal{F}_1)$ and let $T : \Omega_1 \to \Omega_2$ be a measurable map. The *push-forward measure* $T_{\#}\mu$ is a probability measure on $(\Omega_2, \mathcal{F}_2)$ defined by:

$$T_{\#}\mu(A) := \mu(T^{-1}(A)), \quad \text{for all } A \in \mathcal{F}_2.$$

Intuitively, the push-forward measure describes how mass is transported from the source space to the target space via the map T. We now introduce the Monge formulation of the optimal transport problem.

The Monge problem is given as:

Definition 2.8 (Monge problem)**.** Given two probability measures $\mu \in \mathcal{P}(\mathbb{R}^d)$ and $\nu \in \mathcal{P}(\mathbb{R}^d)$ the space of Borel probability measures, and a cost function $c : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, the Monge problem is to find a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_{\#}\mu = \nu$, and such that the total transport cost is minimized:

$$\inf_{T_{\#}\mu = \nu} \int_{\mathbb{R}^d} c(x, T(x))\, d\mu(x).$$

A frequently used cost function in optimal transport is the p-th power of the Euclidean norm, which gives rise to the so-called p-Wasserstein distance. The total transport cost induced by this choice is denoted by $W_p(\mu, \nu)$, and is formulated as:

Definition 2.9 (Monge formulation with p-Wasserstein distance)**.** Let $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$ be probability measures with finite p-th moments, and let $p \in [1, \infty)$. The Monge formulation of the p-Wasserstein distance is given by

$$W_p(\mu, \nu) := \left(\inf_{T_{\#}\mu = \nu} \int_{\mathbb{R}^d} \|x - T(x)\|^p \, d\mu(x) \right)^{1/p},$$

where the infimum is taken over all measurable maps $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_{\#}\mu = \nu$.

The Monge formulation is limited in two important ways: It does not allow for mass to be split; each point x must be mapped to a single destination $T(x)$. Second, it leads to a non-convex optimization problem, which is often ill-posed or lacks solutions in practical settings. To overcome these issues, Kantorovich introduced a relaxed formulation based on transport plans rather than maps.

Definition 2.10 (Kantorovich relaxation)**.** Let $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$ be Borel probability measures, and let $p \in [1, \infty)$. The p-Wasserstein distance between μ and ν is defined via the Kantorovich relaxation as:

$$W_p(\mu, \nu) := \left(\inf_{\pi \in \Pi(\mu, \nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|^p \, d\pi(x, y) \right)^{1/p},$$

where $\Pi(\mu, \nu)$ denotes the set of all transport plans (couplings) between μ and ν, i.e.,

Definition 2.11. The set of transport plans between μ and ν is defined as:

$$\Pi(\mu, \nu) := \left\{ \pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d) \;\middle|\; \begin{array}{l} \pi(A \times \mathbb{R}^d) = \mu(A), \\ \pi(\mathbb{R}^d \times B) = \nu(B), \\ \text{for all measurable } A, B \subset \mathbb{R}^d \end{array} \right\}.$$

If we take the probability measures to lie in real space and use the squared $p = 2$ Wasserstein distance, the resulting quantity can be interpreted as the minimal kinetic energy required to transport one measure into the other. The dynamic formulation introduces $(\alpha_t)_{t=0}^1$, the minimal-length path of intermediate measures, and v_t, a time-dependent velocity field. This leads to the Benamou–Brenier formulation [51]:

$$\begin{aligned} W_2^2(\mu, \nu) &= \min_{\alpha_t,\, v_t} \int_0^1 \int_{\mathbb{R}^d} \|v_t(x)\|^2 \, d\alpha_t(x) \, dt \\ &= \min_{\alpha_t,\, v_t} \int_0^1 \|v_t(x)\|^2_{L^2(\alpha_t)} \, dt \\ \text{s.t.} \quad &\left\{ \tfrac{\partial \alpha_t}{\partial t} + \nabla \cdot (\alpha_t v_t) = 0, \quad \alpha_{t=0} = \mu, \alpha_{t=1} = \nu. \right. \end{aligned} \tag{13}$$

2.5.3 Sinkhorn approximation

Finding the exact W_p distance is an expensive computation with a big $\mathcal{O}(n^3 \log(n))$ where n is the elements of the distribution. Therefore, the Sinkhorn approximation is used. The approximation introduces a Shannon entropic regularization term into the Kantorovich formulation, yielding a strictly convex optimization problem that can be solved efficiently through iterative updates, thereby reducing the complexity to $\mathcal{O}(n^2)$.

The Sinkhorn approximation is presented here for the 2D case, since we are interested in 2D images. Assuming there are a total of n bins with positions $\{x_i\}_{i=1}^n$, the discretized distributions of interest can be written as

$$P = \sum_{i=1}^{n} p_i \, \delta_{x_i}, \quad Q = \sum_{i=1}^{n} q_i \, \delta_{x_i},$$

where δ_{x_i} denotes the Dirac delta function at location $x_i \in \mathbb{R}^2$. For the $p = 2$ Wasserstein distance, the cost matrix is defined by $c_{ij} = \|x_i - x_j\|^2$,

and the transport plan is given by $T \in \mathbb{R}^{n \times n}$. The entropic optimal transport formulation in this discretized setting becomes

$$W_p^\varepsilon = \min_{\Pi} \sum_{i=1}^{n} \sum_{j=1}^{n} \pi_{ij} \, c_{ij} + \varepsilon \sum_{i=1}^{n} \sum_{j=1}^{n} \pi_{ij} \log \pi_{ij}$$
$$\text{s.t.} \quad \sum_{j=1}^{n} \pi_{ij} = a_i, \quad \sum_{i=1}^{n} \pi_{ij} = b_j, \quad \pi_{ij} \geq 0.$$

Where ε is the entropic regularization parameter, often referred to as the temperature. As $\varepsilon \to 0$, the approximation converges to the original Kantorovich optimal transport solution. The computational complexity of the Sinkhorn algorithm in this setting is approximately $\widetilde{\mathcal{O}}\left(\frac{n^2}{\varepsilon}\right)$ [52].

Figure 4 illustrates the transportation plans Π obtained for increasing values of the entropic regularization parameter ε between two Gaussian distributions. As ε increases, the couplings become progressively smoother, resulting in blurrier transport plans. The first image shows the cost matrix, defined by $c_{ij} = \|x_i - x_j\|^2$.

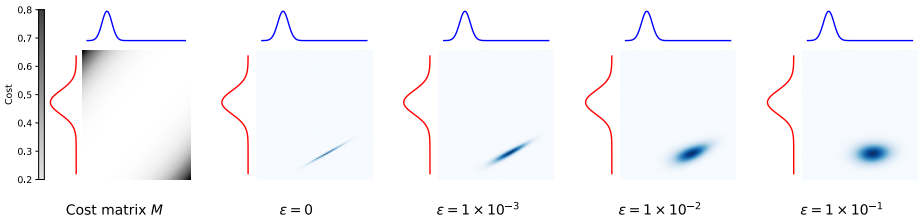


Figure 4: Cost matrix C (with $c_{ij} = \|x_i - x_j\|^2$) and transportation plans for different entropic regularizations γ, computed using the 1D Wasserstein distance with $p = 2$. The marginal distributions $P, Q \in \text{Prob}([0, 1])$ are shown alongside each transport plan.

2.5.4 Wasserstein barycenters

Wasserstein barycenters provide a structured approximation for interpolating between two or more probability distributions (images) based on the previously described Wasserstein distance. The barycenter accounts for underlying mass transport and spatial structure.

In the context of dynamic imaging, this enables the generation of meaningful intermediate frames to guide the model's prediction for missing frames, thereby enforcing realistic transformations between measured time steps that lie along a geodesic path. One such application is in manifold learning of dynamic images [53]. We now provide the formal definition of the Wasserstein barycenter:

Definition 2.12 (Wasserstein barycenters)**.** Let $\mu_i \in P(R^d)$ for $i \in \{1, \ldots, n\}$. Then for some sequence α_i with $\sum_{i=1}^{n} \alpha_i = 1$, a Wasserstein bary center μ is defined as

$$\mu = \arg\min_{\mu \in P(R^d)} \sum_{i=1}^{n} \alpha_i W_2^{\epsilon}(\mu_i, \mu) \tag{14}$$

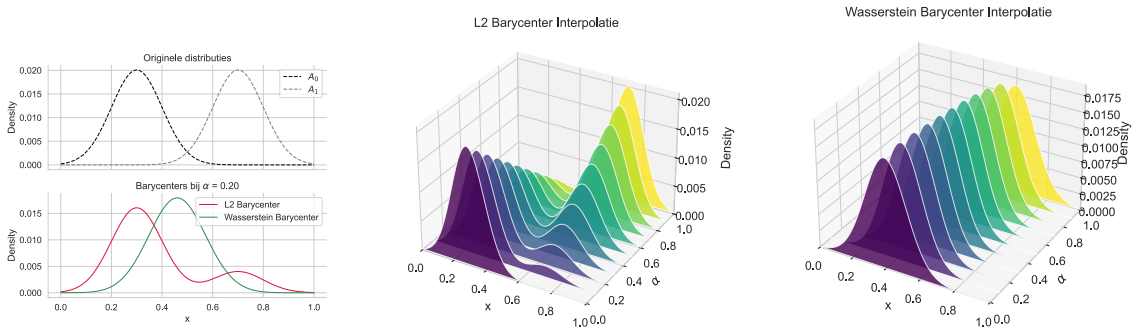
In particular, if we only have two reference images with positions 0 and 1 respectively, then $\lambda \in [0, 1]$, we can rewrite it to the following equation:

$$B(\mu_1, \mu_2, t) = \arg\min_{\mu \in P(R^d)} (1 - \lambda) W_2^{\epsilon}(\mu_1, \mu) + \lambda W_2^{\epsilon}(\mu_2, \mu) \tag{15}$$

For the calculations, the Convolutional Wasserstein Distances approach [54] is used. The implementation by [53] is employed for the experiments. To intuitively demonstrate the concept of a Wasserstein barycenter, consider two probability measures $\mu_0 = \mathcal{N}(m_0, \sigma_0^2)$ and $\mu_1 = \mathcal{N}(m_1, \sigma_1^2)$ defined on $\mathbb{R}$. The Wasserstein barycenter μ_λ, with interpolation parameter $\lambda \in [0, 1]$, is defined as the distribution that minimizes the weighted sum of squared Wasserstein distances to μ_0 and μ_1:

$$\mu_\lambda = \arg\min_{\mu} (1 - \lambda)\, W_2^2(\mu, \mu_0) + \lambda\, W_2^2(\mu, \mu_1). \tag{16}$$

In the one-dimensional Gaussian case, the barycenter is also Gaussian [50]. Figure 5 shows two input Gaussians and their barycenter for $\lambda = 0.2$. The plot illustrates how the barycenter lies between the two inputs, both in terms of mean and standard deviation. Crucially, this interpolation follows the geodesic under the L^2-Wasserstein metric, rather than a pointwise (density) average.



(a) Originele distributies (b) L2-barycenter (c) Wasserstein-barycenter

Figure 5: **(a)** shows the original input distributions A_0 and A_1. **(b)** displays the interpolation in L^2 space, which performs pointwise averaging. **(c)** illustrates the interpolation in Wasserstein space, which captures smooth mass transport between distributions.

3 Model

This chapter introduces two types of INR models for the MRI reconstruction problem. The first model uses the forward operator for the prediction. The forward operator is a computationally intensive operation; therefore, we will also examine an approach that directly predicts the k-space value, bypassing the forward operator. The data consistency term for both methods is the same, namely $\frac{1}{2}||\cdot||_2^2$ as explained in section 2.1. Note that in [26, 29], they proposed a relative L2 loss due to the order of magnitude difference between the absolute values in k-space data; however, in [27], they did not find a significant performance difference.

3.1 Image INR

The image INR, denoted as F_θ, where θ are the trainable model parameters, directly learns the spatiotemporal image content. The learned image is then mapped to k-space using the forward operator described in Section 2.2.2. The network takes as input spatiotemporal coordinates $\mathbf{v} = (x, y, t) \in [-1, 1]^2 \times [0, T]$ and outputs complex-valued image intensities:

$$x_t(\mathbf{r}) \;=\; F_\theta(\underbrace{x, y}_{\mathbf{r}},\, t), \qquad \mathbf{r} = (x, y).$$

To simulate multi-coil acquisitions, the predicted image series is element-wise multiplied with coil sensitivity maps $S^i(\mathbf{r})$, yielding coil-specific spatial images for each time frame. These are then transformed to the frequency domain using a non-uniform fast Fourier transform (NUFFT). The NUFFT maps the spatial images onto a set of non-Cartesian k-space coordinates $\mathbf{v}_k = (k_x, k_y, t)$, where the same sampling pattern is assumed for each time step.

The NUFFT consists of a two-step process: first, a fast Fourier transform (FFT) is applied on a uniform grid; then, interpolation using a precomputed kernel is used to estimate the values at the non-Cartesian k-space coordinates.[55]

The complete training objective can be written as:

$$\theta = \arg\min_\theta \frac{1}{2} \left\| \mathcal{F}\mathcal{S}F_\theta(\mathbf{v}) - f^\delta \right\|_2^2 + R(F_\theta(\mathbf{v})) \tag{17}$$

where $\mathcal{S}$ applies coil sensitivities, $\mathcal{F}$ denotes the NUFFT operator, and f^δ are the measured noisy k-space samples and R is the regularization term.

3.2 k-space INR

The k-space implicit neural representation (INR), denoted as G_θ, directly learns to map spatiotemporal frequency coordinates to complex-valued k-space measurements. The trainable parameters θ are optimized such that the INR matches the acquired k-space data at the sampled locations. The input to the network consists of coordinates $\mathbf{v}_k = (k_x, k_y, t) \in [-1, 1]^2 \times [0, T]$, and the output corresponds to the predicted k-space value at that location.

This formulation eliminates the need for an explicit image-domain representation, instead operating directly in the frequency domain. In the multi-coil setting, the network can either take the coil index c as an additional input to the network: $G_\theta(k_x, k_y, t, c)$or have directly prediting all coil values together INR. Here, we opt for the latter approach, as it

reduces training time and leverages the fact that the coils overlap significantly in k-space, especially in the low-frequency regions.

The training objective minimizes the discrepancy between predicted and measured k-space data:

$$\theta = \arg\min_{\theta} \frac{1}{2} \left\| G_\theta(\mathbf{v}_k) - f^\delta(\mathbf{v}_k) \right\|_2^2 + R(G_\theta(\mathbf{v}_k)), \tag{18}$$

where $f^\delta(\mathbf{v}_k)$ denotes the measured noisy k-space samples at coordinates $\mathbf{v}_k$, and R is a regularization term that can encourage smoothness, temporal consistency, or sparsity in the learned k-space function.

3.3 Regularization

The optimal transport regularization is chosen to provide better geometric alignment with human visual perception compared to pixel-wise losses, such as L2.[56] As described in 2.5.2 equation 13, the Wasserstein distance can be interpreted as minimizing the kinetic energy between two distributions, helping guide the network to a meaningful physical reconstruction. A variation of OT-based losses is the concept of Wasserstein barycenters, which yield non-linear yet semantically coherent averages of probability distributions. The Wasserstein barycenters can be used as a template and compared to the model's prediction to learn plausible transitions.

To ensure numerical stability when using the Sinkhorn based approximation of the Wasserstein distance (W_2^ϵ), the reconstructed outputs are normalized to have equal total mass. The normalization can have unwanted effects. For example, if frame t contains a bright artifact not present in frame t-1, normalization would reduce the mass associated with shared structures to maintain a total mass equal to 1. This could potentially lead to incorrect interpolations or intensity changes in regions that should remain unchanged. To mitigate such effects, Unbalanced Optimal Transport is often preferred, as it allows partial mass matching and penalizes the creation or destruction of mass rather than forcing full normalization. In our case, we verified that the total intensity differences across frames remain small (less than 1.06%), and therefore, we initially applied standard normalization without introducing significant bias. To further improve robustness against rare bright artifacts and intensity fluctuations, we incorporate Optimal Transport only in the later stages of training once the base reconstruction has stabilized.

3.3.1 Wasserstein barycenter

For the calculation of the barycenter between predicted images, we choose two times of the original time points with one frame in between. The INR is used to query the model after the predictions made at t_0 and t_2 have been combined into a real-valued image in the spatial domain. The images are chosen to cast digital images as probability distributions through normalization, i.e., dividing each pixel value by the sum of all pixel values, to achieve a total sum (density) of 1. The formulation of the barycenter as described in 15 is used with the alpha parameter set to 0.5.

Image INR. For the image model, the three probability measures are obtained directly from the INR output:

$$\mu_0(x,y) = F_\theta(x,y,t), \qquad \mu_1(x,y) = F_\theta(x,y,t_1), \quad \mu_2(x,y) = F_\theta(x,y,t_2),$$

Each image is defined on a uniform spatial grid $(x,y) \in [-1,1]^2$, discretized into $m \times m$ points. The complex image values are normalized per frame by their global maximum magnitude. The image INR predicts complex-valued images, allowing for the potential incorporation of coil sensitivity maps and preserving phase information. Therefore, normalization is performed on the absolute values of the complex predictions to ensure consistent scaling across frames while retaining both magnitude and phase information.

For training, we randomly select an initial time point t, and construct a sequence of three consecutive frames $(t, t+1, t+2)$, sampled from the full temporal resolution. The normalization of the complex value image is done by:

$$\tilde{\mu}_i(x,y) = \frac{|\mu_i(x,y)|}{\sum_{j=1}^{m}\sum_{k=1}^{m}|\mu_i(x_j,y_k)|},$$

The Wasserstein barycenter is then given as

$$B(\tilde{\mu}_0,\tilde{\mu}_2) = \arg\min_{\mu\in P(\mathbb{R}^2)} \tfrac{1}{2}W_2^\varepsilon(\tilde{\mu}_1,\mu) + \tfrac{1}{2}W_2^\varepsilon(\tilde{\mu}_2,\mu)$$

Then the L2 loss is used to calculate the difference between the model prediction on t_1

$$R(F_\theta(x,y,t)) = \|\tilde{\mu}_1 - B(\hat{\mu}_0,\hat{\mu}_2)\|^2$$

K-space INR. The k-space model G_θ takes as input coordinates (k_x,k_y,t), where (k_x,k_y) form a uniform Cartesian grid over $[-1,1]^2$, corresponding to the Fourier domain resolution, and $t \in [-1,1]$. It predicts complex-valued coil data $G_\theta(k_x,k_y,t)$ for each coil $c \in \{1,\dots,C\}$. Although this Cartesian grid does not align with the actual sampling trajectory (e.g., radial sampling), it is used solely for the regularization loss because it enables the efficient computation of the inverse Fourier transform via the fast Fourier transform (FFT). This practical choice enables imposing spatial consistency through image-domain losses while training on arbitrary sampling patterns.

To obtain real-valued images for barycenter computation, the predicted k-space is first transformed to the image domain using an inverse FFT, followed by a sum-of-squares (SoS) operation across the coil dimension:

$$\hat{\mu}_i(x,y,t_i) = \sqrt{\sum_{c=1}^{C}|\mathcal{F}^{-1}G_{\theta,c}(k_x,k_y,t_i)|^2}. \quad i=1,2$$

These images are normalised as

$$\tilde{\mu}_i(x,y) = \frac{|\mu_i(x,y)|}{\sum_{j=1}^{m}\sum_{k=1}^{m}|\mu_i(x_j,y_k)|},$$

The Wasserstein barycenter is then given as

$$B(\tilde{\mu}_0, \tilde{\mu}_2) = \arg\min_{\mu \in P(\mathbb{R}^2)} \tfrac{1}{2} W_2^{\varepsilon}(\tilde{\mu}_0, \mu) + \tfrac{1}{2} W_2^{\varepsilon}(\tilde{\mu}_2, \mu)$$

Then the L2 loss is used to calculated the difference between the model prediction on t_1 ($\tilde{\mu_1}$)and the predicted barrycenter $B(\tilde{\mu}_0, \tilde{\mu}_2)$

$$R(G_\theta(x, y, t)) = \|\tilde{\mu}_1 - B(\hat{\mu}_0, \hat{\mu}_2)\|^2$$

3.3.2 Wasserstein distance

The Wasserstein distance is calculated between consecutive frames in the dynamic MRI sequence at the acquired time points. To evaluate the temporal consistency of the predicted image sequence, we compute the cumulative Wasserstein distance between consecutive normalized frames, we use the same notation used in the barycenter seciton:

$$R_{=} \sum_{i=0}^{m-1} \mathcal{W}_2^{\epsilon} \left(\tilde{\mu}_i, \tilde{\mu}_{i+1} \right),$$

where $\tilde{\mu}_i$ denotes the normalized image predicted by the INR model at time step t_i. Each $\tilde{\mu}_i$ is a discretized probability distribution over a spatial grid, as defined by:

$$\tilde{\mu}_i(x, y) = \frac{|\mu_i(x, y)|}{\sum_{j=1}^{m} \sum_{k=1}^{m} |\mu_i(x_j, y_k)|},$$

with $\mu_i(x, y) = F_\theta(x, y, t_i)$ being the raw complex-valued output of the model before normalization.

3.4 Inference

After training, we query the INR on a dense grid to render a high-resolution image matching the resolution of the ground truth. For the MRI dataset, due to the radial sampling pattern, no training data is available outside the unit disk. To avoid unreliable extrapolation, all coordinates for which $k_x^2 + k_y^2 > 1.0$ are considered outside the training domain and set to zero.

3.5 Implementation details model

The model is implemented with the PyTorch library (version 2.7.0). We used an SIREN model with periodic activations $\sin(\omega_0 \cdot)$. A total of three hidden layers are used, each containing 256 neurons; the final layer is linear and outputs the real and imaginary parts separately, following the approach of Sitzmann et al. [25], we set the $\omega_0 = 30$ and apply the recommended weight initialization as described in 2.4.2. The weights of the network are updated using the Adam optimizer [57] with a learning rate of 1×10^{-3} , other Adam hyperparameters are left at default values ($\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-7}$). The non-uniform Fast Fourier Transform (NUFFT) for MRI imaging is implemented based on the torchkbnufft package with the standard settings [55].

During the optimization process, all spatiotemporal coordinates were gathered in a single batch, with a batch size of 1. The input coordinates (x, y, t) were isotropically normalized to the range [0, 1] for faster convergence.

4 Results and experiments

In this section, we present the reconstruction results obtained with our implicit neural network (INR) model using two different optimal transport priors: the Wasserstein distance and the Wasserstein barycenter. These regularized INRs are compared to the IGRASP method described in Section 2.3, as well as to a baseline INR model without any regularization. A more detailed analysis and interpretation of the results are provided in the section 5.

Experiments are conducted on a synthetic and a 2D dynamic cardiac MRI dataset. For the synthetic datasets, the models are first trained using spatially undersampled k-space data, followed by temporal k-space undersampling, where no spatial undersampling is performed on the remaining time frame. The MRI dataset is spatially k-space undersampled and extended to temporal subsampling by skipping every other frame, combined with the previous spatial undersampling strategy. The reconstruction quality is quantitatively evaluated using the Structural Similarity Index Measure (SSIM) and Peak Signal-to-Noise Ratio (PSNR).

4.1 Evaluation metrics

The following two full-reference metrics are used to evaluate the reconstruction: Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM). Those metrics are widely considered the de facto standard for quantitative evaluation in computer vision and medical imaging tasks, including MRI reconstruction [58].

Although PSNR, and SSIM are commonly used for quantitative image quality evaluation, they do not always align with clinical utility or expert perception. Kastryulin et al. [58] show a moderate correlation between these metrics and radiologist assessments (correlation coefficients around 0.5). In particular, they emphasize that global metrics frequently overlook the diagnostic quality of specific anatomical regions.

Therefore, while higher PSNR or SSIM scores generally indicate better image quality, they should be interpreted cautiously. In many clinical applications, the visibility of task-relevant structures (e.g., vessel walls, lesions, or motion artifacts) is more important than global fidelity.[58]

4.1.1 Peak Signal to Noise Ratio (PSNR)

PSNR is defined by Mean Squared Error (MSE) and defines the ratio between the maximum possible power of the signal and the power of the distorting noise in decibels. The MSE gives the average squared difference between the reconstructed image $u(x, y)$ and the original image $\hat{u}(x, y)$. Let the image domain consist of M rows and N columns. Then, the MSE is defined as:

$$\mathrm{MSE}(\hat{u}, u) = \frac{1}{MN} \sum_{y=0}^{M-1} \sum_{x=0}^{N-1} [\hat{u}(x, y) - u(x, y)]^2 \tag{19}$$

then the PSNR is defined as:

$$\text{PSNR}(\hat{u}, u) = 10 \log_{10} \left(\frac{MAX_I^2}{\text{MSE}} \right)$$

where MAX_I is the maximum pixel value.

4.1.2 Structural Similarity Index Measure (SSIM)

The Structural Similarity Index Measure (SSIM) combines three distinct components. The first component, l, compares the brightness between two images, c evaluates the contrast between the dark and light regions, and s evaluates the pattern between the two images. The SSIM for the reconstructed image $\hat{u}$ and its ground truth u is given by: [59]

$$SSIM(\hat{u}, u) = [l(\hat{u}, u)]^\alpha \cdot [c(\hat{u}, u)]^\beta \cdot [s(\hat{u}, u)]^\gamma, \quad \alpha, \beta, \gamma > 0$$

Where $l(\hat{u}, u), c(\hat{u}, u), (\hat{u}, u)$ are given by:

$$\begin{aligned}
l(\hat{u}, u) &= \frac{2\mu_{\hat{u}}\mu_u + C_1}{\mu_{\hat{u}}^2 + \mu_u^2 + C_1}, & C_1 &= (k_1 L)^2 \\
c(\hat{u}, u) &= \frac{2\sigma_{\hat{u}}\sigma_u + C_1}{\sigma_{\hat{u}}^2 + \sigma_u^2 + C_1}, & C_2 &= (k_2 L)^2 \\
s(\hat{u}, u) &= \frac{2\sigma_{\hat{u}u} + C_3}{\sigma_{\hat{u}}\sigma_u + C_3}, & C_3 &= (k_3 L)^2
\end{aligned}$$

Here, $\mu_{\hat{u}}$ and μ_u represent the average intensity values of the predicted and reference images, respectively. The terms $\sigma_{\hat{u}}$ and σ_u denote the standard deviations of the intensity. The term $\sigma_{\hat{u}u}$ is the cross-correlation coefficient.

The constants C_i for $i = 1, 2, 3$ are introduced for numerical stability, preventing division by small values where $k_i << 1$ and L is the dynamic range of the pixel values (255 for 8-bit grayscale images).

For the SSIM metric, higher values indicate better image quality. The following properties hold, ensuring that the order of inputs is irrelevant, the maximum value is 1, and there exists a unique maximum:

1. **Symmetry**: $SSIM(\hat{y}, y) = SSIM(y, \hat{y})$
2. **Boundedness**: $SSIM(\hat{y}, y) \leq 1$
3. **Unique Maximum**: $SSIM(\hat{y}, y) = 1 \iff \hat{y} = y$

The settings used in the thesis are $\alpha = \beta = \ gamma = 1$ and $C_3 = \frac{C_2}{2}$, which are the default settings when comparing images. [59]

4.2 Synthetic data

This section presents the numerical results obtained using a synthetic dataset. The synthetic data consists of a sequence of 10 small Gaussian images of size 32×32, where the Gaussian blob gradually moves from left to right while increasing and decreasing in size, see figure 6. This mimics simple dynamic motion over time. To simulate MRI-like measurements, each frame is transformed to the frequency domain using a Fast Fourier Transform (FFT).

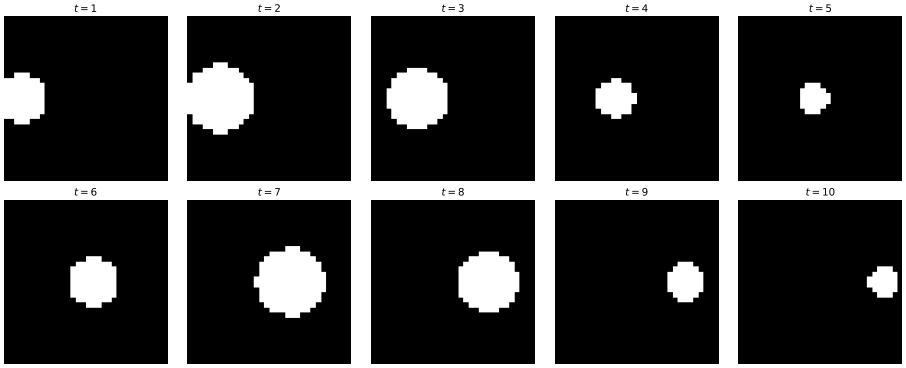


Figure 6: The full dynamic sequence consists of 10 time frames of size 32×32 pixels.

Each model is trained for a total of 5,000 iterations. The optimal transport (OT) regularization is only applied during the final 500 iterations. Since the model's early predictions are dominated by noise, applying optimal transport regularization too early causes the model to align artifacts instead of meaningful structures. For the regularization strength (λ), both regularization approaches are set to 1.

We first investigate how the use of optimal transport regularization affects reconstruction quality under standard k-space spatial subsampling. For each time frame, a subset of the k_y lines is used for the training, also known as Cartesian sampling. A fixed number of central lines, referred to as the calibration region, is retained across all frames to ensure stability and simulate the standard sampling procedure in MRI. The central lines are often used for estimating sensitivity maps. The remaining lines are selected randomly for each frame, resulting in different sampling patterns over time. This temporal variability increases the incoherence of the sampling scheme, which is known to improve reconstruction in compressed sensing and deep learning frameworks. Figure 7 shows the sampling mask of an individual frame and the distribution of sampling patterns across time. Additionally, the inverse Fourier transform of the undersampled k-space data, with missing samples zero-filled, is used as a baseline for comparison.

In the second experiment, we introduce temporal subsampling, where only every other time frame is used during training, creating reduced temporal coherence in the acquisition process. This scenario enables us to assess the model's ability to interpolate intermediate frames and determine whether OT regularization enhances temporal consistency in the reconstructed sequence.

4.2.1 Barycentric OT regularization results

k-space subsampling Figure 8 shows the reconstruction results under spatial subsampling for both the k-space and image INR models, with and without barycenter optimal

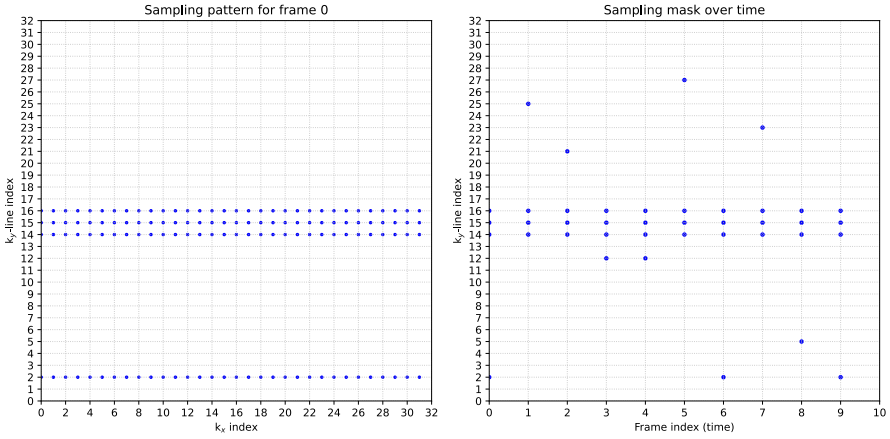


Figure 7: Sampling pattern used for spatial k-space undersampling. **Left:** Sampling mask for a single frame (frame 1), where blue dots indicate the sampled k_y-lines in the frame. The central calibration region (central band) is fully sampled and preserved identically across all frames. The remaining outer k_y-lines are randomly and uniformly selected per frame, resulting in a time-varying sampling pattern. **Right:** Visualization of the temporal variation in the sampling masks across all time frames, illustrating how the outer lines change randomly over time while the central region remains constant.

transport (OT) regularization. The barycenter is computed for every uneven index and compared to the model's prediction at those time steps. The k-space INR (row 2) shows noticeable noise, both in the corners and inside the Gaussian blob. By adding the OT loss (row 3), we observe not only an apparent reduction in noise at the predicted barycenter frames ($t = 1, 3, 5, \dots$) but also an improvement across other frames. The blob boundaries remain smooth, and the noise pattern better matches the expected structure at every location. The image INR already produces well-defined shapes and accurate intensity within the Gaussian blob. As a result, applying OT regularization yields little to no visible improvement in the reconstruction.

Temporal k-space subsampling Figure 9 presents the results comparing the unregulated networks with those trained using the Wasserstein barycenter optimal transport regularization. Without OT (row 2 k-space INR, row 4 image INR), every observed frame is reproduced almost perfectly, yet the interpolated frames give only random noise. This behavior is expected: in the absence of any explicit temporal constraint, the INR is free to assign arbitrary values at time-points for which no k-space measurements exist, and the optimizer drifts toward high-frequency artifacts that do not affect the MSE on the sampled frames. Once OT is introduced (rows 3 and 5), the global shape and motion trajectory are recovered. However, a noticeable amount of residual noise remains, particularly in the reconstructions from the direct k-space fitting (row 3), both inside the object and in the background. Interestingly, the forward model approach (row 5) exhibits far fewer of these artifacts, indicating that the direct k-space formulation struggles more with denoising and temporal consistency, even when guided by OT.

In Figure 10, every third frame is retained, and we observe similar reconstruction behavior for the missing frames across both models as described previously when every second frame was maintained. One notable difference, however, is that the model now consistently underestimates the size of the predicted shapes.

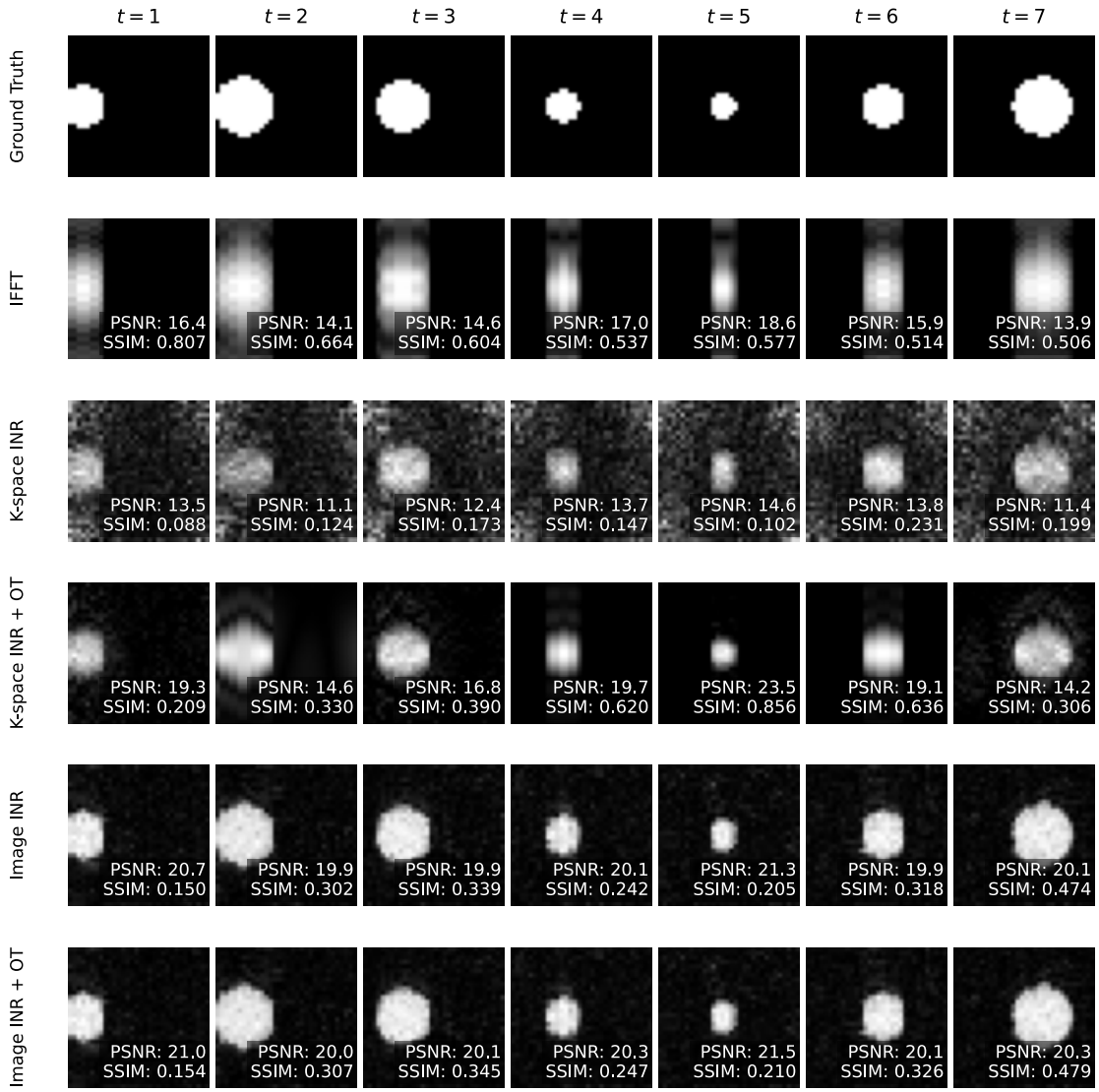


Figure 8: **K-space undersampled data reconstruction for barycenter regularization** Visual comparison of reconstruction results on synthetic data. The top row displays the ground-truth images. The second row shows the zero-filled inverse FFT reconstruction from the undersampled k-space data. The third row shows the output of the k-space INR model without Optimal Transport (OT) regularization, and the fourth row shows the k-space INR with OT regularization. The fifth row shows the image-domain INR without OT, and the sixth row shows the image-domain INR with OT regularization. Each column corresponds to a different time frame.

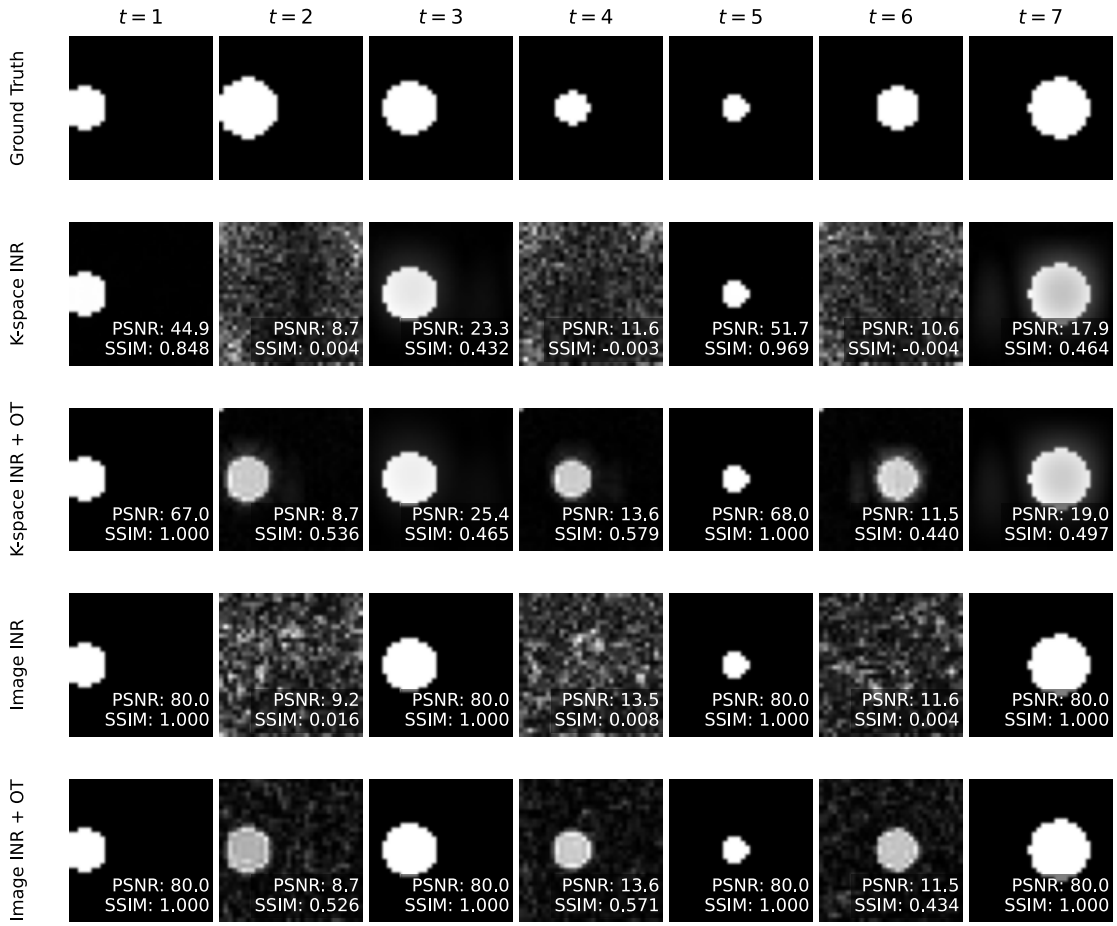


Figure 9: **Temporal undersampled reconstruction for barycenter regularization** visual comparison of reconstruction results on synthetic data. The top row shows the ground truth images, the middle row shows the output from the INR model without regularization, and the bottom row shows the INR model with Optimal Transport (OT) regularization. Each column represents a different time step, where every third image (1, 4, 7, 10) is used for training.

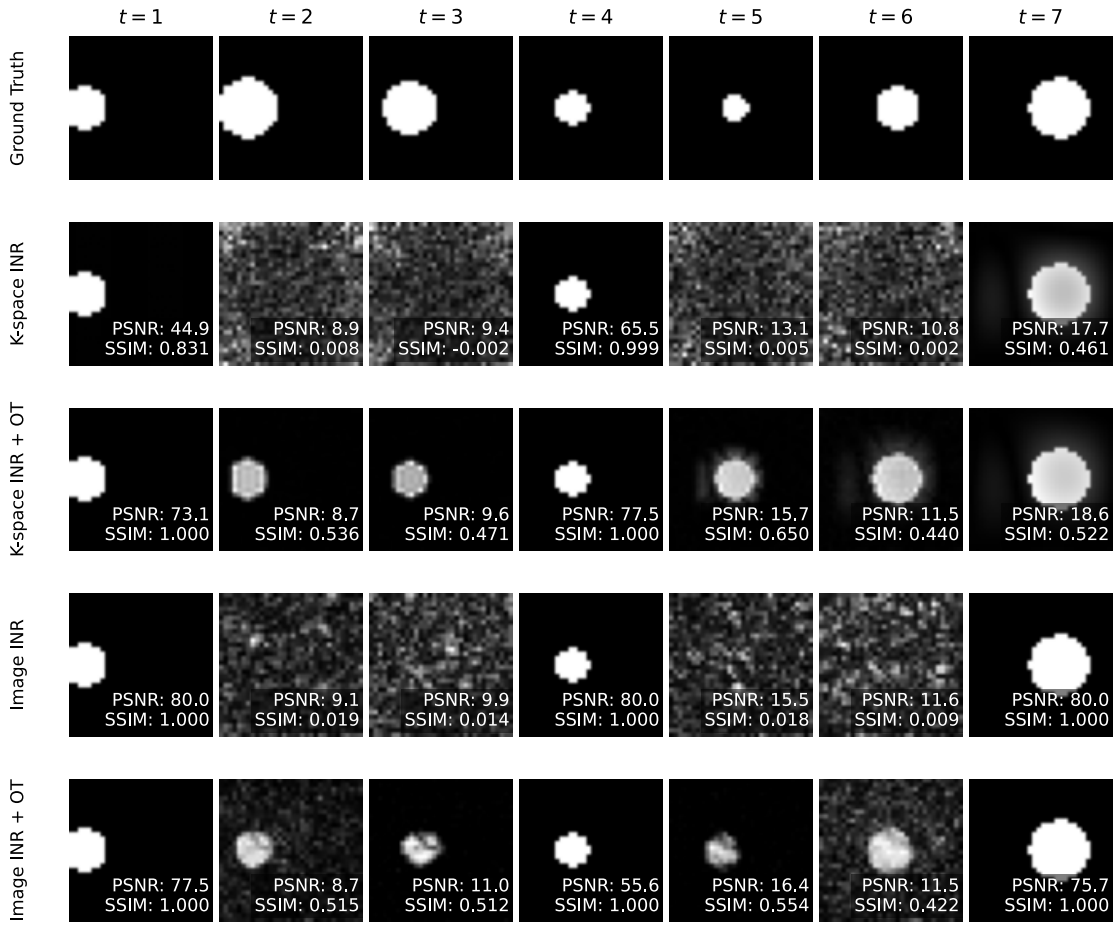


Figure 10: **Temporal undersampled reconstruction for barycenter regularization** visual comparison of reconstruction results on synthetic data. The top row shows the ground truth images, the middle row shows the output from the INR model without regularization, and the bottom row shows the INR model with Optimal Transport (OT) regularization. Each column represents a different time step, where every third image (1, 4, 7, 10) is used for training.

4.2.2 Wasserstein OT regularization results

k-space subsampling Figure 11 shows the results of k-space en image INR, with and without Wasserstein regularization. For k-space, INR showed the Wasserstein distance; however, the noise present in the corners was removed. Nevertheless, some noise remains above and below the object. The shape reconstruction for the step-wise borders is not recovered, and the intensity of the object is only slightly improved for $2, 4, 5, 6$. The image INR shows no visible effect with OT regularization, where the sharp borders and intensity of the object remain unchanged, and no extra noise is added.

Temporal k-space subsampling Figure 12 shows the summed Wasserstein distance calculated between every pair of consecutive frames. When the model is trained without any regularization, both the k-space and image domain INR models (rows 2 and 4) predict noise for the missing time frames. For the sampled time frames, the predicted shapes are correct, although the k-space INR exhibits a consistent intensity offset.

The k-space INR with Wasserstein regularization reconstructs the missing Gaussian blob (at uneven indices) with smooth borders and a slight, spatially varying intensity mismatch noticeable in the center. However, the Wasserstein loss does not significantly affect the reconstruction of previously sampled frames, and some intensity mismatch remains between frames 3 and 5.

The image INR with Wasserstein regularization shows similar behavior. The guided reconstructions at the missing time steps (uneven indices) appear slightly distorted, with more pronounced flattening of intensities across different regions. Additionally, the reconstruction quality for sampled frames (e.g., frame 7) shows slight color intensity shifts, resulting in a minor drop in PSNR.

In Figure 13, every third frame is retained, and we observe similar reconstruction behavior for the missing frames across both models as described previously when every second frame was maintained. One notable difference, however, is that the model now consistently underestimates the size of the predicted shapes.

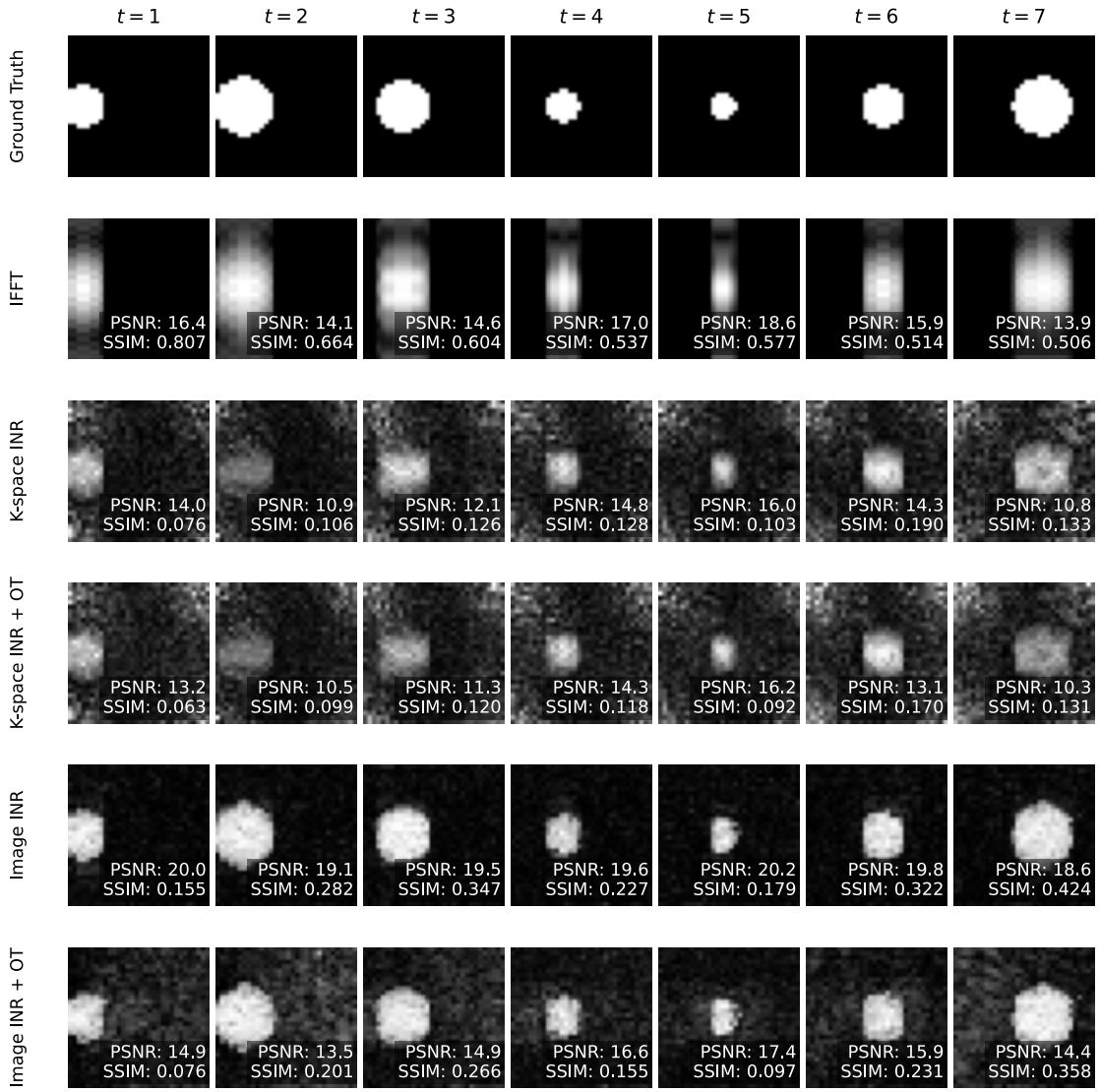


Figure 11: **K-space undersampled reconstruction for Wasserstein regularization** Visual comparison of reconstruction results on synthetic data. The top row displays the ground-truth images. The second row shows the zero-filled inverse FFT reconstruction from the undersampled k-space data. The third row shows the output of the k-space INR model without (OT) regularization, and the fourth row shows the k-space INR with OT regularization. The fifth row shows the image-domain INR without OT, and the sixth row shows the image-domain INR with OT regularization. Each column corresponds to a different time frame.

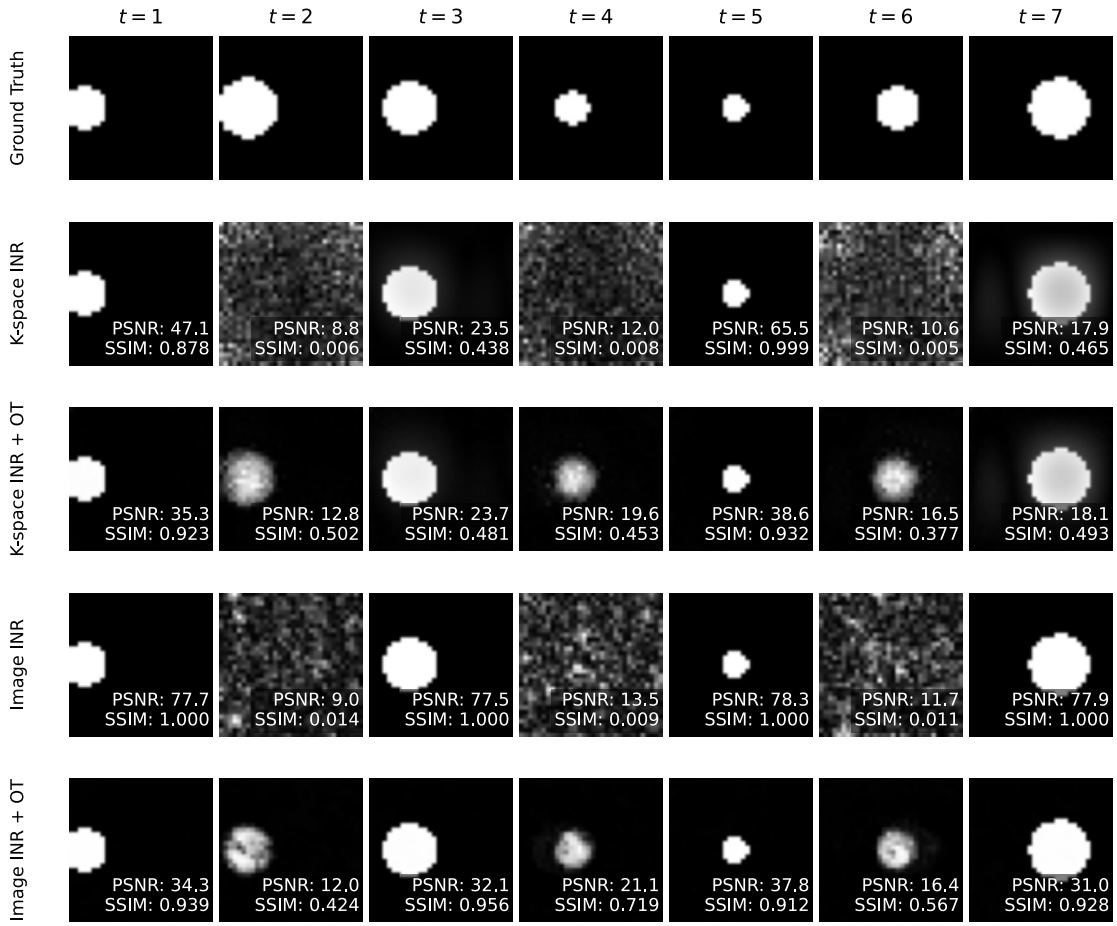


Figure 12: **Temporal undersampled reconstruction for Wasserstein regularization** Comparison of reconstructed dynamic MRI frames with different INR models. Each column shows a single representative frame. **Top row:** Ground truth reconstruction. **Second row:** k-space INR without Wasserstein regularization. **Third row:** k-space INR with Wasserstein regularization as a temporal prior. **Fourth row:** image-domain INR without Wasserstein regularization. **Fifth row:** image-domain INR with Wasserstein regularization. **Note:** Even-numbered frames (e.g., frames 2, 4, 6) were omitted during training and are only used for evaluation, highlighting the models' ability to interpolate missing temporal information.

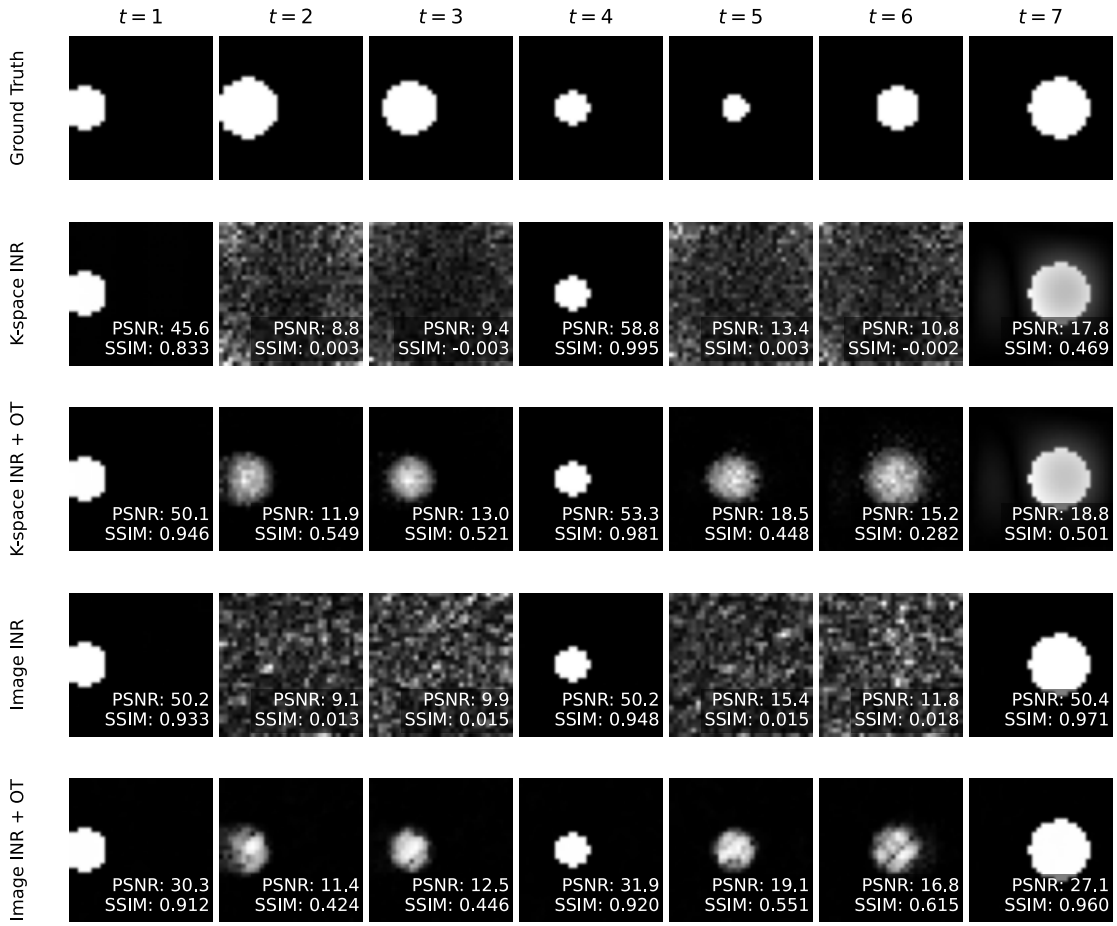


Figure 13: **Temporal undersampled reconstruction for Wasserstein regularization** Comparison of reconstructed dynamic frames with different INR models. Each column shows a single representative frame. **Top row:** Ground truth reconstruction. **Second row:** k-space INR without Wasserstein regularization. **Third row:** k-space INR with Wasserstein regularization as a temporal prior. **Fourth row:** image-domain INR without Wasserstein regularization. **Fifth row:** image-domain INR with Wasserstein regularization. **Note:** Frames 2, 3, 5, and 6 were omitted during training and are only used for evaluation, highlighting the models' ability to interpolate missing temporal information in unseen frames.

4.3 MRI data

For the final MRI reconstruction experiments, we focused exclusively on the k-space INR model with barycenter regularization, as it consistently showed the best performance in both synthetic and real data experiments. We used 10,000 iterations for the k-space spatial undersampling experiments and 20,000 iterations for the k-space spatiotemporal undersampling experiments. Optimal Transport regularization was activated during the final 2,000 and 5,000 iterations of these experiments, respectively.

4.3.1 Dataset

We use a publicly available radial cardiac cine MRI dataset released via the Harvard Dataverse [60]. The dataset comprises breath-held, retrospectively ECG-triggered, 2D radial bSSFP acquisitions from 108 subjects (101 patients and 7 healthy controls), acquired on a 3T MAGNETOM Vida system using body and spine phased-array coils. Imaging was performed in a mid-ventricular slice with the following parameters: TR/TE = 3.06/1.4 ms, flip angle = 48°, FOV = 380 × 380 mm^2, matrix size = 208 × 208, and slice thickness = 8 mm. On average, 196 radial spokes were acquired per cardiac phase across 25 ECG-binned time frames. Zero-padding was removed to avoid introducing implicit priors. [61].In Appendix A, the individual coil images and the corresponding k-space values are visualized.

The 196 radial spokes are uniformly distributed across frames, with consistent angular positions reused. All receiver coils collect data simultaneously and have the same coordiates. We implement spatial under-sampling by selecting every R-th spoke, resulting in a uniform distribution. A one-spoke offset is applied for each subsequent frame to enhance temporal diversity. Figure 14 displays the overall trajectory (left), showing the original spokes in light gray and the black spokes indicating the $R = 10$ undersampling. On the right, the sampling patterns for the first three frames illustrate the rotational progression over time.

The above use of the data differs from the recent INR-based approaches on radial cine data [62, 31, 30]. In those studies, fully sampled scanner data is first transformed into the image domain and then projected into k-space using a non-uniform Fourier transform (NUFFT) to simulate golden-angle undersampling. The same forward operator is later used in the training process, introducing bias in the results, a practice commonly referred to as a data crime [61]. In addition, the simulated data contains less noise and the exact location of the frequency measurement, whereas the original data can have a slight offset.

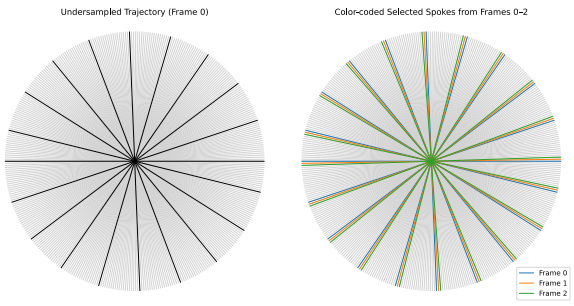


Figure 14: Visualization of the radial k-space sampling patterns. **Left:** Undersampling pattern for frame 0; selected spokes in black, unselected in gray. **Right:** Overlay of selected spokes for frames 0–2, each in a different color.

4.3.2 Spatial k-space subsampling results

Figure 15 display the reconstruction of spatially undersampled k-space data. In the figure, it becomes evident that both the IGRASP reconstruction and the INR-based method tend to smooth out fine details compared to the reference. However, while the IGRASP method shows a more balanced intensity distribution across the image, the INR reconstruction exhibits a noticeable bright cross in the center that was not present in the synthetic dataset, making this a plotting error. Despite these differences, quantitative metrics such as SSIM and PSNR indicate that both methods produce reconstructions of comparable quality. The INR with barycenter-based reconstruction yields an average PSNR in the image series of 21.89 ± 0.07 and an SSIM of 0.78 ± 0.03. compared to GRASP with a PSNR of 19.04 ± 0.06 and an SSIM of 19.05 ± 0.05.

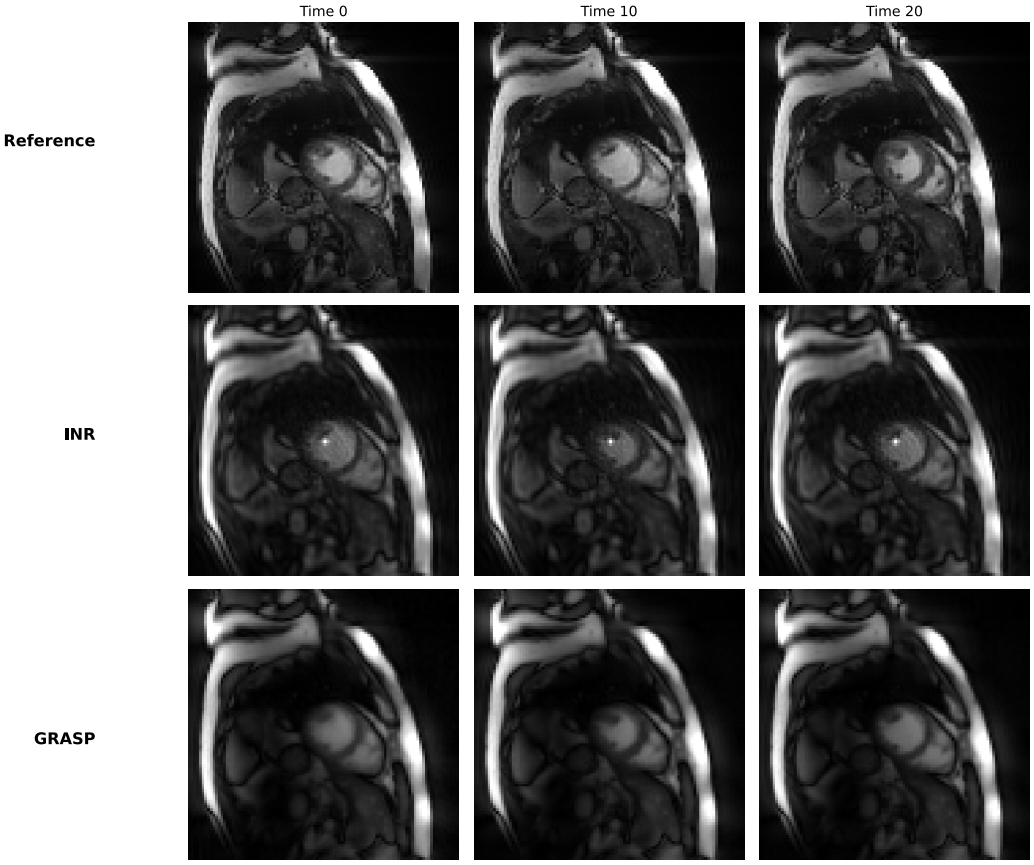


Figure 15: **Spatially undersampled MRI reconstruction.** Reconstruction results on cardiac MRI data with spatial undersampling, where every 5th spoke in k-space is selected. The top row shows the fully sampled reference images. The second row shows the reconstruction from the implicit neural representation (INR) model with barycenter regularization. The third row shows the reconstruction using the iGRASP method.

4.3.3 Spatialtemporal k-space subsampling results

In previous figures, we displayed the first, middle, and last frames to illustrate the full dynamic range of the cardiac motion. In Figure 16, however, we show the first three consecutive frames to specifically highlight the effect of temporal undersampling, where every other frame is skipped, in combination with spatial undersampling by selecting only every fifth spoke. This results in using only approximately 10% of the fully sampled data, allowing us to demonstrate the impact of missing frames during a particular phase of the cardiac cycle. Without any temporal regularization, the INR model produces only a faint outline of the heart and struggles to generalize across the missing frames. In contrast, adding regularization helps the network produce more meaningful intermediate representations.

The INR model barycenter-based regularization achieves similar performance to the reference IGRASP method on the retained frames, suggesting that both methods effectively reconstruct the static structures. However, for the missing frames, the barycenter-based regularization provides additional guidance that improves the temporal consistency of the reconstruction. In this example, the central dynamic region is the heart chamber, while the rest of the anatomy remains relatively static. Notably, the INR reconstruction still exhibits a considerable amount of noise in the region where motion is present, indicating that the model struggles to capture the fine-grained temporal dynamics despite the added regularization.

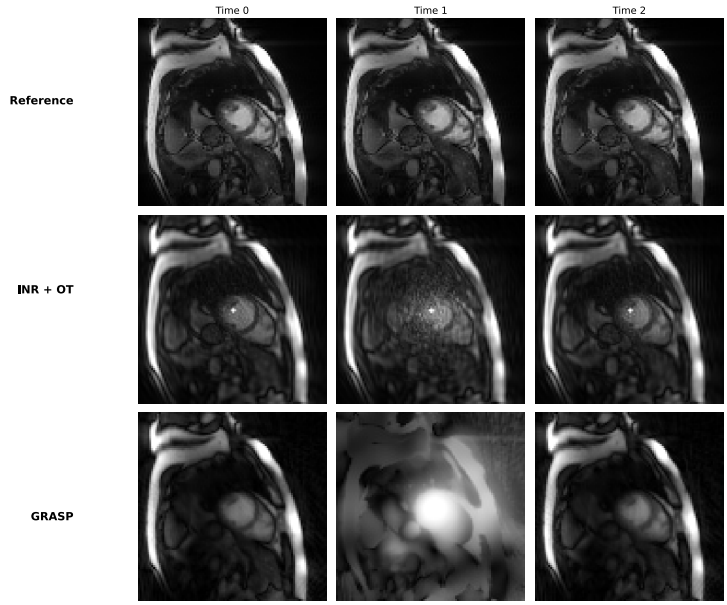


Figure 16: **spatial-temporal k-space undersampling** Reconstruction results on cardiac MRI data where both spatial and temporal undersampling were applied: every 5th spoke was selected in the spatial domain, and only every other frame was sampled in time. The top row shows the fully sampled reference images. The second row shows the reconstruction from the implicit neural representation (INR) model with barycenter regularization. The third row shows the reconstruction using the iGRASP method.

5 Discussion

5.1 Computation time

The current INR training time is approximately 15 minutes, with an additional 1 hour when including optimal transport regularization—covering both Wasserstein distance and barycenter calculations—for a single slice. In comparison, even reconstruction speeds of 10 to 15 minutes per slice already result in several hours of processing time for a full MRI exam. Meanwhile, the latest IGRASP implementation, as cited in [63], achieves reconstruction times of approximately one minute per slice. Similar INR-based approaches require anywhere between 15 minutes to several hours per slice, depending on hardware configuration, image resolution, number of coils, and temporal range [27, 26, 30].

Possible strategies to reduce training time include the use of fully fused MLPs, which have demonstrated a speed-up of up to 2× or 3× in classical NeRF applications [64]. A potential drawback of this approach is its limited network depth and width, resulting from high memory consumption. Additionally, reducing the number of coils or applying coil compression can significantly reduce training time. Finally, transfer learning may offer further gains by pretraining on a reference image and fine-tuning on new frames or subjects,[65] shows a created performance gain in the natural image domain.

5.2 Barycenter vs. Wasserstein distance regularization performance

In our experiments, we observe that pairwise Wasserstein distance regularization applied to synthetic data yields realistic interpolations for the missing frames. However, when applied to the MRI data set, it does not necessarily follow the anatomical motion we aim to capture. When the temporal gap between sampled frames is large, skipping a frame causes the model to learn a smooth, uniform distribution between the frames rather than the predicted noise without regularization. The new smooth interpolation without any further priors gets the INR stuck. In contrast, the barycenter interpolation explicitly forces each missing frame toward the Wasserstein barycenter, where significant deviations are penalized by the l2 norm. This additional constraint sharpens the reconstructed dynamics at the skipped time points. Nevertheless, it is achieved at the expense of slightly reduced spatial detail in the sampled images.

5.3 Image vs k-space INR performance synthetic data

Although image and k-space INR both use the same model and are trained with and without optimal transport regularization, we observe a consistent performance gap favoring the image-space approach. A possible explanation for this lies in the structure of the k-space data. Unlike image-space representations, where spatial features are localized, k-space encodes global frequency information in a highly interdependent manner. This makes learning in the k-space domain significantly more sensitive to local variations, especially in high-frequency components. While the optimal transport regularization aims to enforce temporal consistency, it does not sufficiently constrain the correlations among k-space coefficients. As a result, the k-space INR tends to overfit to the observed measurements without learning a smooth or physically consistent latent representation. The loss surface for the k-space model remains highly irregular, preventing convergence to a meaningful reconstruction.

In contrast, the image-space INR benefits from the use of the Fast Fourier Transform

(FFT) during training to simulate measurements. This introduces a form of implicit regularization: the model is not directly learning unconstrained frequency components, but is instead constrained to produce spatial structures that, when transformed via FFT, yield realistic k-space data. This indirect constraint stabilizes the learning dynamics, helping the model better capture high-frequency variations, such as edges and delicate anatomical structures.

It is also important to note that the same forward operator is used both to simulate the k-space measurements and used in the image INR. This introduces a form of inverse crime [61], potentially biasing the reconstructions. As highlighted by [66], such setups where the forward and inverse models are perfectly matched can suppress artifacts and hide instability, leading to results that may not fully reflect performance under real-world conditions.

5.4 K-space INR vs IGRASP spatial-temporal undersampling performance

In our experiments, we observed that the INR model with Optimal Transport (OT) regularization outperformed the iGRASP method under combined spatial-temporal undersampling. This can be attributed to the flexibility of the INR approach, which allows arbitrary sampling in k-space and combined with Wasserstein barycenter to interpolate missing temporal frames. In contrast, iGRASP is an iterative method that makes predictions based on sampled data locations and has only sensitivity maps available for estimating the missing frames.

For spatial undersampling, the motion between frames in our dataset is relatively small, and the cardiac motion is periodic and consistent over time. This stability makes our dataset particularly suitable for total variation regularization, as the regularity of the motion allows us to utilize TV-based sparsity constraints effectively.[67] Under these conditions, both iGRASP and the INR model with OT regularization can reconstruct the undersampled k-space data with similar accuracy. Since temporal interpolation plays a minor role, both methods can effectively leverage coil sensitivity information to fill in the missing spatial data.

6 Conclusion

In this thesis, we investigate how optimal transport can be leveraged to regularize implicit neural representations (INRs) in the context of dynamic MRI reconstruction under both spatial and spatio-temporal undersampling regimes. To this end, we explored two types of regularization: one based on the total Wasserstein distance between consecutive frames, and one based on the L2 distance to a Wasserstein barycenter of the predictions of the INR model. These regularizations are applied to two types of INR models: one that directly predicts k-space, and one that includes a forward operator.

For k-space INR, barycenter regularization worked best: in spatial undersampling, it reduced some noise around the synthetic Gaussian blob, and in temporal undersampling, it enabled good interpolation of the shape, albeit with a global intensity offset. Additionally, we demonstrated that the model with missing time frames yields noisy interpolated images.

For image INR, Wasserstein distance introduced noise in the spatial undersampling case, even though the unregularized INR already provided good reconstructions. In temporal un-

dersampling, the Wasserstein distance alone produced noise within the object and smaller interpolations of the shape, while barycenter regularization failed to recover a faithful reconstruction.

6.1 Future work

Future work could further explore barycenter-based interpolation for dynamic MRI reconstruction by looking at unprocessed k-space data. The current approach used discretely processed time frames. If we use the raw data, it could enable the barycenter formulation to exploit temporal continuity more effectively, as larger variations in the reconstructed image domain are observed.

So far, we have assumed that each frame contains an equal amount of measurement data. However, it would be interesting to investigate how models handle variations in the amount of data per frame, as, in practice, the data volume can differ between frames in dynamic acquisitions. [68]

We observed that while optimal transport enforces temporal consistency, it does not sufficiently promote spatial sharpness or contrast within each frame. Therefore, combining OT-based temporal regularization with classical spatial priors, such as total variation or wavelet sparsity, could further enhance the spatial reconstruction quality.

In future work, we will consider several strategies to reduce training time and improve reconstruction speed. These include the use of fused MLP architectures for faster inference [64], transfer learning tailored to INR's [65], and reducing the number of coils or applying coil compression to lower the dimensionality of the input space.

References

[1] Robert J. Young and Edmond A. Knopp. "Brain MRI: Tumor Evaluation". In: *Journal of magnetic resonance imaging: JMRI* 24.4 (Oct. 2006), pp. 709–724. DOI: 10.1002/jmri.20704.

[2] M. W. Logue et al. "MRI-measured Atrophy and Its Relationship to Cognitive Functioning in Vascular Dementia and Alzheimer's Disease Patients". In: *Alzheimer's and dementia* 7.5 (Sept. 2011), pp. 493–500. DOI: 10.1016/j.jalz.2011.01.004.

[3] N. C. Fox and P. A. Freeborough. "Brain Atrophy Progression Measured from Registered Serial MRI: Validation and Application to Alzheimer's Disease". In: *JMRI* 7.6 (1997), pp. 1069–1075. ISSN: 1053-1807. DOI: 10.1002/jmri.1880070620.

[4] Tristan van Leeuwen. *Lecture notes, Inverse Problems and Imaging*. 2021.

[5] Roland N. Boubela et al. "Scanning Fast and Slow: Current Limitations of 3 Tesla Functional MRI and Future Potential". In: *Frontiers in Physics* 2 (2014). DOI: 10.3389/fphy.2014.00001.

[6] Giovanni Foti and Chiara Longo. "Deep learning and AI in reducing magnetic resonance imaging scanning time: advantages and pitfalls in clinical practice". In: *Polish Journal of Radiology* 89 (2024), pp. 443–451. DOI: 10.5114/pjr/192822.

[7] Maxim Zaitsev, Julian Maclaren, and Michael Herbst. "Motion Artifacts in MRI: A Complex Problem with Many Partial Solutions". In: *Journal of Magnetic Resonance Imaging* 42.4 (2015), pp. 887–901. DOI: 10.1002/jmri.24850.

[8] J. W. Carlson and T. Minemura. "Imaging Time Reduction through Multiple Receiver Coil Data Acquisition and Image Reconstruction". In: *Magnetic Resonance in Medicine* 29.5 (1993), pp. 681–687. DOI: 10.1002/mrm.1910290516.

[9] Masaya Takahashi, Hidemasa Uematsu, and Hiroto Hatabu. "MR Imaging at High Magnetic Fields". In: *European Journal of Radiology* 46.1 (Apr. 2003), pp. 45–52. DOI: 10.1016/S0720-048X(02)00331-5.

[10] Natalia Gudino and Sebastian Littin. "Advancements in Gradient System Performance for Clinical and Research MRI". In: *Journal of Magnetic Resonance Imaging* 57.1 (2023), pp. 57–70. DOI: 10.1002/jmri.28421.

[11] Jong Chul Ye. "Compressed Sensing MRI: A Review from Signal Processing Perspective". In: *BMC Biomedical Engineering* 1.1 (Mar. 2019), p. 8. DOI: 10.1186/s42490-019-0006-z.

[12] Oren N Jaspan, Roman Fleysher, and Michael L Lipton. "Compressed Sensing MRI: A Review of the Clinical Literature". In: *The British Journal of Radiology* 88.1056 (Dec. 2015), p. 20150487. DOI: 10.1259/bjr.20150487.

[13] Florian Knoll et al. "Second Order Total Generalized Variation TGV for MRI". In: *Magnetic Resonance in Medicine* 65.2 (2011), pp. 480–491. ISSN: 1522-2594. DOI: 10.1002/mrm.22595.

[14] Qu Xiaobo et al. "Iterative Thresholding Compressed Sensing MRI Based on Contourlet Transform". In: *Inverse Problems in Science and Engineering* 18.6 (Jan. 2010), pp. 737–758. DOI: 10.1080/17415977.2010.492509.

[15] Li Feng et al. "Highly Accelerated Real-Time Cardiac Cine MRI Using k-t SPARSE-SENSE". In: *Magnetic Resonance in Medicine* 70.1 (July 2013), pp. 64–74. DOI: 10.1002/mrm.24440.

[16] Hong Jung et al. "K-t FOCUSS: A General Compressed Sensing Framework for High Resolution Dynamic MRI". In: *Magnetic Resonance in Medicine* 61.1 (Jan. 2009), pp. 103–116. DOI: 10.1002/mrm.21757.

[17] Christoph Brune. "4D Imaging in Tomography and Optical Nanoscopy". Ph.D. dissertation. Münster, Germany: University of Münster, July 2010.

[18] Johan Karlsson and Axel Ringh. *Generalized Sinkhorn Iterations for Regularizing Inverse Problems Using Optimal Mass Transport*. Dec. 2016. DOI: 10.1137/17M111208X.

[19] Yiming Gao. *A Wasserstein distance and total variation regularized model to image reconstruction problems*. 2023. eprint: 2303.07713.

[20] Gushan Zeng et al. "A Review on Deep Learning MRI Reconstruction without Fully Sampled K-Space". In: *BMC Medical Imaging* 21.1 (Dec. 2021), p. 195. DOI: 10.1186/s12880-021-00727-9.

[21] Reinhard Heckel et al. "Deep Learning for Accelerated and Robust MRI Reconstruction: A Review". In: arXiv:2404.15692 (Apr. 2024). DOI: 10.48550/arXiv.2404.15692. eprint: 2404.15692.

[22] Md. Biddut Hossain et al. "A Systematic Review and Identification of the Challenges of Deep Learning Techniques for Undersampled Magnetic Resonance Image Reconstruction". In: *Sensors (Basel, Switzerland)* 24.3 (Jan. 2024), p. 753. DOI: 10.3390/s24030753.

[23] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. "Deep Image Prior". In: *International Journal of Computer Vision* 128.7 (Mar. 2020), pp. 1867–1888. ISSN: 1573-1405. DOI: 10.1007/s11263-020-01303-4.

[24] Jaejun Yoo et al. *Time-Dependent Deep Image Prior for Dynamic MRI*. 2021. eprint: 1910.01684.

[25] Vincent Sitzmann et al. "Implicit Neural Representations with Periodic Activation Functions". In: *Advances in Neural Information Processing Systems* 2020-December (June 2020). ISSN: 10495258. URL: https://arxiv.org/abs/2006.09661v1.

[26] Wenqi Huang et al. "Neural Implicit k-Space for Binning-Free Non-Cartesian Cardiac MR Imaging". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 13939 (Jan. 2023), pp. 548–560. DOI: 10.1007/978-3-031-34048-2_42.

[27] Johannes F. Kunz, Stefan Ruschke, and Reinhard Heckel. "Implicit Neural Networks with Fourier-Feature Inputs for Free-Breathing Cardiac MRI Reconstruction". In: *IEEE Transactions on Computational Imaging* 10 (Jan. 2024), pp. 1280–1289. ISSN: 23339403. DOI: 10.1109/TCI.2024.3452008.

[28] Veronika Spieker et al. "ICoNIK: Generating Respiratory-Resolved Abdominal MR Reconstructions Using Neural Implicit Representations in k-Space". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 14533 (Jan. 2024), pp. 183–192. DOI: 10.1007/978-3-031-53767-7_18.

[29] Veronika Spieker et al. "Self-supervised k-Space Regularization for Motion-Resolved Abdominal MRI Using Neural Implicit k-Space Representations". In: *Sprinerlink* 15007 (Jan. 2024), pp. 614–624. ISSN: 16113349. DOI: 10.1007/978-3-031-72104-5_59. URL: http://arxiv.org/abs/2404.08350.

[30] Jie Feng et al. "Spatiotemporal implicit neural representation for unsupervised dynamic MRI reconstruction". In: *IEEE Transactions on Medical Imaging* (Dec. 2022). ISSN: 1558254X. DOI: 10.1109/TMI.2025.3526452. URL: http://arxiv.org/abs/2301.00127.

[31] Tabita Catalán et al. "Unsupervised reconstruction of accelerated cardiac cine MRI using neural fields". In: *Computers in Biology and Medicine* 185 (Feb. 2025), p. 109467. ISSN: 00104825. DOI: 10.1016/j.compbiomed.2024.109467.

[32] Hua Chieh Shao et al. "3D cine-magnetic resonance imaging using spatial and temporal implicit neural representation learning (STINR-MR)". In: *Physics in Medicine and Biology* 69 (9 Apr. 2024), p. 095007. ISSN: 0031-9155. DOI: 10.1088/1361-6560/AD33B7.

[33] Liyue Shen, John Pauly, and Lei Xing. "NeRP: Implicit Neural Representation Learning With Prior Embedding for Sparsely Sampled Image Reconstruction". In: *IEEE Transactions on Neural Networks and Learning Systems* 35 (1 Jan. 2024), pp. 770–782. ISSN: 21622388. DOI: 10.1109/TNNLS.2022.3177134.

[34] Siemens Healthineers. *Compressed Sensing GRASP-VIBE: Beyond Speed. Beyond Motion.* https://www.siemens-healthineers.com/magnetic-resonance-imaging/options-and-upgrades/clinical-applications/compressed-sensing-grasp-vibe. June 2025.

[35] Anagha Deshmane et al. "Parallel MR Imaging". In: *Journal of magnetic resonance imaging : JMRI* 36.1 (July 2012), pp. 55–72. DOI: 10.1002/jmri.23639.

[36] Li Feng et al. "Golden-Angle Radial Sparse Parallel MRI: Combination of Compressed Sensing, Parallel Imaging, and Golden-Angle Radial Sampling for Fast and Flexible Dynamic Volumetric MRI". In: *Magnetic Resonance in Medicine* 72.3 (2014), pp. 707–717. DOI: 10.1002/mrm.24980.

[37] Nahida Nazir, Abid Sarwar, and Baljit Singh Saini. "Recent developments in denoising medical images using deep learning: An overview of models, techniques, and challenges". In: *Micron* 180 (2024), p. 103615. DOI: 10.1016/j.micron.2024.103615.

[38] Bao Yang, Leslie Ying, and Jing Tang. "Artificial Neural Network Enhanced Bayesian PET Image Reconstruction". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1297–1309. DOI: 10.1109/TMI.2018.2803681.

[39] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT press, 2016.

[40] Vishwanath Saragadam et al. *WIRE: Wavelet Implicit Neural Representations*. Jan. 2023. DOI: 10.48550/arXiv.2301.05187. eprint: 2301.05187.

[41] Nasim Rahaman et al. *On the Spectral Bias of Neural Networks*. 2019. arXiv: 1806.08734 [stat.ML]. URL: https://arxiv.org/abs/1806.08734.

[42] Matthew Tancik et al. *Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains*. 2020. eprint: 2006.10739. URL: https://arxiv.org/abs/2006.10739.

[43] Ben Mildenhall et al. "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis". In: *European Conference on Computer Vision (ECCV)*. 2020. URL: https://arxiv.org/abs/2003.08934.

[44] Thomas Muller et al. "Instant Neural Graphics Primitives with a Multiresolution Hash Encoding". In: *ACM Transactions on Graphics* 41.4 (July 2022), pp. 1–15. DOI: 10.1145/3528223.3530127. eprint: 2201.05989.

[45] Shin-Fang Chng et al. "GARF: Gaussian Activated Radiance Fields for High Fidelity Reconstruction and Pose Estimation". In: *(ECCV)*. Springer, 2022, pp. 276–292. DOI: 10.1007/978-3-031-19827-4_16.

[46] Danzel Serrano, Jakub Szymkowiak, and Przemyslaw Musialski. "HOSC: A Periodic Activation Function for Preserving Sharp Features in Implicit Neural Representations". In: *arXiv preprint arXiv:2401.10967* (2024). URL: https://arxiv.org/abs/2401.10967.

[47] Hemanth Saratchandran et al. "A Sampling Theory Perspective on Activations for Implicit Neural Representations". In: *arXiv preprint arXiv:2402.05427* (2024). URL: https://arxiv.org/abs/2402.05427.

[48] Gabriel Peyré and Marco Cuturi. *Computational Optimal Transport: With Applications to Data Science*. Hanover, MA, USA: Now Publishers, 2019.

[49] Marco Cuturi. *Sinkhorn Distances: Lightspeed Computation of Optimal Transportation Distances*. June 2013. DOI: 10.48550/arXiv.1306.0895. eprint: 1306.0895.

[50] Riccardo Cristoferi. *NWI-WM246 - OPTIMAL TRANSPORT LECTURE NOTES*.

[51] Jean-David Benamou and Yann Brenier. "A Computational Fluid Mechanics Solution to the Monge-Kantorovich Mass Transfer Problem". In: *Numerische Mathematik* 84.3 (Jan. 2000), pp. 375–393. DOI: 10.1007/s002110050002.

[52] Khiem Pham et al. *On Unbalanced Optimal Transport: An Analysis of Sinkhorn Algorithm*. Nov. 2020. DOI: 10.48550/arXiv.2002.03293. eprint: 2002.03293.

[53] Sven Dummer, Puru Vaish, and Christoph Brune. *Joint Manifold Learning and Optimal Transport for Dynamic Imaging*. May 2025. DOI: 10.48550/arXiv.2505.11913. eprint: 2505.11913 (eess).

[54] Justin Solomon et al. "Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains". In: *ACM Trans. Graph.* 34.4 (July 27, 2015), 66:1–66:11. DOI: 10.1145/2766963.

[55] Matthew J Muckley et al. "TorchKbNufft: A High-Level, Hardware-Agnostic Non-Uniform Fast Fourier Transform". In: ().

[56] Yonatan Dukler et al. "Wasserstein of Wasserstein Loss for Learning Generative Models". In: *Proceedings of the 36th International Conference on Machine Learning*. PMLR, May 2019, pp. 1716–1725.

[57] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG]. URL: https://arxiv.org/abs/1412.6980.

[58] Sergey Kastryulin et al. "Image Quality Assessment for Magnetic Resonance Imaging". In: *arXiv preprint arXiv:2203.07809* (2022). URL: https://arxiv.org/abs/2203.07809.

[59] Zhou Wang et al. "Image quality assessment: From error visibility to structural similarity". In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612. DOI: 10.1109/TIP.2003.819861.

[60] Hossam El-Rewaidy et al. *Replication Data for: Multi-Domain Convolutional Neural Network (MD-CNN) For Radial Reconstruction of Dynamic Cardiac MRI*. Version V2. 2020. DOI: 10.7910/DVN/CI3WB6. URL: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/CI3WB6.

[61] Li Feng et al. "Golden-Angle Radial Sparse Parallel MRI: Combination of Compressed Sensing, Parallel Imaging, and Golden-Angle Radial Sampling for Fast and Flexible Dynamic Volumetric MRI". In: *Magnetic Resonance in Medicine* 72.3 (Sept. 2014), pp. 707–717. DOI: 10.1002/mrm.24980.

[62] Dayoung Baik and Jaejun Yoo. *Dynamic-Aware Spatio-temporal Representation Learning for Dynamic MRI Reconstruction*. 2025. arXiv: 2501.09049 [eess.IV]. URL: https://arxiv.org/abs/2501.09049.

[63] *Li Feng on Rapid Imaging, Professional Trajectories, and Never Giving Up Center for Advanced Imaging Innovation and Research*. Mar. 2023. URL: https://cai2r.net/li-feng-on-rapid-imaging-professional-trajectories-and-never-giving-up/ (visited on 06/18/2025).

[64] Thomas Müller et al. "Real-Time Neural Radiance Caching for Path Tracing". In: *ACM Transactions on Graphics* 40.4 (Aug. 2021), pp. 1–16. DOI: 10.1145/3450626.3459812.

[65] Kushal Vyas et al. *Learning Transferable Features for Implicit Neural Representations*. Jan. 2025. DOI: 10.48550/arXiv.2409.09566. eprint: 2409.09566 (cs).

[66] Burak Ozturkler et al. “Implicit data crimes: Machine learning bias arising from misuse of public data in computational imaging”. In: *arXiv preprint arXiv:2303.07353* (2023).

[67] Li Feng et al. “XD-GRASP: Golden-angle radial MRI with reconstruction of extra motion states using compressed sensing”. In: *Magnetic Resonance in Medicine* 75.2 (2016), pp. 775–788. DOI: 10.1002/mrm.25665.

[68] Li Feng et al. “Highly-accelerated real-time cine MRI using compressed sensing and parallel imaging”. In: *Proceedings of the 14th Annual Meeting of the International Society for Magnetic Resonance in Medicine (ISMRM)*. Stockholm, Sweden, 2010, p. 3602.

A Dataset visualization

Figure 17 shows a cropped region of the fully sampled image data from the first cardiac frame for all coils, focusing on the area around the heart. The radial k-space data was transformed to the image domain using a non-uniform fast Fourier transform (NUFFT). In addition, the corresponding k-space values for each coil are shown in Figure 18, where the square root of the absolute values is displayed to account for the large dynamic range of the data.

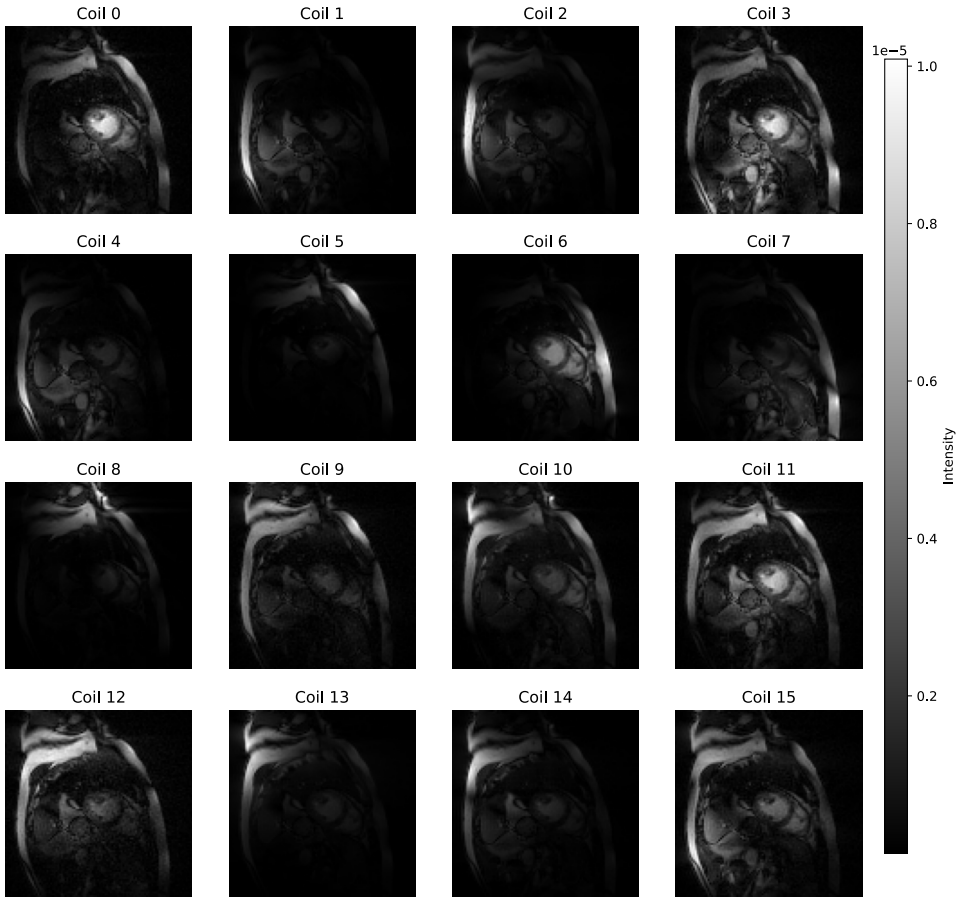


Figure 17: Absolute value images of all 16 coils for the first cardiac frame. Each image corresponds to the signal received by a different receiver coil, highlighting spatially varying sensitivity profiles and localized signal reception.

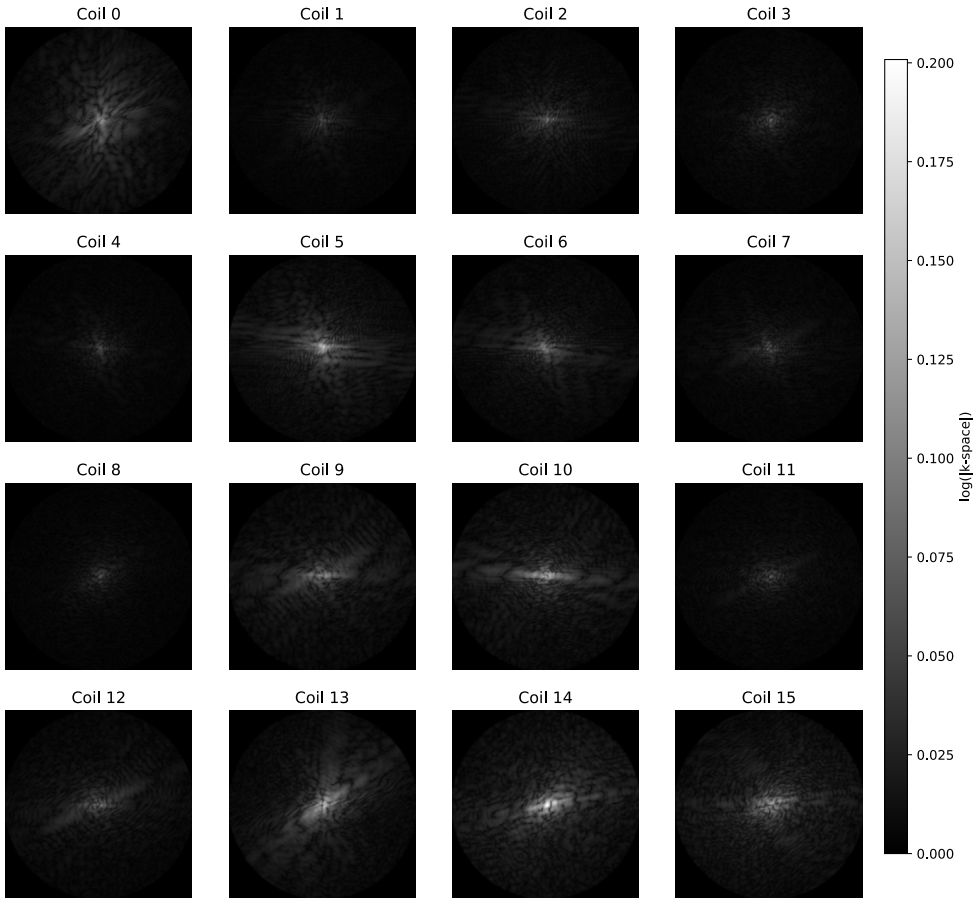


Figure 18: Log-magnitude k-space for the first frame across 16 coils. Log scaling highlights both high- and low-frequency content received by each coil.

B Wasserstein barycenter code

The code used for the Wasserstein barycenter where the default values of each function are used.

```
import math
import torch
import numpy as np
from einops import rearrange, parse_shape
from torch.nn.functional import avg_pool2d, avg_pool3d, interpolate

SUBSAMPLE = {
    2: (lambda x: 4 * avg_pool2d(x, 2)),
    3: (lambda x: 8 * avg_pool3d(x, 2)),
}

def pyramid(img, dim=2, height=32):
    img_s = [img]
    for _ in range(int(math.log2(height))):
        img = SUBSAMPLE[dim](img)
        img_s.append(img)
    img_s.reverse()
    return img_s

def epsilon_schedule(p, diameter, blur, scaling):
    r"""Creates a list of values for the temperature "epsilon"
       ↪ across Sinkhorn iterations.We use an aggressive strategy
       ↪ with an exponential cooling
    schedule: starting from a value of :math:`\text{diameter}^p`,
    the temperature epsilon is divided
    by :math:`\text{scaling}^p` at every iteration until reaching
    a minimum value of :math:`\text{blur}^p`.
    Args:
        p (integer or float): The exponent of the Euclidean
           ↪ distance
            :math:`\|x_i-y_j\|` that defines the cost function
            :math:`\text{C}(x_i,y_j) =\tfrac{1}{p} \|x_i-y_j\|^p`.
        diameter (float, positive): Upper bound on the largest
           ↪ distance between
            points :math:`x_i` and :math:`y_j`.

        blur (float, positive): Target value for the entropic
           ↪ regularization
            (":math:`\varepsilon = \text{blur}^p`").

        scaling (float, in (0,1)): Ratio between two successive
            values of the blur scale.
    Returns:
        list of float: list of values for the temperature epsilon.
    """

    eps_list = (
            [diameter ** p]
            + [
                math.exp(e)
                for e in np.arange(
```

```
            p * math.log(diameter), p * math.log(blur), p *
                ↪ math.log(scaling)
        )
            ]
            + [blur ** p]
    )
    return eps_list

def convolutional_barycenter_calculation(batch, weights=None,
    ↪ stab_thresh=1e-30, scaling=0.7, need_diffable=False):
    """
    computes the batched convolutional barycenter2d debiased in
        ↪ the log domain
    :param batch: input shape (M, B, C, H, W, [D])
    :param weights: the weight for each sample in the bary center
        ↪ calculation
    :param stab_thresh: default 1e-30, for not dividing by zero.
    :return: the barycenter of the batch with prescribed weights
    """
    def convol_img(_log_img, _kernel):
        _log_img = torch.logsumexp(_kernel[None, None, None, :, :,
            ↪ None] + _log_img[:, :, :, None, ...], dim=-2)
        _log_img = torch.logsumexp(
            _kernel[None, None, None, :, :, None] + _log_img[:, :,
                ↪ :, None, ...].permute(0, 1, 2, 3, -1, -2), dim=-2
        ).permute(0, 1, 2, -1, -2)
        return _log_img

    with torch.no_grad():
        if len(batch.shape) == 6:
            raise NotImplementedError("This method is currently
                ↪ not implemented for 3d")
        nh = batch.shape[0]  # number of histograms for each image
        b = batch.shape[1]  # batch size

        if weights is None:
            weights = 0.5 * torch.ones((nh, 1, 1, 1, 1),
                ↪ dtype=batch.dtype, device=batch.device)

        log_batch = torch.log(batch + stab_thresh)
        log_batch = rearrange(log_batch, 'm b c h w -> (m b) c h
            ↪ w')
        log_img_s = pyramid(log_batch,
            ↪ height=log_batch.shape[-1])[3:]
        log_img_s = [rearrange(log_img, '(m b) c h w -> m b c h
            ↪ w', m=nh, b=b) for log_img in log_img_s]

        _, b, c, h0, w0 = parse_shape(log_img_s[0], 'm b c h
            ↪ w').values()

        db = torch.zeros((b, c, h0, w0), dtype=log_batch.dtype,
            ↪ device=log_batch.device)

        g = torch.zeros(*log_img_s[0].shape,
            ↪ dtype=log_batch.dtype, device=log_batch.device)
```

```
        for s, log_img in enumerate(log_img_s):
            n = log_img.shape[-1]
            t = torch.linspace(0, 1, n, dtype=batch.dtype,
                ↪ device=batch.device)
            C = - (t.view(n, 1) - (t.view(1, n))) ** 2
            eps_list = epsilon_schedule(2, 2 / n, 1 / n, scaling)

            for eps in eps_list:
                for _ in range(
                        5 + 15 * (s == len(log_img_s) - 1 and eps
                            ↪ == eps_list[-1])
                ):  # do only the last iteration twice
                    m = C / eps
                    log_ku = convol_img(log_img - convol_img(g,
                        ↪ m), m)
                    log_bar = db + torch.sum(weights * log_ku,
                        ↪ dim=0)
                    db = 0.5 * (db + log_bar -
                        ↪ convol_img(db[None], m)[0])
                    g = log_bar[None, ...] - log_ku

            # if not the last scale
            if s != len(log_img_s) - 1:
                # upscale log_bar, c and g
                db = interpolate(db, scale_factor=2,
                    ↪ mode='bilinear', align_corners=False)
                g = rearrange(
                    interpolate(
                        rearrange(g, 'm b c h w -> (m b) c h w'),
                            ↪ scale_factor=2, mode='bilinear',
                            ↪ align_corners=False),
                    '(m b) c h w -> m b c h w', m=nh, b=b
                )

                log_bar = interpolate(log_bar, scale_factor=2,
                    ↪ mode='bilinear', align_corners=False)
    if need_diffable:
        with torch.enable_grad():
            log_img = torch.log(batch + stab_thresh)
            log_ku = convol_img(log_img - convol_img(g, m), m)
            log_bar = db + torch.sum(weights * log_ku, dim=0)
    return torch.exp(log_bar)
```