NeSVoR: Implicit Neural Representation for Slice-to-Volume Reconstruction in MRI

Junshen Xu [1], Daniel Moyer [2], Borjan Gagoski [2], Juan Eugenio Iglesias [2], P. Ellen Grant [2], Polina Golland [2], and Elfar Adalsteinsson [2]

[1]MIT
[2]Affiliation not available

October 30, 2023

Abstract

Reconstructing 3D MR volumes from multiple motion-corrupted stacks of 2D slices has shown promise in imaging of moving subjects, e.g., fetal MRI. However, existing slice-to-volume reconstruction methods are time-consuming, especially when a high-resolution volume is desired. Moreover, they are still vulnerable to severe subject motion and when image artifacts are present in acquired slices. In this work, we present NeSVoR, a resolution-agnostic slice-to-volume reconstruction method, which models the underlying volume as a continuous function of spatial coordinates with implicit neural representation. To improve robustness to subject motion and other image artifacts, we adopt a continuous and comprehensive slice acquisition model that takes into account rigid inter-slice motion, point spread function, and bias fields. NeSVoR also estimates pixel-wise and slice-wise variances of image noise and enables removal of outliers during reconstruction and visualization of uncertainty. Extensive experiments are performed on both simulated and in vivo data to evaluate the proposed method. Results show that NeSVoR achieves state-of-the-art reconstruction quality while providing two to ten-fold acceleration in reconstruction times over the state-of-the-art algorithms

In submission to IEEE Transactions on Medical Imaging (TMI)

NeSVoR: Implicit Neural Representation for Slice-to-Volume Reconstruction in MRI

Junshen Xu, Daniel Moyer, Borjan Gagoski, Juan Eugenio Iglesias, P. Ellen Grant, Polina Golland, Elfar Adalsteinsson

***Abstract*— Reconstructing 3D MR volumes from multiple motion-corrupted stacks of 2D slices has shown promise in imaging of moving subjects, *e.g.*, fetal MRI. However, existing slice-to-volume reconstruction methods are time-consuming, especially when a high-resolution volume is desired. Moreover, they are still vulnerable to severe subject motion and when image artifacts are present in acquired slices. In this work, we present NeSVoR, a resolution-agnostic slice-to-volume reconstruction method, which models the underlying volume as a continuous function of spatial coordinates with implicit neural representation. To improve robustness to subject motion and other image artifacts, we adopt a continuous and comprehensive slice acquisition model that takes into account rigid inter-slice motion, point spread function, and bias fields. NeSVoR also estimates pixel-wise and slice-wise variances of image noise and enables removal of outliers during reconstruction and visualization of uncertainty. Extensive experiments are performed on both simulated and *in vivo* data to evaluate the proposed method. Results show that NeSVoR achieves state-of-the-art reconstruction quality while providing two to ten-fold acceleration in reconstruction times over the state-of-the-art algorithms.**

***Index Terms*— MRI, slice-to-volume reconstruction, motion correction, super-resolution, 3D reconstruction, implicit neural representation, fetal brain MRI.**

I. Introduction

A. Motivation

High-resolution 3D Magnetic Resonance Imaging (MRI) plays an important role in clinical examinations but is vulnerable to artifacts caused by subject motion. To address this problem, ultra-fast sequences, *e.g.*, single-shot fast spin echo T_2-weighted imaging [1], have been developed to "freeze" in-plane motion, making within slice motion artifacts less severe compared to multi-shot methods. Nevertheless, inter-slice motion still exists and remains to be a problem. Therefore, in order to reconstruct 3D volumes, multiple stacks of slices at different orientations are acquired. These slices are then realigned to correct subject motion with slice-to-volume registration and then combined using super-resolution reconstruction [2]–[4]. This slice-to-volume reconstruction (SVR) framework has a wide range of applications in clinical practice and image analysis, including fetal and neonatal MRI [4], [5], cardiac MRI [6] and diffusion-weighted MRI [7]. Existing SVR algorithms explicitly represent the reconstructed volume as a discrete function on a pre-defined grid. In this formulation, the complexity and memory footprint of SVR is proportional to the number of voxels in the volume. For example, reducing the voxel spacing by half in every dimension would approximately increase the run time by a factor of eight. Moreover, the discrete representation of the volume may also introduce discretization error during reconstruction.

Recently, implicit neural representation (INR) has gained popularity in a variety of tasks in computer vision and graphics [8], [9]. In contrast to explicit representation, INR models a 2D slice or a 3D volume as a continuous function of spatial coordinates, and parameterizes the function with a neural network, *e.g.*, a multi-layer perceptron (MLP). INR has several advantages. i) It is resolution-agnostic, i.e., the network learns a continuous function during training and is able to sample volumes at different resolutions during inference. ii) Prior knowledge and constraints of the problem can be injected into the model by designing the architecture of the implicit network [10]. iii) INR can also overcome the high storage costs of dense discretized voxel grids [8].

However, few works have applied INR to 3D MRI reconstruction and the continuous forward model for slice acquisition is poorly studied. Here, we propose **Ne**ural **S**lice-to-**Vo**lume **R**econstruction (NeSVoR), a novel method to solve the problem of 3D volumetric MR reconstruction from multiple motion-corrupted stacks of slices utilizing implicit neural representation, which allows a continuous and resolution-agnostic representation of the reconstructed volume.

B. Related Works

1) Slice-to-Volume Reconstruction: Rousseau *et al.* [2] proposed a 3D fetal brain reconstruction approach that consisted of three steps: i) motion correction with multi-resolution slice alignment, ii) intensity correction for the local relative

(Corresponding authors: Junshen Xu.)

J. Xu and E. Adalsteinsson are with Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA, USA. (e-mail:{junshen, elfar}@mit.edu).

D. Moyer and P. Golland are with the Computer Science and Artificial Intelligence Lab (CSAIL), MIT, Cambridge, MA, USA. (e-mail:{dmoyer, polina}@csail.mit.edu).

B. Gagoski and P. E. Grant is with Fetal-Neonatal Neuroimaging and Developmental Science Center, Boston Children's Hospital, Boston, MA, USA, and Harvard Medical School, Boston, MA, USA. (e-mail:{borjan.gagoski, ellen.grant}@childrens.harvard.edu).

J. E. Iglesias is with the Center for Medical Image Computing, UCL, London, UK, the Martinos Center for Biomedical Imaging, Harvard Medical School, Boston, MA, USA, and the Computer Science and Artificial Intelligence Lab (CSAIL), MIT, Cambridge, MA, USA. (e-mail:jei@mit.edu).

intensity distortion between stacks, and iii) super-resolution reconstruction using scattered interpolation with a Gaussian point spread function (PSF). Jiang *et al*. [5] improved the scattered interpolation method by utilizing a multi-level B-spline kernel. Kim *et al*. [11] proposed a slice intersection motion correction method that realigned stacks of slices based on the slice intersection, followed by a gradient-weighted averaging step for volume reconstruction. Gholipour *et al*. [3] formulated volume reconstruction as a maximum likelihood error norm minimization problem and developed a robust M-estimation solution that reduced the influence of potential outliers in the acquired data. Building on this idea, Kuklisova-Murgasova *et al*. [4] proposed an SVR approach with complete outlier removal using robust statistics based on expectation maximization (EM). An additional intensity matching step was used to compensate for inconsistent scaling factors and bias fields of different slices. Tourbier *et al*. [12] extended the method in [4] with a total variation regularization, which can be solved efficiently using the primal-dual hybrid gradient (PDHG) method. Kainz *et al*. [13] developed a fast reconstruction algorithm based on [4], which leveraged the acceleration from multiple GPUs. Ebner *et al*. [14] proposed an automated reconstruction framework that included fetal brain localization and segmentation, and used a novel slice-level outlier rejection method by removing outlier slices with low similarity scores.

2) Implicit Neural Representation: The idea of INR has been widely applied in neural rendering. Mildenhall *et al*. [8] proposed Neural Radiance Fields (NeRF) to learn a 3D scene with 2D images at different camera positions. NeRF modeled the density and RGB color as continuous fields in 5D space (3D spatial location + 2D viewing direction) and simulated 2D observations from the radiance fields following the principles of volume rendering. The network was optimized by minimizing the error between the simulated and ground-truth images. To mitigate the misalignment of images in cases where ground-truth camera poses are unknown, NeRF−− [15] was proposed to optimize the camera parameters and neural networks simultaneously. NeRF-W [16] introduced image-dependent embedding vectors to model the appearance and transient objects that vary from image to image. These embeddings were optimized during reconstruction and helped synthesize scenes robust to variations in appearance and occluders.

In the field of medical imaging, attempts have been made to reconstruct super-resolved volumes from 2D slices using INR. IREM [17] was proposed for super-resolution reconstruction of adult brain MRI from stacks of thick slices, where only the motion between stacks is considered. Inspired by NeRF−−, Yeung *et al*. developed ImplicitVol [18] for 3D ultrasound reconstruction. ImplicitVol optimizes both the implicit network and the rigid transformation of each slice to compensate for inter-slice fetal motion. However, the aforementioned methods ignore the complex slice acquisition model as well as the artifacts and noise that occur during acquisition, and therefore cannot be directly applied to fetal or neonatal MRI.

Furthermore, the training of NeRF is known to be time-consuming. Recent researches have revealed that the encoding layer, *i.e*., the input layer of the implicit network, had a significant impact on the convergence of the network [19], [20]. Pre-defined encoding functions, such as sine and cosine, required a deep network and long training time to fit the underlying function. To this end, parametric encodings [20], [21] were proposed, which had trainable parameters in addition to the network weights. These parameters were arranged in a sparse data structure and helped reduce the depth of network and shorten training time significantly [20].

C. Contribution

In this work, we present NeSVoR, a novel SVR method than extends INR to learn a continuous representation of the unknown 3D volume from multiple 2D slices corrupted by subject motion and image artifacts. The main contributions of NeSVoR are: 1) We use INR to model the underlying volume as a neural network that is resolution-agnostic and more efficient than the explicit grid representation, especially when high-resolution volumes are desired. 2) In tandem with the INR, We adopt a continuous slice acquisition model that takes into account inter-slice subject motion, PSF, and bias fields. 3) We also introduce a novel approach for outlier removal during reconstruction by estimating the pixel-level and slice-level variances. 4) With GPU accelerated implementation, NeSVoR achieves 2 to 10 times speedup compared to the baselines while providing state-of-the-art results.

II. Materials and Methods

A. Slice Acquisition Model

1) Discrete Model: Let $I \in \mathbb{R}^{N_s \times N_p}$ be the data of the acquired slices, where I_{ij} is the intensity of the j-th pixel in the i-th slice, N_s and N_p are the number of slices and the number of pixels in each slice, respectively. The goal of 3D reconstruction is to find an unknown volume V of the 3D object, which is represented as an array $V \in \mathbb{R}^{N_v}$ in traditional methods, where N_v is the number of voxels. The forward slice acquisition model can be expressed as [3], [4], [13], [14]

$$I_{ij} = C_i B_{ij} \sum_{k=1}^{N_v} M_{ijk} V_k. \tag{1}$$

The relationship between voxels of the reconstructed volume and pixels of the acquired data is described by $M \in \mathbb{R}^{N_s \times N_p \times N_v}$, where M_{ijk} is the coefficient of the spatially aligned, discretized PSF for the acquisition of pixel I_{ij} from voxel V_k in the volume. B_{ij} is the multiplicative bias field for pixel I_{ij} and C_i is the scaling factor of the i-th slice which accounts for global intensity inconsistency in different slices.

2) Proposed Continuous Model: There are two disadvantages of the aforementioned discrete model. First, the discrete representation of the volume, bias field and, PSF might introduce discretization errors during reconstruction. Second, this formulation reconstructs a volume only at a specific resolution. To address these problems, we propose a continuous slice acquisition model in NeSVoR:

$$I_{ij} = C_i \int_{\Omega} M_{ij}(x) B_i(x) \left[V(x) + \epsilon_i(x) \right] \mathrm{d}x, \tag{2}$$

where Ω is the 3D region of interest (ROI). The main differences between the proposed model and the discrete model

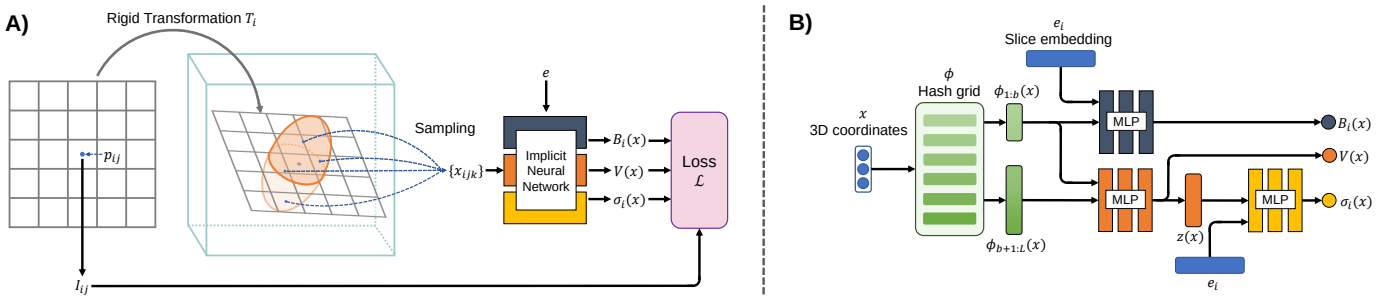

Fig. 1. A) The overview of NeSVoR. 1) We describe the relationship between the unknown volume and the acquired slices with a continuous slice acquisition model and approximate the effect of PSF with random sampling (Section II-A). 2) The sampled coordinates are then fed into the implicit neural network to regress bias field, volume intensity, and noise variance (Section II-B). 3) Finally, we train the model by minimizing the error between the simulated and acquired pixel values (Section II-C). B) The architecture of the proposed implicit neural network in NeSVoR. First, encodings of coordinates are generated from a multi-resolution hash grid data structure with look-up and interpolation. Then, the coordinate encodings and slice embeddings are fed to three different MLPs to regress bias field, volume intensity, and variance, respectively.

are as follows: i) Instead of discretized arrays, we model the volume, PSF, and bias field as continuous functions of spatial coordinates x. ii) The bias fields are modeled in volume coordinates rather than slice coordinates so that they share coordinates encoding with the volume in INR. Since the movement of the fetus changes its position relative to the scanner and creates inconsistencies in the bias field of the reconstructed volume, we keep the bias fields to be slice-dependent as in the previous model. iii) We also adopt a residual (noise) term ϵ_i in our formulation. The aim is twofold: to model slice-dependent noise in the acquired data; and to enable automatic outlier removal during reconstruction.

Assume that $\epsilon_i(x)$ is white Gaussian noise with $\mathbb{E}[\epsilon_i(x)] = 0$ and $\mathbb{E}[\epsilon_i(x)\epsilon_i(y)] = \sigma_i^2(x)\delta(x-y)$, where $\delta(\cdot)$ is the Dirac delta function. The mean and variance of pixel I_{ij} are

$$\overline{I}_{ij} = \mathbb{E}\left[I_{ij}\right] = C_i \int_\Omega M_{ij}(x)B_i(x)V(x)\mathrm{d}x, \tag{3}$$

$$\sigma_{ij}^2 = \mathrm{var}\left(I_{ij}\right) = C_i^2 \int_\Omega M_{ij}^2(x)B_i^2(x)\sigma_i^2(x)\mathrm{d}x. \tag{4}$$

In general, there are no closed-form solutions to the integrals in Eq. (3) and (4). Therefore, we use Monte Carlo sampling to estimate them. In many cases, the PSF can be modeled as an anisotropic 3D Gaussian distribution [2], [4],

$$M_{ij}(x) = g(T_i^{-1} \circ x - p_{ij}; \Sigma), \tag{5}$$

$$g(u;\Sigma) = \frac{1}{\sqrt{(2\pi)^3 \det(\Sigma)}} \exp\left(-\frac{1}{2}u^T\Sigma^{-1}u\right) \tag{6}$$

where T_i is the (unknown) rigid transformation from the i-th slice to the 3D space, p_{ij} is the location of pixel I_{ij} in the slice coordinates, and Σ is the covariance matrix of the Gaussian PSF. The expression $(T_i^{-1} \circ x - p_{ij})$ maps the 3D position x back to the slice coordinates centered at p_{ij}, where the PSF is unrotated. Therefore, we generate K i.i.d. samples from the Gaussian distribution, with $x_{ijk} = T_i \circ (u_{ijk} + p_{ij})$, $u_{ijk} \sim \mathcal{N}(\mathbf{0}, \Sigma)$, $k = 1, \ldots, K$, to compute Eq. (3) and (4).

$$\mathbb{E}[I_{ij}] = \frac{C_i}{K}\sum_{k=1}^{K} B_i(x_{ijk})V(x_{ijk}), \tag{7}$$

$$\mathrm{var}(I_{ij}) = \frac{C_i^2}{K}\sum_{k=1}^{K} M_{ij}(x_{ijk})B_i^2(x_{ijk})\sigma_i^2(x_{ijk}). \tag{8}$$

B. Implicit Neural Representation

1) Hash Grid Encoding: In INR, a volume is modeled as a continuous function $f(x)$ parameterized by a neural network that takes coordinates as inputs, $f(x) = \mathrm{MLP}(\phi(x))$, where ϕ is an encoding function that maps coordinates x to a high-dimensional feature vector which is then fed into an MLP to fit $f(x)$. For example, ϕ can be the multi-resolution sequence of L sine and cosine functions [8], [19], $\phi(x) = [\sin(2^0x), \ldots, \sin(2^{L-1}x), \cos(2^0x), \ldots, \cos(2^{L-1}x)]$. However, with fixed encoding functions, we only rely on the weights of MLP to fit the target function $f(x)$, and thus require a deeper network that typically converges slower.

To enable fast INR training, we adopt the recently proposed hash grid encoding [20], which arranges additional trainable parameters in multi-resolution 3D grids, and therefore reduces the depth of MLP. Specifically, Let $\Phi_l \in \mathbb{R}^{N_l \times N_l \times N_l \times F}$ be the grid of parameters at the l-th level, $l = 1, \ldots, L$, where L is the number of levels, N_l is the size of the l-th grid in every dimension, and each vertex of the grid conceptually stores a feature vector with a length of F. To compute the encoding at position x, we perform trilinear interpolation on the grid, *i.e.*, we find the eight vertices around position x, and compute the linear combination of the feature vectors stored in these eight vertices as the feature vector at position x.

Starting with the coarsest grid with a size of N_1, each following grid increases the size by a factor of s, *i.e.*, $N_l = \lfloor N_1 s^{l-1} \rfloor$. The coarse-to-fine strategy enables the model to learn multi-scale features in a progressively refined manner, where low-level grids encode slowly varying features, such as bias field, while high-level grids learn high-frequency details, like edges in the image. However, if we store the multi-level grids naively as dense 3D arrays, the memory footprint of each level increases as cube of grid size, $O(N_l^3)$. In multi-scale representation, high-resolution details tend to be sparse, so a large amount of memory space in the dense array is wasted. To this end, the dense 3D array Φ_l is replaced by a hash table $\Phi_l^{\mathrm{hash}} \in \mathbb{R}^{N_h \times F}$, where N_h is the size of the hash table and $N_h \ll (N_L)^3$. Therefore, the query of the grid Φ_l at index (i,j,k) is translated into two steps: i) mapping (i,j,k) to a hash code with a hash function, ii) accessing the hash table Φ_l^{hash} with the hash code, $\Phi_l(i,j,k) = \Phi_l^{\mathrm{hash}}(i \oplus j\pi_1 \oplus k\pi_2$

$\mod N_h$), where π_1 and π_2 are two large primes, and $\oplus$ denotes the bit-wise XOR operation.

With the hash table structure, we essentially compress the grids at high levels so that they have a much smaller memory footprint and only store details that are necessary for volume reconstruction. The final encoding provided to the networks is the concatenation of encodings at different levels, $\phi(x) = [\phi_1(x), \ldots, \phi_L(x)]$.

2) Implicit Networks: The architecture of the proposed implicit network is shown in Fig. 1-B. Given the multi-resolution encoding $\phi(x)$, an MLP is used to regress the intensity of the volume at position x. The MLP also outputs a feature vector $z(x)$ for downstream processing.

$$[V(x), z(x)] = \text{MLP}_V(\phi(x)) \tag{9}$$

Since bias fields are slice-dependent, we model the slice-specific information with latent variable optimization [16], [22] by assigning each slice an embedding vector $e_i, i = 1, ..., N_s$. These slice embedding vectors are trainable and able to learn slice-specific information during optimization. It is worth noting that we do not incorporate these embeddings in MLP_V, since we want MLP_V to only learn information that is slice-independent, *i.e.*, the intensity of the underlying volume.

Note that in Eq. (7), $B_i(x)$ is more general than $V(x)$, as it can be slice-dependent. Hence, if we use the same input encoding $\phi(x)$ as in MLP_V, $B_i(x)$ may learn the product of bias field and volume such that $V(x)$ becomes a constant. To avoid this undesired solution, we need to limit the information going through the bias field branch. One important prior of the bias field is that it is a smoothly varying function of spatial location. Therefore, instead of the full encoding $\phi(x)$, we only use the first b levels of the encoding as the input to the bias field network, which contains the low-frequency information. In summary, a second MLP is adopted to regress the bias field $B_i(x)$ from the low-level encoding $\phi_{1:b}(x) = [\phi_1(x), \ldots, \phi_b(x)]$ and the slice embedding e_i as well.

$$B_i(x) = \text{MLP}_B(\phi_{1:b}(x), e_i). \tag{10}$$

The last component for evaluating Eq. (7) and (8) is the variance σ_i^2 which is also slice-dependent. We use a third MLP to estimate the variance at location x from the feature vector $z(x)$, and the slice embedding e_i,

$$\sigma_i^2(x) = \text{MLP}_\sigma(z(x), e_i). \tag{11}$$

C. Loss Functions

1) Slice Reconstruction: Given the estimates of the mean and variance of pixel intensity in Eq. (7) and (8), the underlying volume can be reconstructed by minimizing the negative log-likelihood of Gaussian distribution:

$$\mathcal{L}_{ij} = \frac{\left(I_{ij} - \overline{I}_{ij}\right)^2}{2\sigma_{ij}^2} + \frac{1}{2}\log\left(\sigma_{ij}^2\right). \tag{12}$$

Another way to interpret this loss function is from the perspective of outlier removal. The acquired MR slices are often corrupted by different types of artifacts, *e.g.*, motion blurring, and spin history. Such slices or pixels should be excluded during reconstruction to avoid artifacts in the reconstructed volume. The precision $1/\sigma_{ij}^2$ can be interpreted as the weight of pixel I_{ij}. The model should assign a large variance (small weight) to the outlier so that they would be ignored during reconstruction. Also, the log-variance term prevents σ_{ij}^2 from going to infinity. The NeSVoR model is optimized with stochastic gradient descent, *i.e.*, in each iteration, a batch of data $\mathcal{B} \subset \{1, \ldots, N_s\} \times \{1, \ldots, N_p\}$ is sampled to compute the loss function:

$$\mathcal{L}_I = \frac{1}{|\mathcal{B}|}\sum_{(i,j)\in\mathcal{B}} \mathcal{L}_{ij} \tag{13}$$

2) Image Regularization: SVR is an ill-posed problem due to subject motion and insufficient ROI coverage. Thus, regularization methods are adopted to improve image quality and suppress noise. Although the network architecture implicitly regularizes the outputs [10], [23], we provide a way to add (optional) explicit regularizations to the loss function to demonstrate the flexibility of NeSVoR. A widespread approach is the first-order regularizer,

$$\mathcal{R}_V = \int_\Omega r(\|\nabla V(x)\|_2)\mathrm{d}x. \tag{14}$$

The function r can be the identity function (isotropic total variation), square function (first-order Tikhonov), or Huber function (edge-preserving). Although it is possible to compute $\nabla V(x)$ with automatic differentiation, the extra computation graph significantly increases computational cost. Instead, we approximate the first-order regularizer by estimating the directional derivative from the random samples. Specifically, we split the K samples for computing Eq. (7) and (8) into $K/2$ pairs, $\{(1, 1 + K/2), \ldots, (K/2, K)\}$. The directional derivative for each pair is $|V(x_{ijk}) - V(x_{ijl})|/\|x_{ijk} - x_{ijl}\|_2$, where $l = k + K/2$. Then the regularization can be approximated by

$$\mathcal{R}_V = \frac{2}{K|\mathcal{B}|}\sum_{(i,j)\in\mathcal{B}}\sum_{k=1}^{K/2} r\left(\frac{|V(x_{ijk}) - V(x_{ijl})|}{\|x_{ijk} - x_{ijl}\|_2}\right). \tag{15}$$

This method requires no extra forward/backward pass of the network, and therefore, adds only marginal computational cost.

We use isotropic total variation as the default regularization for the reconstructed volume.

3) Bias Field: In Eq. (7), the bias field B_i and volume V are only unique up to a constant factor. If (B_i, V) is a solution, $(cB_i, \frac{1}{c}V)$ is also a feasible solution for any constant $c > 0$. In order to disambiguate, extra constraints are required. For example, we can force the mean of log bias field to be zero, $\int_\Omega \log B_i(x)\mathrm{d}x = 0$, which can be achieved with the following regularization of the sample mean of log bias field:

$$\mathcal{R}_B = \left(\frac{1}{K|\mathcal{B}|}\sum_{(i,j)\in\mathcal{B}}\sum_{k=1}^{K}\log B_i(x_{ijk})\right)^2 \tag{16}$$

D. Other trainable parameters

1) Transformation: To allow unconstrained optimization of the rigid transformation with gradient descent, we adopt the axis-angle representation for the transformation of each slice,

i.e., the transformation of the i-th slice T_i is parameterized by a six-dimensional vector $(\theta_i n_{i1}, \theta_i n_{i2}, \theta_i n_{i3}, t_{i1}, t_{i2}, t_{i3})$, where (n_{i1}, n_{i2}, n_{i3}) is a unit vector representing the rotation axis, θ_i is the rotation angle, and (t_{i1}, t_{i2}, t_{i3}) is the translation vector.

2) Slice Scaling Factor: The scaling factor C_i in Eq. (7) introduces an arbitrary constant factor to the solution, and therefore constraints on C_i need to be imposed. Here, we assume that the average of the scaling factor is 1, *i.e.*, $\frac{1}{N_s}\sum_{i=1}^{N_s} C_i = 1$ and reparameterize C to satisfy this constraint,

$$C = N_s \text{softmax}(c), \quad C_i = \frac{N_s \exp(c_i)}{\sum_{j=1}^{N_s} \exp(c_j)}, \tag{17}$$

so that the new parameter vector c is unconstrained.

3) Slice-wise Variance: Under severe artifacts, the whole slice might be corrupted. Therefore, in addition to the pixel-wise variance $\text{var}(I_{ij})$, we also introduce a slice variance ν_i^2 which downplays the whole slice from reconstruction when the entire slice is of low quality. Eq. (4) is then modified as

$$\sigma_{ij}^2 = \text{var}\,(I_{ij}) + \nu_i^2, \tag{18}$$

i.e., the total variance of pixel I_{ij} is the sum of pixel-wise variance $\text{var}(I_{ij})$ and slice-wise variance ν_i^2.

E. Training and Inference

During training, we solve the optimization problem:

$$\arg\min_{\Theta} \mathcal{L}(\Theta), \quad \mathcal{L} = \mathcal{L}_I + \lambda_B \mathcal{R}_B + \lambda_V \mathcal{R}_V, \tag{19}$$

where λ_B and λ_V are the weights for the regularization terms, Θ is the set of trainable parameters, including the weights of MLPs, the hash grid Φ^{hash}, the slice transformations T, the slice embeddings e, the scaling factors c, and the log slice variances $\log \nu^2$. We adopt an Adam optimizer [24] with an initial learning rate of 5×10^{-3} which decays with a factor of $\gamma = 1/3$ at iteration $N_\tau/2$ and $3N_\tau/4$, where N_τ is the total number of iterations. We use a batch size of 4096 and set the number of samples $K = 128$. The covariance matrix of the Gaussian PSF is defined as in [4], $\Sigma = \text{diag}((\frac{1.2r_1}{2.355})^2, (\frac{1.2r_2}{2.355})^2, (\frac{r_3}{2.355})^2)$, where r_1 and r_2 are the in-plane pixel spacings and r_3 is the slice thickness.

All MLPs have one hidden layer with 64 units and ReLU activation. MLP_V use softplus as the output activation while the other MLPs use the exponential function. The length of slice embedding is set to 16 and the slice embedding is initialized with the standard normal distribution. For the hash encoding, we choose the hyperparameters following the strategy in [20], $s = 1.38$, $N_1 \in [6, 16]$, $L \in [9, 12]$, depending on the size of slices. The parameters in the hash grid are initialized with a uniform distribution $U(-10^4, 10^{-4})$. When there are more than one input stacks, we first perform a volume-to-volume registration to coarsely correct the motion between different stacks [4]. The stack transformation after volume-to-volume registration is then used to initialize the transformation of each slice T_i.

After training the model, $V(x)$ learns a continuous representation of the underlying volume. Slices at different views and volumes of different field of views (FOV) can then be sampled from the function $V(x)$. Sampling directly from $V(x)$ might result in aliasing and image noise. Therefore, we sample the intensity at position x using the PSF model,

$$V^{\text{out}}(x) = \int_{\Omega} M(x)V(x)\text{d}x. \tag{20}$$

where M is an isotropic Gaussian PSF with $\sigma = r/2.3548$ and r is the isotropic voxel spacing of the output volume.

F. Implementation

All models were tested on a system with two Intel Xeon Gold 6238R CPUs @2.20 GHz with 768 GB RAM, and an NVIDIA Tesla V100 GPU with 32 GB RAM. The networks were implemented with PyTorch [25] and Tiny CUDA NN [20], [26]. To further accelerate the training of INR, we adopt the strategy of automatic mixed precision training [27] where the hash grid and MLPs are trained with half-precision format while the other parameters are stored in single-precision format. The source code is available on GitHub[1].

III. Experiments

A. Datasets

We performed extensive experiments to evaluate NeSVoR using the following four datasets.

1) Simulated Adult Brain Data: An Adult brain MRI dataset was synthesized from the data in the Human Connectome Project (HCP) [28]. We randomly selected the T_1-weighted and T_2-weighted images of 30 subjects, which were acquired at 0.7 mm isotropic resolution, and used as ground truth. We simulated 3 orthogonal stacks of slices from each volume, with in-plane resolution of 1 mm and slice thickness of 2 mm. Simulated inter-slice motion with random translations and rotations was incorporated. The translations along the $x-$, $y-$, and $z-$axes of each slice were sampled independently from the range of $[-3, 3]$ mm. The angles of 3D rotation were randomly sampled from the range of $[-6, 6]$ degree. In-plane motion artifacts and ghosting artifacts were simulated as in [29]. Rician noise [30], with a standard deviation of 3% the maximum intensity, was added to each slice.

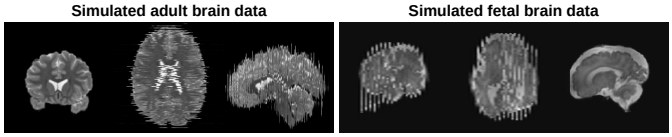

Fig. 2. A simulated stack of the adult brain data (left) and fetal brain data (right).

2) Simulated Fetal Brain Data: We also simulated fetal data from the FeTA [31] dataset, which consisted of fetal brain volumes with 0.5 mm isotropic resolution. We selected 10 volumes with gestational age (GA) from 27 to 35 weeks as ground truths. For each subject, 3 orthogonal stacks were simulated with in-plane resolution of 0.8 mm and slice thickness of 3 mm. Fetal brain motion trajectories were simulated using

[1]https://github.com/daviddmc/NeSVoR

the method in [32]. Specifically, we sampled head motion from a fetal keypoint dataset [33] that represents realistic fetal brain motion trajectories during MRI scans. The maximum translation and rotation motion in the motion trajectory dataset are 21.4 mm/s and 59.7 degree/s respectively. The fetal brain volumes were transformed according to the sampled trajectory, and slices were extracted from the volume at the corresponding positions. Bias fields and signal void artifacts are also simulated. Image noise was added as in the adult brain data.

Fig 2 shows example stacks from the two simulated datasets. The goal of the simulated data is twofold: i) to quantitatively evaluate our approach with ground-truth data, ii) to show that the proposed method can be applied to data with different contrasts, sizes of ROI, and image artifacts.

3) Clinical Neonatal Brain Data: We used the clinical neonatal brain data from the Developing Human Connectome Project (dHCP) [34] to evaluate the proposed method. The raw T_2-weighted magnitude images of 10 neonatal subjects were selected from this dataset. Each subject consists of 2 to 4 image stacks with in-plane resolution of 0.8 mm, slice gap of 0.8 mm, and slice thickness of 1.6 mm. Details on acquisition parameters can be found in [34].

4) Clinical Fetal Brain Data: A fetal MRI dataset was collected to evaluate the method. This dataset consisted of T_2-weighted MRI from 20 fetuses, with GA from 21 to 32 weeks. All scans were performed in accordance with the local institutional review board protocol. The data were acquired with in-plane resolution of 1-1.3 mm, slice thickness of 2-4 mm, no gap, TE = 100-120 ms, TR = 1.4-1.8 s. Each subject had 3 to 10 stacks of slices. Fetal brains are segmented from the slices using an existing, trained segmentation network [35].

B. Baselines

We adopted as baseline three state-of-the-art SVR methods that have open-source implementations: 1) SVRTK[2]: The slice-to-volume reconstruction toolkit is an implementation of the algorithm in [4], which is accelerated with multi-core parallelism on CPU. 2) SVR-GPU[3]: This approach [13] is a GPU-accelerated implementation of [4]. Although it has been pointed out in previous works [14] that the GPU-accelerated approach tends to produce blurrier outcomes, we still incorporated this method in experiments mainly for comparing the efficiency of algorithms on GPU. 3) NiftyMIC[4]: It is the implementation of the reconstruction algorithm in [14], which runs on CPU with no parallelism and is slow when the size output volume is large. We excluded this method from the experiments on the simulated adult brain data and the neonatal brain data as the run time for each case exceeded 12 hours.

For tuning hyperparameters for different methods, we randomly picked one subject from the simulated adult/fetal dataset and adjusted the hyperparameters to minimize the mean squared error between the reconstructed volume and the ground truth. The tuned hyperparameters were then applied to the other data in the same dataset. For the clinical neonatal brain dataset, we used the same hyperparameters as the simulated adult brain data, and for the clinical fetal brain dataset, we used the same hyperparameters as the simulated fetal brain data.

[2]https://github.com/SVRTK/SVRTK
[3]https://github.com/bkainz/fetalReconstruction
[4]https://github.com/gift-surg/NiftyMIC

C. Results on Simulated Data

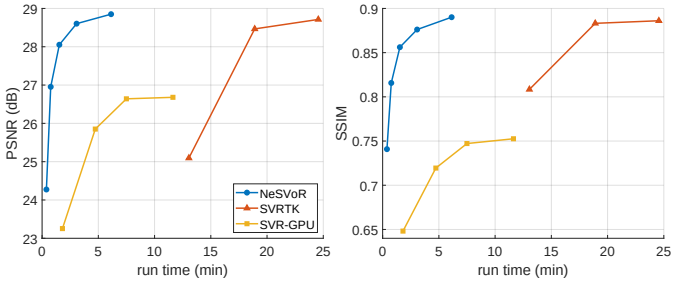

Fig. 3. The image quality of reconstructed volume vs. run time (number of iterations) for different methods. NeSVoR: number of iterations = 1000, 2000, 4000, 8000, 16000; SVRTK: number of outer iterations = 1, 2, 3; SVR-GPU: number of outer iterations = 1, 2, 4, 6.

We reconstructed the simulated adult and fetal data at the isotropic resolution of 0.7 mm and 0.5 mm respectively to match the original resolutions of the ground truths. We compared the reconstructed volumes and ground truths by different quantitative metrics, including peak signal-to-noise ratio (PSNR), structural similarity (SSIM) [36], normalized root mean square error (NRMSE), and normalized cross-correlation (NCC). Results are shown in Table I. NeSVoR achieved comparable results with SVRTK on the simulated adult brain data with 4× speedup. Although SVR-GPU was faster than SVRTK due to GPU acceleration, the reconstruction quality of this implementation was lower than NeSVoR and SVRTK. For the simulated fetal data, NeSVoR outperformed the baselines in terms of both accuracy and speed.

To further study the trade-off between the run time and the reconstruction quality, we ran NeSVoR on the simulated adult data with different numbers of training steps N_τ and altered the number of outer iterations in the baselines, *i.e.*, the number of cycles of registration and reconstruction. The resulting curves are shown in Fig. 3. NeSVoR converged much faster than SVRTK, requiring only 6% to 25% run time to achieve comparable results with SVRTK. Although running on GPU, SVR-GPU suffered from lower image quality compared to the other methods, resulting in a sub-optimal trade-off curve.

Since fetal MRI routinely suffers from subject motion during scans, we also evaluated the methods with different motion levels. We randomly chose a fetal brain volume and simulated motion trajectories of different levels (3D translation and 3D rotation were simulated and evaluated separately). Fig. 4 shows the PSNRs and SSIMs of different methods. In comparison, NeSVoR is more robust than the baselines when the motion is small to moderate. As the intensity of motion increases, the PSNRs and SSIMs of all the reconstruction methods decrease. Since the slice transformations are optimized by maximizing the local similarity between the slices and the volume, they easily get stuck in local minima, and therefore, result in a limited capture range of motion.

TABLE I

MEAN VALUES OF QUANTITATIVE METRICS FOR DIFFERENT MODELS ON THE SIMULATED DATASETS (STANDARD DEVIATION IN PARENTHESES). ↓ INDICATES LOWER VALUES BEING MORE ACCURATE, AND VICE VERSA.

Methods	PSNR / dB ↑	SSIM ↑	NRMSE ↓	NCC ↑	run time / min ↓
Simulated adult brain data					
SVRTK [4]	28.71 (3.33)	0.8861 (0.0459)	0.1129 (0.0452)	0.8706 (0.0598)	24.57 (3.42)
SVR-GPU [13]	26.68 (4.38)	0.7526 (0.1936)	0.1518 (0.0867)	0.7360 (0.2775)	11.63 (2.23)
NeSVoR	28.85 (3.22)	0.8901 (0.0317)	0.1103 (0.0427)	0.8772 (0.0526)	6.13 (0.30)
Simulated fetal brain data					
SVRTK [4]	22.10 (2.39)	0.8694 (0.0792)	0.1772 (0.0504)	0.6517 (0.2007)	8.38 (2.83)
SVR-GPU [13]	21.64 (0.70)	0.8031 (0.0460)	0.1809 (0.0169)	0.6552 (0.0779)	2.36 (0.87)
NiftyMIC [14]	21.09 (2.20)	0.7919 (0.1651)	0.1978 (0.0525)	0.5647 (0.2445)	189.24 (79.81)
NeSVoR	23.63 (1.17)	0.9290 (0.0354)	0.1446 (0.0199)	0.7804 (0.055)	1.92 (0.09)

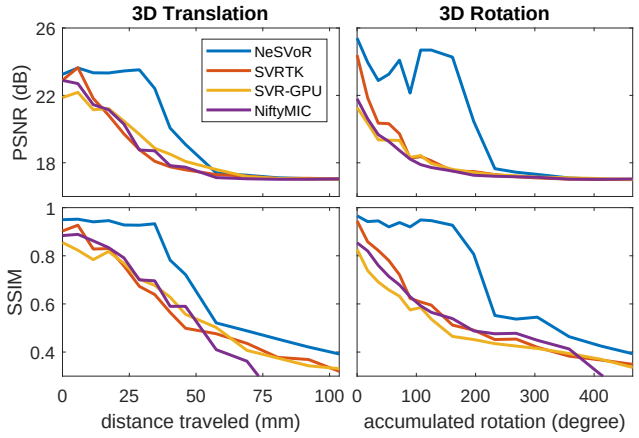

Fig. 4. Comparative reconstruction performance (PSNR and SSIM) of different methods on the simulated fetal data with different levels of motion. The x-axes represent the distance traveled and the accumulated rotation over the trajectory for translation and rotation, respectively.

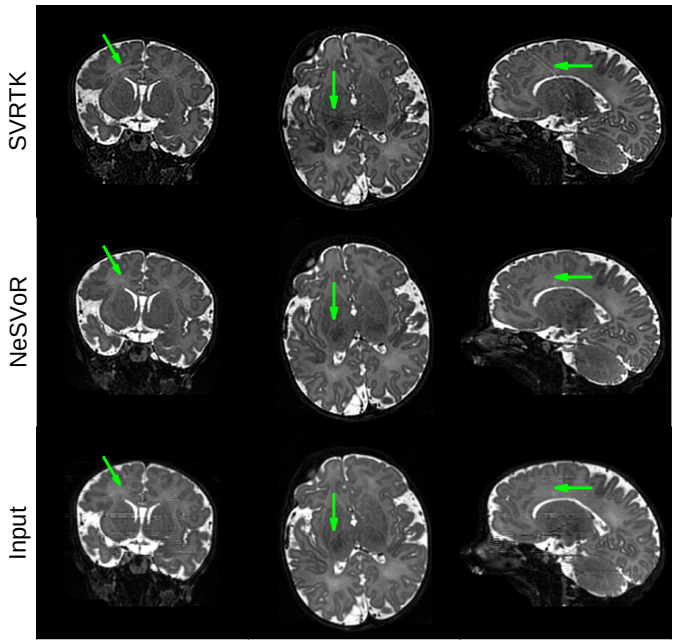

Fig. 6. The reconstruction results and an input stack of a subject in the dHCP dataset. Green arrows indicate artifacts in SVRTK that are eliminated in NeSVoR.

D. Results on Clinical MRI data

We reconstructed the neonatal data with resolution of 0.5 mm. As there was no ground truth, we evaluated the methods by measuring the consistency between the output volumes and the input slices. We extracted slices from the motion-corrected locations and computed the NCC and SSIM between the extracted slices and the corresponding slices of an input stack. The results are presented in Fig. 5 and show that NeSVoR had similar SSIM and higher NCC while achieving 9× speedup on average compared to SVRTK. Fig. 6 shows the reconstruction results of a neonatal subject. NeSVoR produced results with fewer image artifacts than the baseline method.

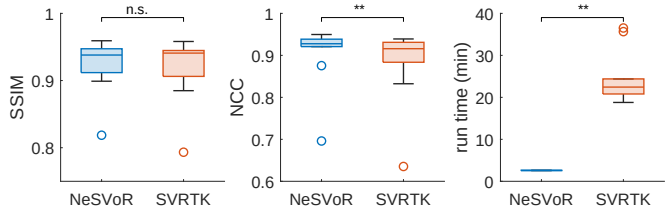

Fig. 5. Quantitative comparison based on the similarity metrics between the input slices and the slices extracted from the reconstructed volumes. Results of Wilcoxon signed rank test are presented, $*$: $p<0.05$, $**$: $p<0.01$, and n.s.: not significant ($p\geq 0.05$).

For the clinical fetal data, we reconstructed volumes with isotropic resolution of 0.8 mm. Since there was also no ground truth for the clinical fetal data, we adopted an automated MRI quality assessment (QA) approach to evaluate the image quality of reconstructed volumes. Specifically, we used the trained QA network proposed in [37], which can predict a QA score between 0 and 1 for a 2D T_2 weighted fetal brain MR image to assess the artifacts in the image, with higher scores indicating better image quality. The score of the volume was computed as the average of the scores of all slices in the volume. We also evaluated the reconstructed volumes in terms of signal-to-noise ratio (SNR) and partial volume effect (PVE). We considered the Gaussian mixture model (GMM) in [38], [39] that models three types of brain tissues, i.e., cerebrospinal fluid (CSF), gray matter (GM), and white matter (WM). The SNR of a volume is computed as $\text{SNR} = 20\log_{10}(\mu/\sigma)$, where s and σ are the weighted averages of the mean signal intensities and the standard deviations of noise of three Gaussian components. We consider the voxels are from the k-th tissue if their intensities are within $\pm\delta_k$ of the mean of the k-th Gaussian component, where δ_k is the corresponding

half FWHM. Thus, the percentage of voxels outside the three ranges can be used as a proxy for PVE.

Fig. 7 shows the QA scores, SNRs, and PVEs of different methods. NeSVoR achieved higher image quality and SNR compared to the baselines. While there is no difference among the PVEs of different reconstruction algorithms. Fig. 8 presents a visual comparison of NeSVoR and the baselines for three challenging cases that are corrupted by severe fetal motion. NeSVoR yielded results with the best perceptual quality.

E. Reconstructing Volumes at Different Resolutions

In NeSVoR, the INR learns a continuous representation of the reconstructed volume. Therefore, the model only needs to be trained once, from which volumes at different resolutions can be sampled (the time for sampling a volume is negligible compared to the training time). In comparison, conventional methods reconstruct a volume only at a specific resolution, and therefore, to reconstruct volumes at different resolutions, the algorithm needs to be re-run. We call this property of NeSVoR resolution-agnostic reconstruction.

To demonstrate this, we reconstructed a fetal subject with different isotropic voxel spacings (0.8 mm, 0.6 mm, and 0.4 mm). We also performed the same experiment using SVRTK, where the reconstruction algorithm was re-run with different resolutions. Fig. 9 shows the reconstruction results at different resolutions for the two methods. Volumes reconstructed at higher resolution yielded sharper edges. The bar plot in Fig. 9 shows the run time of the two methods. The run time of SVRTK increases drastically as the voxel spacing decreases because the number of voxels in the volume is inversely proportional to the cube of voxel spacing. In contrast, the run time of NeSVoR is independent of the voxel spacing. When reconstructing a volume with 4 mm voxel spacing, NeSVoR achieved $18\times$ speedup compared to SVRTK.

F. Ablation Study

To investigate the contribution of each component in NeSVoR, we evaluated the model on the simulated fetal dataset by ablating PSF, bias field estimation, transformation optimization, variance estimation, slice embedding, and INR. When ablating INR, we represented the volume, bias fields, and variance as dense 3D grids that were optimized directly. The PSNR and SSIM of different variants of the model are shown in Table. II. The results show that the full model outperforms other variants.

1) Bias Field: We compared the reconstructed volumes with and without bias field correction, and the results are shown in Fig. 10. NeSVoR was able to mitigate the effect of bias field. For comparison, we performed the same experiment with SVRTK whose forward model also took into account the bias fields. However, it failed to correct the smoothly varying bias field in this subject. The last row shows a selected intensity profile of the resulting reconstructions without bias field correction (bottom left) and with bias field correction (bottom middle).

TABLE II
MEAN VALUES OF PSNR AND SSIM FOR DIFFERENT ABLATED MODELS ON THE SIMULATED FETAL BRAIN DATASETS (STANDARD DEVIATION IN PARENTHESES).

Methods	PSNR / dB	SSIM
full model	23.63 (1.17)	0.9290 (0.0354)
w/o PSF	19.49 (1.23)	0.7101 (0.0916)
w/o bias field correction	23.57 (1.29)	0.8848 (0.0623)
w/o transformation optimization	19.26 (0.70)	0.7179 (0.0495)
w/o variance estimation	17.59 (0.85)	0.4917 (0.1438)
w/o slice embedding	19.94 (0.63)	0.8856 (0.0317)
w/o INR	18.17 (0.48)	0.7402 (0.0599)

Moreover, since we implement the bias field model in a separate network, computational cost can reduced by disabling this module, when the effect of the bias field is small or when other techniques of bias field correction are available.

2) Variance: Fig. 11 shows examples of estimated slice-wise and pixel-wise variances. From the original slices (the first row), we can see that images corrupted by severe artifacts have high values of log slice variance $\log \nu_i^2$ (the number labeled on top of each slice), indicating that NeSVoR can identify outlier slices and reduce their influence by assigning high slice variances. The second row of Fig. 11 shows the maps of pixel-wise variance learned by the model. The pixels with large variances match the locations of image artifacts. Thus, the variance maps also provide a way to visualization of pixel-level uncertainty. The reconstructed volumes show that the result without the variance model suffered from severe artifacts propagated from the corrupted slices, while the variance model succeeded in excluding those slices from reconstruction.

3) PSF: Fig. 12 shows the reconstructed volumes of the full model and the model ablating PSF during training or inference. The model trained without PSF suffered from partial volume effects leading to blurred results. Ablating the PSF during inference improved the sharpness of the output while being more vulnerable to image noise and aliasing.

G. Understanding Hash Grid Encoding

Fig. 13 visualizes the learned hash grids at different levels as well as the corresponding reconstruction result. The low-level hash grid is of low resolution, and therefore, learns low-frequency features in the images. The middle level has a finer grid, but the conceptual grid size is still comparable to the actual size of the hash table, so it is able to encode anatomical structures of the brain volume. For the high-level grid, however, the conceptual grid size is far greater than the actual size of the hash table, resulting in severe hash collisions. Therefore, it would encode some sparse features or high-frequency noise in the images.

One of the advantages of hash grid encoding compared to other encodings, *e.g.*, frequency encoding, is the convergence speed. Fig. 14 shows the convergence of NeSVoR on a fetal subject. brain data. The model converged to a high-quality volume in one to two minutes. Fig. 15 shows the reconstruction results of the same subject by replacing the hash grid encoding with frequency encoding and using eight-layer MLPs as in [16]. The frequency encoding needs much

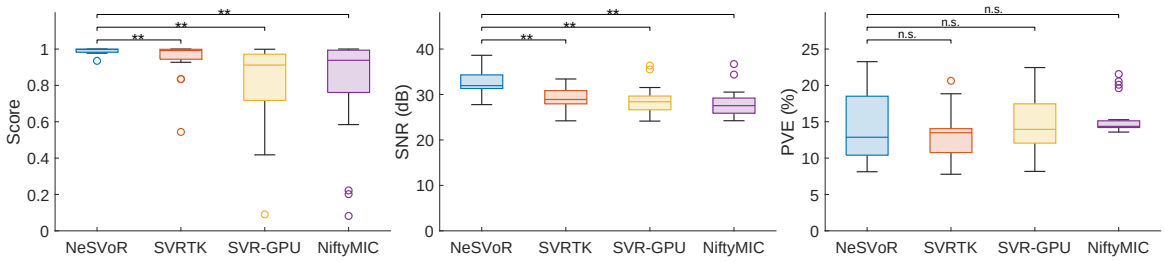

Fig. 7. QA scores, SNR, and PVE of different methods on the clinical fetal dataset. Results of Wilcoxon signed rank test are presented, $*$: $p<0.05$, $**$: $p<0.01$, and n.s.: not significant ($p\geq 0.05$).

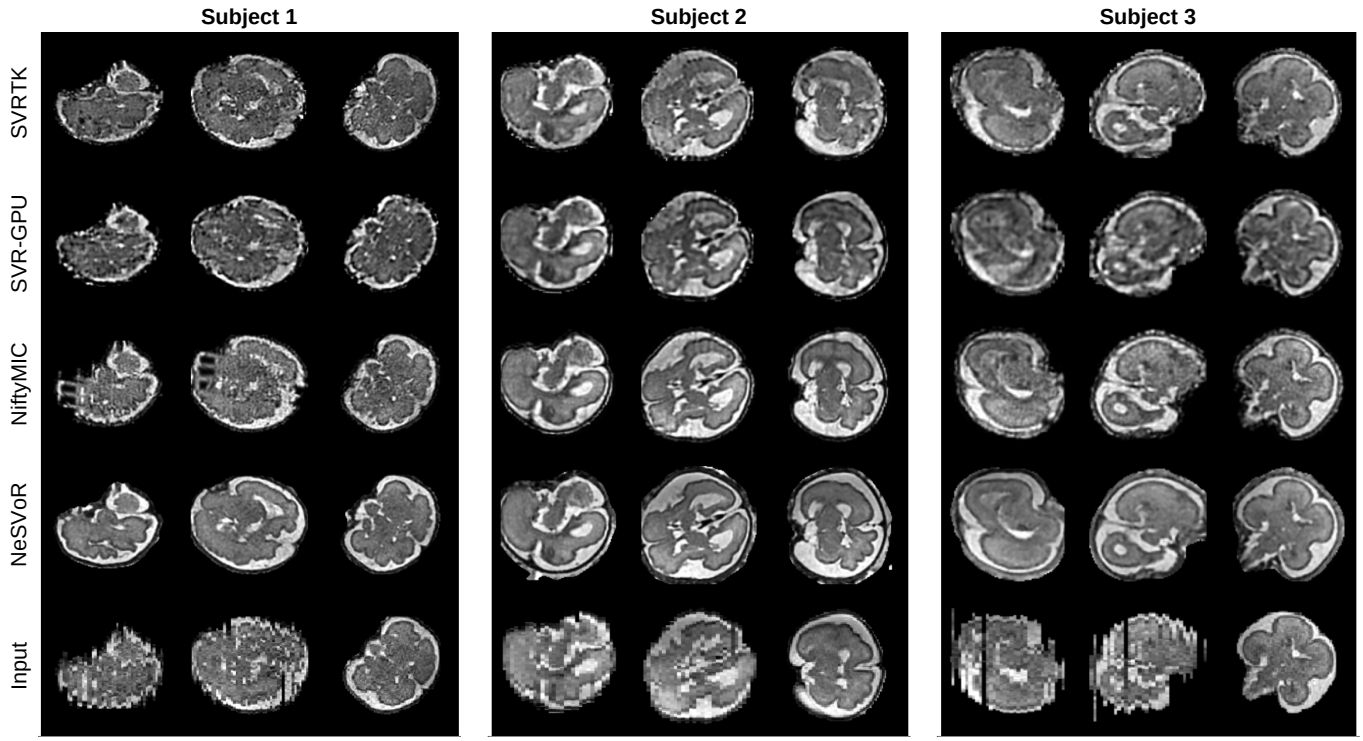

Fig. 8. The reconstruction results of three challenging cases in the clinical fetal brain dataset. For each reconstructed volume, three orthogonal views are displayed. Rows 1 to 4 show the results of different methods and the last row shows one of the input stacks for each case.

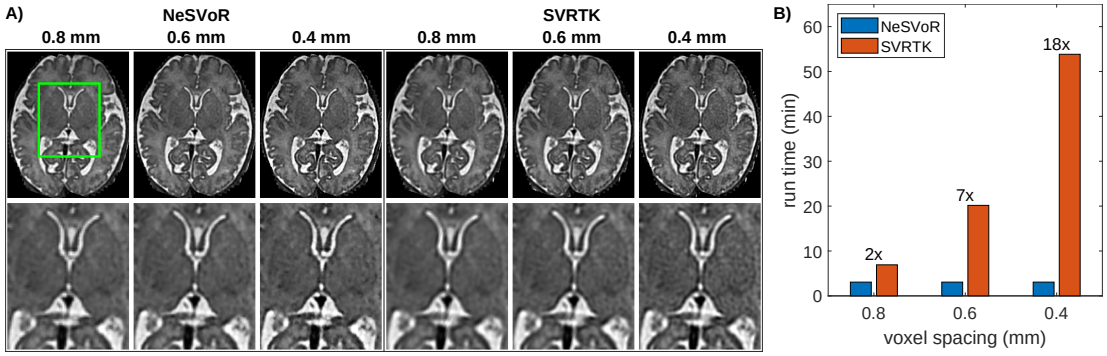

Fig. 9. The reconstruction results of a fetal brain at different resolutions. A) The reconstruction results of NeSVoR and SVRTK, where different columns show volumes reconstructed with different voxel spacings. B) The bar plot of the run time for the two methods. The numbers shown on top of the bars indicate the speedup of NeSVoR compared to SVRTK.

more time to converge and results tend to be smoother than that of hash grid encoding.

H. Hyperparameters

In this section, we study the impact of different hyperparameters on the performance of NeSVoR. Fig. 16 shows the PSNR and runtime of NeSVoR per hyperparameter setting,

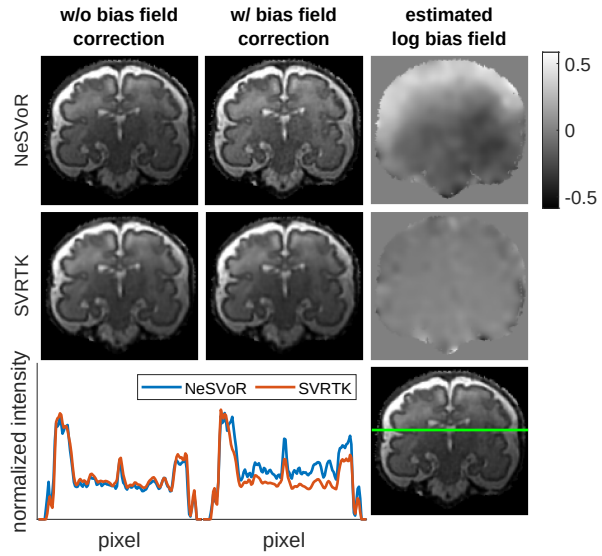

Fig. 10. Results of bias field correction. Row 1: reconstruction results of NeSVoR. Row 2: reconstruction results of SVRTK. Row 3: intensity profiles without bias field correction (bottom left) and with bias field correction (bottom middle) along a selected line (bottom right).

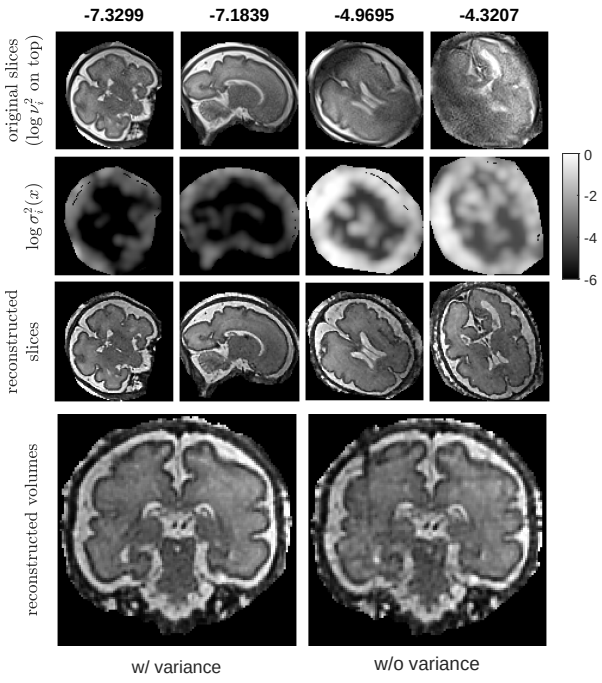

Fig. 11. Examples of estimated slice variances and variance maps in NeSVoR. Each column shows a different slice from the same subject. The numbers on the top are the log slice variances, $\log \boldsymbol{\nu_i^2}$. Row 1: the input slices. Row 2: the log variance maps, $\log \boldsymbol{\sigma_i^2}(\boldsymbol{x})$. Row 3: the corresponding slices sampled from the reconstructed volume. Row 4: the volume reconstructed with and without variance in the model.

where we varied one hyperparameter at a time.

The PSNR increases with the size of slice embeddings, since it could potentially encode more slice-specific information. The gain starts to saturate after 16, while the run time increases faster.

The number of hidden layers has a minimal impact on the PSNR. This result is consistent with the previous work [20]

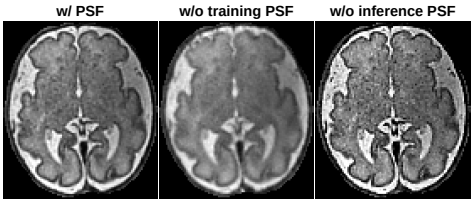

Fig. 12. The reconstruction results of a fetal subject with PSF model (left), without PSF during training (middle), and without PSF during inference (right).

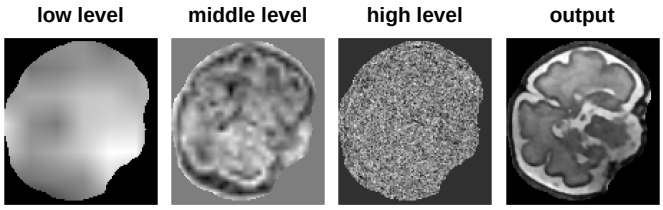

Fig. 13. Visualization of the learned hash grids at different levels

and indicates that most of the information of the volume is encoded in the hash grid.

The scale factor s of the hash grid determines how the size of the grid increases per level. When s is too small, the grid size is not enough to encode high-frequency details in the images. On the other hand, if s is too large, the grid size increases too fast while the actual size of the hash table is fixed, leading to severe hash collisions.

Fixing the factor s, the PSNR first increases with the number of levels, as more and more features are encoded. It also saturates after a threshold, which indicates the resolution at the highest level is finer than the finest detail in the data.

In NeSVoR, we propose a method to impose image regularization using sampling. To demonstrate the efficacy of the regularization, we reconstructed a fetal brain with different weights of regularization λ_V. As expected, the reconstructed volume becomes smoother as λ_V increases.

For slice-to-volume reconstruction, multiple stacks of slices of different orientations are acquired to oversample the brain ROI and the number of input stacks would affect the quality of reconstruction. We collected 10 stacks of slices of a subject (2 axial, 4 sagittal, and 4 coronal) and used different subsets of data to reconstruct the brain volume with NeSVoR. Fixing the number of input stacks, the setting that contains more different orientations yields better reconstruction results (*e.g.*, 4S vs. 2S+2C). Moreover, increasing the number of stacks per orientation can further increase the reconstruction quality (*e.g.*, 1A+1S+1C vs 2A+2S+2C).

I. Incorporating Deep Initializer

Slice-to-volume reconstruction is vulnerable to subject motion, since the slice transformations are optimized by maximizing the local slice-to-volume consistency, leading to a limited capture range of subject motion. Fig. 19 shows the reconstruction of a fetal subject (GA = 21 weeks) with 7 input stacks. The input data suffered from severe motion and the transformation optimization in NeSVoR failed to correct the slice misalignment in the data.

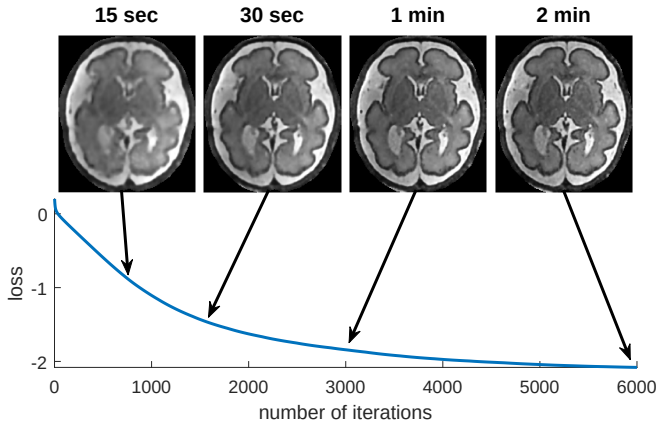

Fig. 14. The convergence of NeSVoR on the fetal brain data. The bottom row shows the training loss at each iteration. The top row shows results sampled from NeSVoR at different training times. The model converged to a high-quality volume after one minute.

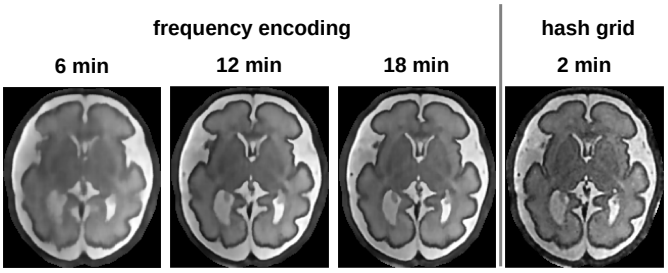

Fig. 15. Results of reconstruction with frequency encoding and hash grid encoding.

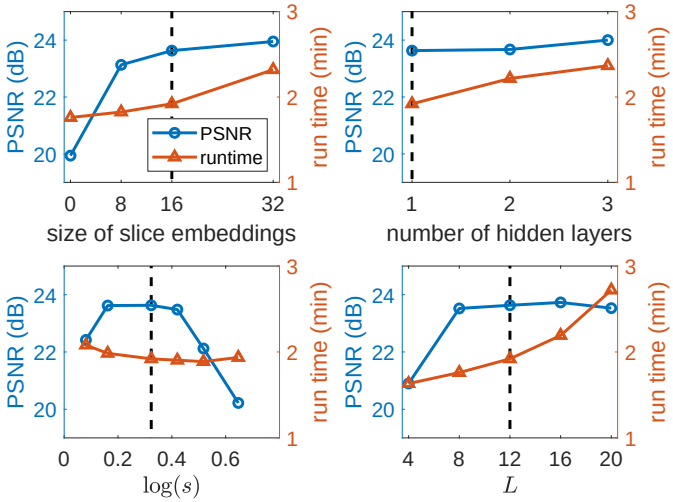

Fig. 16. PSNR and runtime of NeSVoR on the simulated fetal data per hyperparameter setting. In each experiment, we varied one hyperparameter (size of slice embeddings, number of hidden layers, scaling factor of hash grid s, and number of levels in hash grid L), and used the default hyperparameters for the rest (labeled with black dash lines in the figures).

Recently, many methods were proposed to address this problem by formulating the slice-to-volume registration as a learning problem [40]–[42], where deep neural networks are trained to predict the 3D location of each input slice. By learning from a large dataset in a supervised manner, these approaches are able to identify and correct large motions in fetal MRI. The predicted slice transformations can be used to initialize downstream reconstruction algorithms to improve the robustness in presence of motion.

To demonstrate the potential of combining NeSVoR with learning based slice-to-volume registration methods, we adopted the Slice-to-Volume Registration Transformer (SVoRT) [42] to predict the slice transformation of the data in Fig. 19, and used them to initialize NeSVoR. The reconstruction results show that, with the SVoRT initialization, NeSVoR is able to restore the correct 3D anatomical structures from the 2D slices even though they are corrupted by extreme subject motion during the scan. Fig. 20 visualizes the output transformations of NeSVoR, where three input slices from different stacks are placed at the corresponding output locations. NeSVoR alone cannot correct the large movement in the data, leading to a local minimum with significant slice misalignment. With the help of SVoRT, however, NeSVoR yields results with better consistency between different slices.

IV. Discussion and Conclusion

We have presented NeSVoR, a novel approach for fast, robust slice-to-volume reconstruction based on implicit neural representation. We adopt a continuous representation for the underlying volume and the slice acquisition model, which is resolution-agnostic and efficient for reconstructing volumes at high resolution. We also introduce a probabilistic noise model for outlier removal in reconstruction. Extensive evaluations on both simulated and clinical data show that NeSVoR produces high-quality results that are robust to subject motion, bias fields, and artifacts while achieving a significant speedup over traditional SVR methods. The proposed implementation can reconstruct a high-quality fetal brain volume in about a minute (Fig. 14), and potentially enables online reconstruction of fetal MRI during scans, which can be combined with online image quality assessment [43] and fetal brain tracking [44] to implement a fully automated pipeline for fetal MRI. Also, for an input dataset with 9 stacks (309 slices), the peak GPU memory usage of NeSVoR is only 832MB. Therefore, it can also be run on a GPU with less RAM.

It is noteworthy that the current model only focuses on the rigid motion for brain MRI. For applications with larger ROIs, such as fetal body and placenta reconstruction, a deformable motion model should be employed [45], [46]. Moreover, as the slice transformations are optimized with gradient descent, NeSVoRis only able to recover relatively limited transformations of the target object. To this end, we further demonstrate that deep learning based slice-to-volume registration methods, *e.g.*, SVoRT [42], can be used to initialize slice transformations and improve the robustness of NeSVoR in presence of large motion. Also, PSFs beyond the Gaussian function, such as the sinc kernel, will be explored in the future. The developed formulation of INR-based reconstruction is general and well suited to other reconstruction problems that involve a PSF model. For future work, we plan to extend NeSVoR to other types of MR acquisitions and even other modalities.

Taken together, NeSVoR provides a rather general framework for exploiting implicit neural representation in slice-to-

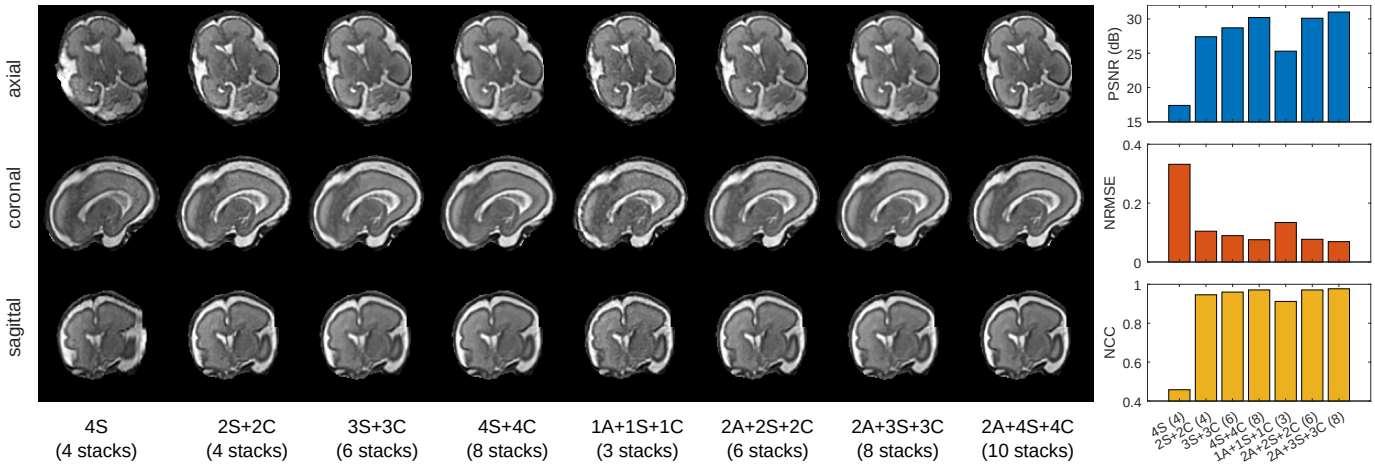

Fig. 17. The reconstruction results and quantitative metrics for a subject using different numbers of input stacks. We use the result with 10 input stacks as reference. A, S, and C mean axial, sagittal, and coronal respectively. For instance, 2A+4S+4C means the input stacks consist of two axial stacks, three sagittal stacks, and three coronal stacks.

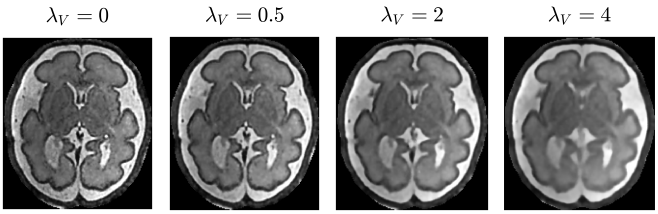

Fig. 18. The reconstruction results of a subject from the clinical fetal dataset with different weights for the image regularization term.

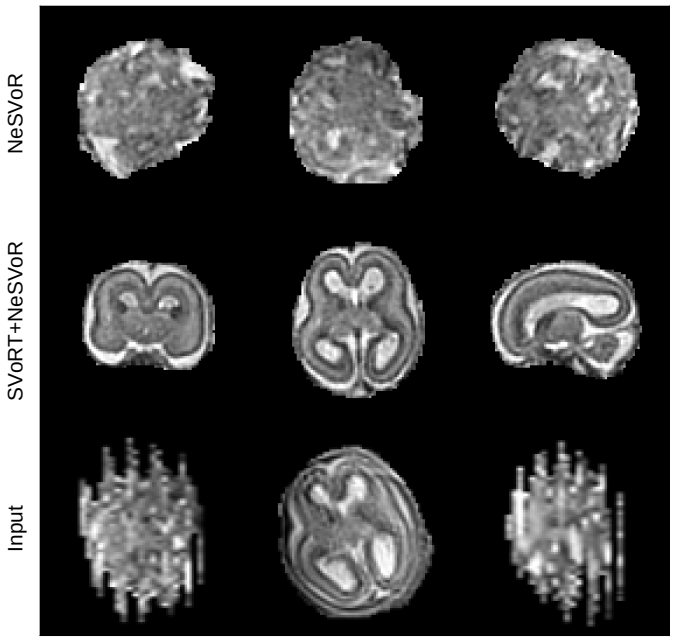

Fig. 19. The reconstruction results of NeSVoR with and without SVoRT initialization on a fetal subject with severe motion. The last row shows one of the input stacks that is corrupted by fetal motion.

volume reconstruction, which is potentially applicable to a broader range of reconstruction problems in medical imaging.

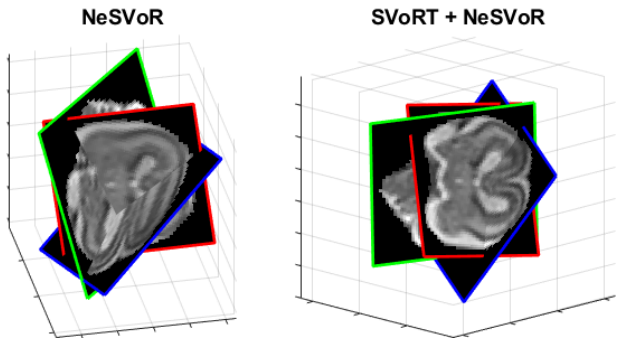

Fig. 20. Visualization of output transformations of NeSVoR with and without SVoRT initialization. Three input slices from different stacks are placed in the 3D space according to the corresponding output locations.

Acknowledgment

This research was supported by NIH R01EB032708, R01HD100009, NIBIB NAC P41EB015902, U01HD087211, R01EB017337, R01AG070988, RF1MH123195, ERC Starting Grant 677697, and ARUK-IRG2019A-003.

References

[1] S. N. Saleem, "Fetal MRI: An approach to practice: A review," *Journal of advanced research*, vol. 5, no. 5, pp. 507–523, 2014.

[2] F. Rousseau, O. A. Glenn, B. Iordanova, C. Rodriguez-Carranza, D. B. Vigneron, J. A. Barkovich, and C. Studholme, "Registration-based approach for reconstruction of high-resolution in utero fetal MR brain images," *Academic radiology*, vol. 13, no. 9, pp. 1072–1081, 2006.

[3] A. Gholipour, J. A. Estroff, and S. K. Warfield, "Robust super-resolution volume reconstruction from slice acquisitions: application to fetal brain MRI," *IEEE transactions on medical imaging*, vol. 29, no. 10, pp. 1739–1758, 2010.

[4] M. Kuklisova-Murgasova, G. Quaghebeur, M. A. Rutherford, J. V. Hajnal, and J. A. Schnabel, "Reconstruction of fetal brain MRI with intensity matching and complete outlier removal," *Medical image analysis*, vol. 16, no. 8, pp. 1550–1564, 2012.

[5] S. Jiang, H. Xue, A. Glover, M. Rutherford, D. Rueckert, and J. V. Hajnal, "MRI of moving subjects using multislice snapshot images with volume reconstruction (SVR): application to fetal, neonatal, and adult brain studies," *IEEE transactions on medical imaging*, vol. 26, no. 7, pp. 967–980, 2007.

[6] F. Odille, A. Bustin, B. Chen, P.-A. Vuissoz, and J. Felblinger, "Motion-corrected, super-resolution reconstruction for high-resolution 3D cardiac cine MRI," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 435–442, Springer, 2015.

[7] B. Marami, B. Scherrer, O. Afacan, B. Erem, S. K. Warfield, and A. Gholipour, "Motion-robust diffusion-weighted brain MRI reconstruction through slice-level registration-based motion tracking," *IEEE transactions on medical imaging*, vol. 35, no. 10, pp. 2258–2269, 2016.

[8] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *European conference on computer vision*, pp. 405–421, Springer, 2020.

[9] V. Sitzmann, M. Zollhöfer, and G. Wetzstein, "Scene representation networks: Continuous 3d-structure-aware neural scene representations," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[10] K. Zhang, G. Riegler, N. Snavely, and V. Koltun, "NeRF++: Analyzing and improving neural radiance fields," *arXiv preprint arXiv:2010.07492*, 2020.

[11] K. Kim, P. A. Habas, F. Rousseau, O. A. Glenn, A. J. Barkovich, and C. Studholme, "Intersection based motion correction of multislice MRI for 3-D in utero fetal brain image formation," *IEEE transactions on medical imaging*, vol. 29, no. 1, pp. 146–158, 2009.

[12] S. Tourbier, X. Bresson, P. Hagmann, J.-P. Thiran, R. Meuli, and M. B. Cuadra, "An efficient total variation algorithm for super-resolution in fetal brain MRI with adaptive regularization," *NeuroImage*, vol. 118, pp. 584–597, 2015.

[13] B. Kainz, M. Steinberger, W. Wein, M. Kuklisova-Murgasova, C. Malamateniou, K. Keraudren, T. Torsney-Weir, M. Rutherford, P. Aljabar, J. V. Hajnal, *et al.*, "Fast volume reconstruction from motion corrupted stacks of 2D slices," *IEEE transactions on medical imaging*, vol. 34, no. 9, pp. 1901–1913, 2015.

[14] M. Ebner, G. Wang, W. Li, M. Aertsen, P. A. Patel, R. Aughwane, A. Melbourne, T. Doel, S. Dymarkowski, *et al.*, "An automated framework for localization, segmentation and super-resolution reconstruction of fetal brain MRI," *NeuroImage*, vol. 206, p. 116324, 2020.

[15] Z. Wang, S. Wu, W. Xie, M. Chen, and V. A. Prisacariu, "NeRF--: Neural radiance fields without known camera parameters," *arXiv preprint arXiv:2102.07064*, 2021.

[16] R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth, "NeRF in the wild: Neural radiance fields for unconstrained photo collections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7210–7219, 2021.

[17] Q. Wu, Y. Li, L. Xu, R. Feng, H. Wei, Q. Yang, B. Yu, X. Liu, J. Yu, and Y. Zhang, "IREM: High-resolution magnetic resonance image reconstruction via implicit neural representation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 65–74, Springer, 2021.

[18] P.-H. Yeung, L. Hesse, M. Aliasi, M. Haak, W. Xie, A. I. Namburete, *et al.*, "Implicitvol: Sensorless 3D ultrasound reconstruction with deep implicit representation," *arXiv preprint arXiv:2109.12108*, 2021.

[19] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng, "Fourier features let networks learn high frequency functions in low dimensional domains," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7537–7547, 2020.

[20] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *arXiv preprint arXiv:2201.05989*, 2022.

[21] R. Chabra, J. E. Lenssen, E. Ilg, T. Schmidt, J. Straub, S. Lovegrove, and R. Newcombe, "Deep local shapes: Learning local sdf priors for detailed 3d reconstruction," in *European Conference on Computer Vision*, pp. 608–625, Springer, 2020.

[22] P. Bojanowski, A. Joulin, D. Lopez-Paz, and A. Szlam, "Optimizing the latent space of generative networks," *arXiv preprint arXiv:1707.05776*, 2017.

[23] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9446–9454, 2018.

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

[25] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[26] T. Müller, F. Rousselle, J. Novák, and A. Keller, "Real-time neural radiance caching for path tracing," *arXiv preprint arXiv:2106.12372*, 2021.

[27] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, *et al.*, "Mixed precision training," *arXiv preprint arXiv:1710.03740*, 2017.

[28] D. C. Van Essen, S. M. Smith, D. M. Barch, T. E. Behrens, E. Yacoub, K. Ugurbil, W.-M. H. Consortium, *et al.*, "The WU-Minn human connectome project: an overview," *Neuroimage*, vol. 80, pp. 62–79, 2013.

[29] F. Pérez-García, R. Sparks, and S. Ourselin, "TorchIO: a Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning," *Computer Methods and Programs in Biomedicine*, vol. 208, p. 106236, 2021.

[30] H. Gudbjartsson and S. Patz, "The Rician distribution of noisy MRI data," *Magnetic resonance in medicine*, vol. 34, no. 6, pp. 910–914, 1995.

[31] K. Payette, P. de Dumast, H. Kebiri, I. Ezhov, J. C. Paetzold, S. Shit, A. Iqbal, R. Khan, R. Kottke, P. Grehten, *et al.*, "An automatic multi-tissue human fetal brain segmentation benchmark using the fetal tissue annotation dataset," *Scientific Data*, vol. 8, no. 1, pp. 1–14, 2021.

[32] J. Xu, E. Abaci Turk, P. E. Grant, P. Golland, and E. Adalsteinsson, "STRESS: Super-resolution for dynamic fetal MRI using self-supervised learning," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 197–206, Springer, 2021.

[33] J. Xu, M. Zhang, E. A. Turk, L. Zhang, P. E. Grant, K. Ying, P. Golland, and E. Adalsteinsson, "Fetal pose estimation in volumetric mri using a 3d convolution neural network," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 403–410, Springer, 2019.

[34] E. J. Hughes, T. Winchman, F. Padormo, R. Teixeira, J. Wurie, M. Sharma, M. Fox, J. Hutter, L. Cordero-Grande, A. N. Price, *et al.*, "A dedicated neonatal brain imaging system," *Magnetic resonance in medicine*, vol. 78, no. 2, pp. 794–804, 2017.

[35] M. Ranzini, L. Fidon, S. Ourselin, M. Modat, and T. Vercauteren, "MONAIfbs: MONAI-based fetal brain MRI deep learning segmentation," *arXiv preprint arXiv:2103.13314*, 2021.

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

[37] J. Xu, S. Lala, B. Gagoski, E. Abaci Turk, P. E. Grant, P. Golland, and E. Adalsteinsson, "Semi-supervised learning for fetal brain MRI quality assessment with ROI consistency," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 386–395, Springer, 2020.

[38] Y. Sui, O. Afacan, C. Jaimes, A. Gholipour, and S. K. Warfield, "Scan-specific generative neural network for mri super-resolution reconstruction," *IEEE Transactions on Medical Imaging*, 2022.

[39] D. H. Laidlaw, K. W. Fleischer, and A. H. Barr, "Partial-volume bayesian classification of material mixtures in mr volume data using voxel histograms," *IEEE transactions on medical imaging*, vol. 17, no. 1, pp. 74–86, 1998.

[40] B. Hou, B. Khanal, A. Alansary, S. McDonagh, A. Davidson, M. Rutherford, J. V. Hajnal, D. Rueckert, B. Glocker, and B. Kainz, "3-D reconstruction in canonical co-ordinate space from arbitrarily oriented 2-D images," *IEEE transactions on medical imaging*, vol. 37, no. 8, pp. 1737–1750, 2018.

[41] W. Shi, H. Xu, C. Sun, J. Sun, Y. Li, X. Xu, T. Zheng, Y. Zhang, G. Wang, and D. Wu, "Affirm: Affinity fusion-based framework for iteratively random motion correction of multi-slice fetal brain mri," *arXiv preprint arXiv:2205.05851*, 2022.

[42] J. Xu, D. Moyer, P. E. Grant, P. Golland, J. E. Iglesias, and E. Adalsteinsson, "SVoRT: Iterative transformer for slice-to-volume registration in fetal brain MRI," in *Medical Image Computing and Computer Assisted Intervention - MICCAI 2022 - 25th International Conference, Singapore, September 18-22, 2022, Proceedings, Part VI* (L. Wang, Q. Dou, P. T. Fletcher, S. Speidel, and S. Li, eds.), vol. 13436 of *Lecture Notes in Computer Science*, pp. 3–13, Springer, 2022.

[43] B. Gagoski, J. Xu, P. Wighton, M. D. Tisdall, R. Frost, W.-C. Lo, P. Golland, A. van Der Kouwe, E. Adalsteinsson, and P. E. Grant, "Automated detection and reacquisition of motion-degraded images in fetal HASTE imaging at 3T," *Magnetic resonance in medicine*, vol. 87, no. 4, pp. 1914–1922, 2022.

[44] M. Hoffmann, E. Abaci Turk, B. Gagoski, L. Morgan, P. Wighton, M. D. Tisdall, M. Reuter, E. Adalsteinsson, P. E. Grant, L. L. Wald, *et al.*, "Rapid head-pose detection for automated slice prescription of fetal-brain MRI," *International journal of imaging systems and technology*, vol. 31, no. 3, pp. 1136–1154, 2021.

[45] A. Alansary, M. Rajchl, S. G. McDonagh, M. Murgasova, M. Damodaram, D. F. Lloyd, A. Davidson, M. Rutherford, J. V.

Hajnal, D. Rueckert, *et al.*, "PVR: patch-to-volume reconstruction for large area motion correction of fetal MRI," *IEEE transactions on medical imaging*, vol. 36, no. 10, pp. 2031–2044, 2017.

[46] A. Uus, T. Zhang, L. H. Jackson, T. A. Roberts, M. A. Rutherford, J. V. Hajnal, and M. Deprez, "Deformable slice-to-volume registration for motion correction of fetal body and placenta MRI," *IEEE transactions on medical imaging*, vol. 39, no. 9, pp. 2750–2759, 2020.