

RESEARCH ARTICLE

Magnetic Resonance in Medicine

SPICER: Self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction

Yuyang Hu¹  | Weijie Gan²  | Chunwei Ying³  | Tongyao Wang⁴ | Cihat Eldeniz³  | Jiaming Liu¹ | Yasheng Chen⁵ | Hongyu An^{1,3,4,5}  | Ulugbek S. Kamilov^{1,2}  ¹Department of Electrical and Systems Engineering, Washington University in St. Louis, St. Louis, Missouri²Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, Missouri³Mallinckrodt Institute of Radiology, Washington University in St. Louis, St. Louis, Missouri⁴Department of Biomedical Engineering, Washington University in St. Louis, St. Louis, Missouri⁵Department of Neurology, Washington University in St. Louis, St. Louis, Missouri

Correspondence

Ulugbek S. Kamilov, Department of Electrical and Systems Engineering, Washington University in St. Louis, One Brookings Drive MSC 1045-213-1010J, St. Louis, MO 63130, USA.
Email: kamilov@wustl.com

Funding information

National Science Foundation, Grant/Award Number: CCF-2043134; National Institutes of Health, Grant/Award Numbers: NIH R01 EB032713, NIH R01HL129241, R21NS127425, RF1NS116565

Abstract

Purpose: To introduce a novel deep model-based architecture (DMBA), SPICER, that uses pairs of noisy and undersampled k-space measurements of the same object to jointly train a model for MRI reconstruction and automatic coil sensitivity estimation.**Methods:** SPICER consists of two modules to simultaneously reconstructs accurate MR images and estimates high-quality coil sensitivity maps (CSMs). The first module, CSM estimation module, uses a convolutional neural network (CNN) to estimate CSMs from the raw measurements. The second module, DMBA-based MRI reconstruction module, forms reconstructed images from the input measurements and the estimated CSMs using both the physical measurement model and learned CNN prior. With the benefit of our self-supervised learning strategy, SPICER can be efficiently trained without any fully sampled reference data.**Results:** We validate SPICER on both open-access datasets and experimentally collected data, showing that it can achieve state-of-the-art performance in highly accelerated data acquisition settings (up to 10×). Our results also highlight the importance of different modules of SPICER—including the DMBA, the CSM estimation, and the SPICER training loss—on the final performance of the method. Moreover, SPICER can estimate better CSMs than pre-estimation methods especially when the ACS data is limited.**Conclusion:** Despite being trained on noisy undersampled data, SPICER can reconstruct high-quality images and CSMs in highly undersampled settings, which outperforms other self-supervised learning methods and matches the performance of the well-known E2E-VarNet trained on fully sampled ground-truth data.

KEYWORDS

coil sensitivity estimation, deep learning, image reconstruction, inverse problems, parallel MRI

Yuyang Hu and Weijie Gan have contributed equally to the work.

1 | INTRODUCTION

MRI is a medical imaging technology known to suffer from slow data acquisition. Parallel MRI (PMRI) is a widely used acceleration strategy that relies on the spatial encoding provided by multiple receiver coils to reduce the amount of data to acquire.¹⁻⁴ The multicoil under-sampled data can be reconstructed by fitting the missing k-space lines^{1,2} or in the image space using coil sensitivity maps (CSMs).^{3,4} Compressed sensing (CS) is a complementary technique used to further accelerate data collection by using prior knowledge on the unknown image (sparsity, low-rankness).^{5,6}

Deep learning (DL) has recently emerged as a promising paradigm for image reconstruction in CS-PMRI.⁷⁻⁹ Traditional DL methods train convolutional neural networks (CNNs) to map acquired measurements to the desired images.^{10,11} Recent work has shown that deep model-based architectures (DMBAs) can perform better than generic CNNs by accounting for the measurement model of the parallel imaging system.¹²⁻¹⁷ Most of these methods require precalibrated CSMs as an important element in their model. However, CSM pre-estimation strategies rely on sufficient auto-calibration signal (ACS) lines, which limits the acceleration rates for data acquisition. To address this limitation, recent work has proposed to jointly estimate high-quality images and CSMs in an end-to-end manner.¹⁸⁻²⁰ However, these methods still require fully sampled ground-truth images as training targets, which limits their applicability to settings where ground-truth is difficult to obtain or unavailable. On the other hand, there has also been a broad interest in developing self-supervised DL methods that rely exclusively on the information available in the undersampled measurements.^{17,21-26}

Despite the rich literature on DMBAs and self-supervised DL, the existing work in the area has not investigated joint image reconstruction and coil sensitivity estimation directly from noisy and undersampled data. We bridge this gap by presenting Self-Supervised Learning for MRI with Automatic Coil Sensitivity Estimation (SPICER) as a new self-supervised learning framework for parallel MRI that is equipped with an automatic CSM estimator. SPICER is a synergistic combination of a powerful model-based architecture and a flexible self-supervised training scheme. The SPICER architecture consists of two branches: (a) a CNN for estimating CSMs from possibly limited ACS data, while ensuring physically realistic predictions; (b) a DMBA that uses the estimated CSMs for high-quality image reconstruction. The SPICER training is performed using undersampled and noisy measurements without any fully sampled ground-truth. For training, SPICER necessitates at least one pair of undersampled

and noisy measurements from each slice. We extensively validated SPICER on in vivo MRI data for several acceleration factors. Our results show that SPICER can achieve state-of-the-art performance on PMRI at high acceleration rates (up to 10×). Moreover, SPICER can estimate better CSMs than pre-estimation methods especially when the ACS data is limited.

This paper extends the preliminary work presented in the workshop paper.²⁷ Compared to the method in Reference 27, SPICER uses a different forward model, model-based DL architecture, and training loss function. This paper also provides an expanded discussion on related work, additional technical details, as well as completely new numerical results using real in vivo MRI data acquired using a 32-channel coil.

2 | THEORY

2.1 | Problem formulation

Consider the following CS-PMRI measurement model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}, \quad (1)$$

where $\mathbf{x} \in \mathbb{C}^n$ is an unknown image, $\mathbf{y} = (\mathbf{y}_1, \dots, \mathbf{y}_{n_c})$ are the multicoil measurements from $n_c \geq 1$ coils, $\mathbf{e} = (\mathbf{e}_1, \dots, \mathbf{e}_{n_c})$ is the noise vector, and $\mathbf{A} = (\mathbf{A}_1, \dots, \mathbf{A}_{n_c})$ is the measurement operator (or forward operator). The measurement model for each coil can be represented as

$$\mathbf{y}_k = \underbrace{\mathbf{P}\mathbf{F}\mathbf{S}_k}_{\mathbf{A}_k} \mathbf{x} + \mathbf{e}_k, \quad k = 1, 2, \dots, n_c, \quad (2)$$

where $\mathbf{S}_k \in \mathbb{C}^{n \times n}$ is the CSM of the k th coil, $\mathbf{F} \in \mathbb{C}^{n \times n}$ is the Fourier transform operator, $\mathbf{P} \in \mathbb{C}^{n \times n}$ is the k-space sampling operator, and $\mathbf{e} \in \mathbb{C}^n$ is the noise vector. Note that $\mathbf{S} = (\mathbf{S}_1, \dots, \mathbf{S}_{n_c})$ varies for each scan, since it depends on the relative location of the coils with the object being imaged. When $\mathbf{S}$ are known, image reconstruction can be formulated as regularized optimization

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} f(\mathbf{x}) \quad \text{with} \quad f(\mathbf{x}) = g(\mathbf{x}) + h(\mathbf{x}), \quad (3)$$

where g is the data fidelity term that quantifies consistency with the observed data $\mathbf{y}$ and h is a regularizer that infuses prior knowledge on $\mathbf{x}$. Examples of g and h used in CS-PMRI are the least-squares and *total variation* (TV) functions²⁸

$$g(\mathbf{x}) = \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 \quad \text{and} \quad h(\mathbf{x}) = \tau \|\mathbf{D}\mathbf{x}\|_1, \quad (4)$$

where $\mathbf{D}$ denotes the image gradient and $\tau > 0$ is a regularization parameter.

DL has recently gained popularity in MRI image reconstruction due to its excellent empirical performance.^{7,8,29} Traditional DL methods are based on training CNNs (such as U-Net³⁰) to map the corrupted images^{31,32} or the under-sampled measurements^{11,33} to their desired fully sampled ground-truth versions. There is also growing interest in DMBA that can combine physical measurement models and learned CNN priors. Well known examples of DMBA are plug-and-play priors,^{34,35} regularized by denoising,³⁶ and deep unfolding (DU).^{15,16,37} In particular, DU has gained considerable recognition due to its ability to achieve the state-of-the-art performance, while providing robustness to changes in data acquisition. DU architectures are typically obtained by unfolding iterations of an image reconstruction algorithm as layers, representing the regularizer within image reconstruction as a CNN, and training the resulting network end-to-end. Different DU architectures can be obtained by using various optimization/reconstruction algorithms. In this paper, we will rely on a DU variant of the regularized by denoising model as the basis of our image reconstruction method.³⁸

2.2 | Reconstruction using precalibrated CSMs

There are two widely-used image formation approaches in CS-PMRI (see recent review⁸): (a) reconstruction in the k-space domain and (b) reconstruction in the image domain. GRAPPA² is a well-known example of (a) that fills in unacquired k-space values by linearly interpolating acquired neighboring k-space samples. Recent work³⁹ extends GRAPPA by using a CNN to learn a non-linear interpolator in k-space. SENSE³ and ESPIRiT⁴ are two well-known examples of (b) that first precalibrate CSMs and then use it to solve the inverse problem (1). Our work in this paper adopts strategy (b), which will be the focus of the subsequent discussion.

Pre-estimated CSMs can either be obtained by doing a separate calibration scan⁴⁰ or estimated directly from the ACS region of the undersampled measurements. The drawback of the former approach is that it extends the total scan time. ESPIRiT⁴ is based on the latter approach. There are several issues and challenges with the pre-estimated approaches.^{40,41} One issue is that the inconsistencies between the calibration scan and the accelerated scan can result in imaging artifacts. Another issue is that estimating CSMs from a small number of ACS lines may not be sufficiently accurate. DeepSENSE,⁴² a recent supervised DL method, uses a CNN to learn a mapping from the ACS data to CSM references, obtained by dividing the

fully sampled individual coil images by the sum-of-squares reconstruction from the fully sampled measurements. While DeepSENSE improves over methods based on pre-estimating CSMs, especially when the ACS data is limited, DeepSENSE still requires fully-sampled data to generate training CSMs.

2.3 | Joint reconstruction and CSM estimation

Traditionally, optimization-based methods for joint image reconstruction and CSM estimation treat $\mathbf{S}$ as another unknown variable in (3) and alternate between updating the image and updating the coil sensitivities.^{40,41} DU has recently been adopted to perform joint estimation of image and CSMs without any precalibration procedure.^{18–20,43–46} The concept behind these methods is to model CSM estimated as a trainable DNN module that can be optimized simultaneously with other learnable parameters in the deep network. The inputs to the CSM estimated modules could be the original undersampled measurements²⁰ or the intermediate results available at different layers of the deep unfolded networks.^{18,19} However, these joint learning methods rely on fully-sampled ground-truth images as training targets, which limits their applicability when ground-truth data is not available. Our work contributes to this area by investigating a self-supervised learning approach for joint image reconstruction and CSM calibration that requires no fully sampled ground-truth data.

2.4 | Self-supervised image reconstruction

There is a growing interest in DL-based image reconstruction to reduce the dependence of training on high-quality ground-truth data (see recent reviews^{47–49}). Some well-known strategies include Noise2Noise (N2N),²³ Noise2Void,⁵⁰ deep image prior,⁵¹ Compressive Sensing using Generative Models,⁵² and equivariant imaging.⁵³ This work is most related to N2N, where a DNN R_θ is trained on a set of noisy images $\{\hat{\mathbf{x}}_{i,j} = \mathbf{x}_i + \mathbf{e}_{i,j}\}$ with j indexing different realizations of the same underlying image i . The N2N learning is formulated as follows

$$\hat{\theta} = \arg \min_{\theta} \sum_i \sum_{j \neq j'} \left\| f_{\theta}(\mathbf{y}_{i,j}) - \mathbf{y}_{i,j'} \right\|_2^2, \quad (5)$$

where f_{θ} denotes a CNN with θ being trainable parameters. N2N and its extensions have been investigated in several papers on PMRI reconstruction.^{24–26} For example, DURED-Net⁵⁴ proposes an unsupervised learning method

for MRI image reconstruction combining an unsupervised denoising network and a plug-and-play priors method. SSDU and NLINV-Net are related self-supervised DL methods for MRI reconstruction that use a similar approach of dividing each MRI acquisition into two subsets for training. However, they have distinct strategies for CSM estimation. Specifically, SSDU relies on pre-calibrated CSMs, while NLINV-Net adopts a model-based approach for estimating CSMs. Our SPICER method is different from NLINV-Net in that it trains a CNN module to estimate CSMs within the proposed DMBA architecture, thus distinguishing it from the NLINV-Net's pure model-based approach.

While the concept of N2N enables the training of the DNN for PMRI without any fully sampled data, to the best of our knowledge, the prior work is based on using pre-estimated CSMs. Our SPICER method does not require prescan calibration, instead using the N2N framework for joint reconstruction and CSM estimation without any ground-truth. In addition, our CSMs are estimated using a learning based strategy.

3 | METHODS

3.1 | SPICER model

Our SPICER method takes multicoil undersampled measurements as its input and produces the reconstructed images and CSMs at its output. As illustrated in Figure 1, SPICER consists of two modules: (a) a CSM estimation

module that uses information extracted from the raw measurements, and (b) a DMBA-based MRI reconstruction module that forms reconstructed images from the input measurements and the estimated CSMs. Our training procedure uses a pair of multicoil undersampled measurement $\{(\mathbf{y}_i, \mathbf{y}'_i)\}_i$ which are acquired from the same object

$$\mathbf{y}_i = \mathbf{A}_i \mathbf{x}_i + \mathbf{e}_i \quad \text{and} \quad \mathbf{y}'_i = \mathbf{A}'_i \mathbf{x}_i + \mathbf{e}'_i, \quad (6)$$

where $(\mathbf{A}_i, \mathbf{A}'_i)$ and $(\mathbf{e}_i, \mathbf{e}'_i)$ denote distinct forward operators and noise vectors, respectively. The measurements $\mathbf{y}_i$ and $\mathbf{y}'_i$ can correspond to two subsets extracted from a single acquisition¹⁷ or two separate MRI acquisitions. Note that our training procedure does not require any ground-truth images or known CSMs.

3.2 | Coil sensitivity estimation Module

Let $\hat{\mathbf{y}}$ be an input measurement and $\mathbf{P}$ the corresponding sampling matrix. The coil sensitivity estimation module forms CSMs from multicoil measurements *without* a prescan calibration by performing three steps: (a) the ACS regions of the undersampled measurements are extracted, which are represented as $\mathbf{y}_{\text{ACS}}$; (b) $\mathbf{y}_{\text{ACS}}$ is mapped back to the image domain by applying the zero-filled inverse Fourier transform $\mathbf{p}^0 = \mathbf{F}^{-1}(\mathbf{y}_{\text{ACS}})$; (c) $\mathbf{p}^0$ is fed into a CNN $\mathbf{P}_\phi$ with trainable parameters $\phi \in \mathbb{R}^q$ to obtain estimated CSMs: $\hat{\mathbf{S}} = \mathbf{P}_\phi(\mathbf{p}^0)$. (d) Finally, the estimated sensitivity maps $\hat{\mathbf{S}}$ are normalized by dividing their root-square-of-sum to ensure that $\hat{\mathbf{S}}^{-1} \hat{\mathbf{S}} = \mathbf{I}$.

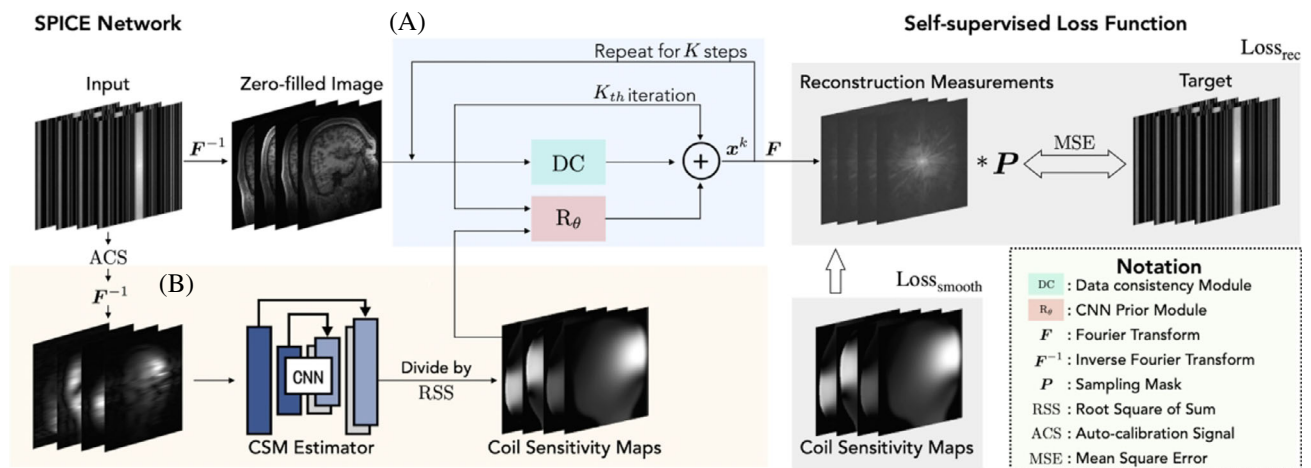


FIGURE 1 The self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER) method consists of a deep model-based architecture-based MRI reconstruction module and a coil sensitivity estimation module that map multicoil undersampled measurements to a single high-quality image and a set of coil sensitivity maps, respectively. The reconstruction module, described in (7), is composed of two key components, a data consistency module and a convolutional neural network prior. The network is trained directly on raw k-space measurements where the input and the target measurement correspond to a pair of undersampled measurements from the same object. (A) DMBA MRI reconstruction module; (B) CSM estimation module.

3.3 | Image reconstruction module

The image reconstruction module of SPICER incorporates a DMBA, which is built upon regularized by denoising.³⁸ Figure 1 visually represents the DMBA MRI reconstruction module, highlighting two essential steps: the data consistency module and the CNN Prior module. The image reconstruction process is performed iteratively by integrating information from CNN R_θ with learnable parameters θ and imposing consistency between the predicted and the raw k-space measurements via ∇g in (8). Let $\hat{\mathbf{c}}^0 = \mathbf{F}^{-1}\hat{\mathbf{y}}$ represent the initial image, and $K \geq 1$ be the total number of DU layers, The steps of SPICER are given by

$$\mathbf{c}^{k+1} = \mathbf{c}^k - \gamma^k \left(\underbrace{\nabla g(\mathbf{c}^k, \mathbf{y})}_{\text{Data consistency}} + \tau^k \underbrace{\mathbf{S} \mathbf{R}_\theta^k(\mathbf{S}^{-1} \mathbf{c}^k)}_{\text{Regularization}} \right), \quad (7)$$

where τ^k and γ^k are learnable parameters, $\mathbf{c}^k$ are the intermediate *multicoil* images in the k th step, and

$$\nabla g(\mathbf{c}^k, \mathbf{y}) = \mathbf{F}^{-1} \mathbf{P}^{-1} (\mathbf{P} \mathbf{F} \mathbf{c}^k - \mathbf{y}). \quad (8)$$

The CNN R_θ^k in (7) takes a single image as an input, requiring $\mathbf{S}^{-1}$ and $\mathbf{S}$ for fusing multiple images into a single image and expanding a single image into multiple images, respectively. The final reconstructed image $\mathbf{x}$ can be obtained from the output of the last step as $\mathbf{x} = \mathbf{S}^{-1} \mathbf{c}^K$. It is worth noting that unlike existing DU methods that rely on pre-calibrated $\mathbf{S}$,^{14–17} SPICER jointly trains a $\mathbf{S}$ calibration network $\mathbf{P}_\phi$ simultaneously with the image reconstruction network. During inference, the pretrained $\mathbf{P}_\phi$ network predicts $\mathbf{S}$ as a preliminary step, which is then utilized by the image reconstruction network.

3.4 | Self-supervised training procedure

We use standard stochastic gradient method to jointly optimize $\{\theta_k\}_{k=1}^K$ and ϕ by minimizing the loss function

$$\text{Loss} = \text{Loss}_{\text{rec}} + \lambda \cdot \text{Loss}_{\text{smooth}}, \quad (9)$$

where λ is a regularization parameter.

Loss_{rec} seeks to map each $\mathbf{y}_i$ and the corresponding $\mathbf{y}'_i$ to each other. The key idea here is to map the reconstructed images back to the k-space domain by applying the forward operator of the training target. For example, one can map $\mathbf{x}_i$ back to the k-space domain by applying the forward operator $\mathbf{A}'_i$ then penalize the discrepancy between the resulting measurements $\mathbf{A}'_i \mathbf{x}_i$ and raw measurements $\mathbf{y}'_i$. Note that, forward operator $\mathbf{A}'_i$ uses the same sampling

mask $\mathbf{P}'$ employed in the acquisition of the undersampled raw measurement $\mathbf{y}'_i$. Here, the CSMs $\mathbf{S}_i$ in $\mathbf{A}_i$ are estimated by the coil sensitivity estimation module after feeding $\mathbf{y}_i$ as the input. The formulation of Loss_{rec} is

$$\text{Loss}_{\text{rec}} = \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{\text{rec}}(\mathbf{A}'_i \mathbf{x}_i, \mathbf{y}'_i) + \mathcal{L}_{\text{rec}}(\mathbf{A}_i \mathbf{x}'_i, \mathbf{y}_i), \quad (10)$$

where N is number of samples, $\mathbf{x}'_i$ is the reconstructed image when $\mathbf{y}'_i$ is the input measurement, and $\mathcal{L}_{\text{rec}}$ denotes the ℓ_2 -norm. During minimization, Loss_{rec} enforces the accuracy between the predicted and the raw measurements, but it can also generate nonsmooth CSMs that are not physically realistic and cause overfitting. Therefore, we include $\text{Loss}_{\text{smooth}}$, a smoothness regularization for CSMs, to impose smoothness within the field of view (FOV) region of estimated CSMs.

$$\text{Loss}_{\text{smooth}} = \frac{1}{N} \sum_{i=1}^N \left\| \mathbf{D}(\mathbf{S})_i^{\text{FOV}} \right\|_2^2, \quad (11)$$

where the $\mathbf{D}$ is the gradient of the CSMs $\mathbf{S}_i$. The combination of Loss_{rec} and $\text{Loss}_{\text{smooth}}$ can enable the optimization on the measurement domain and the stability of CSMs, thus it is named as a smoothness-enhanced measurement domain loss function.

3.5 | Implementation

In our experiment, all CNNs in SPICER are based on U-Net³⁰ and implemented using PyTorch.⁵⁵ To process complex-valued data, we reshape all the complex-valued images by splitting their real and imaginary parts and concatenating them into the channel (= 2) dimensions. The total number of unrolling layers K in (7) is 8 and the regularization parameter λ in (9) is 0.01. The initial values we used in (7) is $\gamma^0 = 1$ and $\tau^0 = 0.1$. We used Adam as the optimizer with a learning rate 0.001 for the initial 30 epochs and 0.0001 for the rest. We performed all our experiments on a machine equipped with an Intel Xeon Gold 6130 Processor and an NVIDIA GeForce RTX 3090 GPU. The implementation details can be found in: <https://github.com/wustl-cig/SPICER>.

3.6 | In vivo brain dataset

The data acquisition was performed on a Siemens 3T Prisma scanner (Siemens Healthcare) with a 32-channel head coil. Images were collected with sagittal T1 magnetization-prepared rapid gradient-echo sequence. The acquisition parameters were as follows: repetition

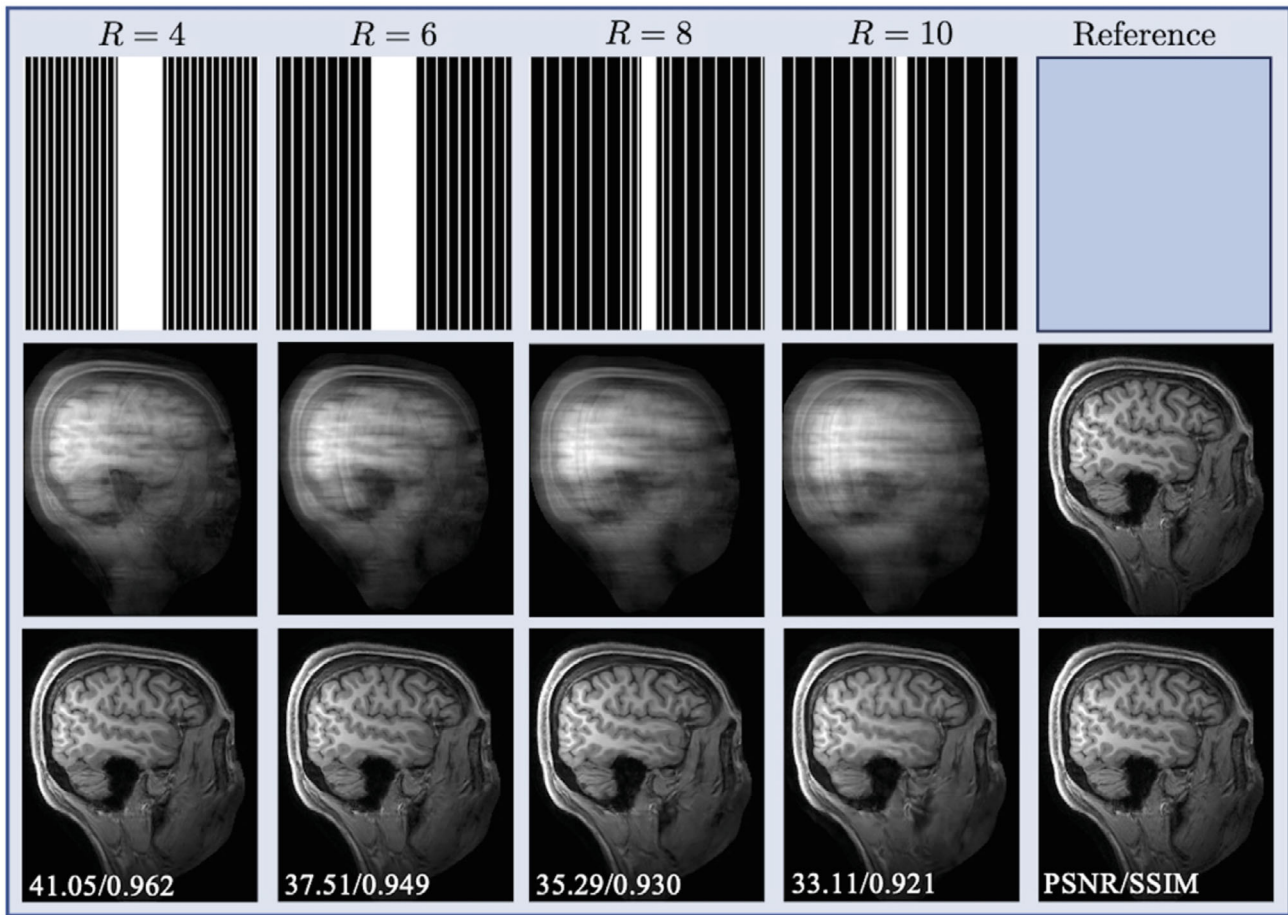


FIGURE 2 The leftmost four in the first row shows the cartesian equispaced sampling masks P used in the experimental validation: $R = 4, 6, 8, 10$, with corresponding auto-calibration signal (ACS) lines = 24, 24, 8, and 5. The second row shows the zero-filled images of the same slice for the four acceleration rates. The third row shows the self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER) reconstructed images of the same slice and include the peak signal-to-noise ratio (PSNR)/structural similarity index (SSIM) values with respect to the reference as shown in the rightmost column.

time = 2400 ms, echo time = 2.62 ms, inversion time = 1000 ms, flip angle = 8 degrees, field of view = 256 mm $\times$ 256 mm, voxel size = 1 $\times$ 1 $\times$ 1 mm³, slices per slab = 176, slice and phase resolution = 100% and slice and phase partial Fourier off. A 2 $\times$ oversampling was used in the frequency encoding direction, and the asymmetric echo was turned on to allow short echo time. Fully sampled measurements were acquired with GRAPPA turned off, and the total acquisition time was 10 min and 16 s. The reference images were obtained by the sum-of squares reconstruction computed on the fully sampled multicoil data. Upon the approval of our Institutional Review Board, we used brain MRI data from 14, 1, and 5 participants in this study for training, validation, and testing, respectively. To obtain undersampled measurements, the multicoil k-space data were retrospectively undersampled using one-dimensional (1D) Cartesian equispaced sampling masks with ACS lines. The sampling strategy is based on the clinical-used 2 $\times$ acceleration sampling pattern,

which is acquired with GRAPPA $R = 2$ in phase encoding direction with 24 ACS lines. As shown in Figure 2, we conducted our experiments for acceleration factors $R = 4, 6, 8$, and 10 with 1D equispaced Cartesian masks, that contain 24, 24, 8, and 5 ACS lines, respectively. They correspond to the retrospective sampling rates of 32%, 24%, 15%, and 12%. For each undersampling pattern, one pair of under-sampled measurements are generated for self-supervised training. We generate 1932 (138 $\times$ 14) pairs of training data for each undersampling rate using the entire training dataset. Each training data pair corresponds to one of 138 slices from each of the 14 subjects in the dataset.

3.7 | fastMRI Brain dataset

The T2 MR brain dataset was obtained from the multicoil fastMRI dataset. We used the T2 MR brain acquisitions of

165 subjects obtained from the multicoil dataset as the raw reference. These 165 subjects were split into 130, 15, and 20 for training, validation, and testing, respectively. For each subject, we extracted the first 12 slices on the transverse plane, containing the most relevant regions of the brain. In order to obtain undersampled measurements, we use the 1D Cartesian equispaced sampling pattern provided in the fastMRI database with acceleration rate = 4, 8.

3.8 | Comparisons

3.8.1 | Baseline methods

We selected several well-known methods as references to compare the performance of SPICER:

- *TV (with ESPIRiT)*: The traditional TV regularized image reconstruction.²⁸
- *GRAPPA*: Traditional GRAPPA¹ that linearly interpolates missing k-space points using nearby acquired k-space points from all coils.
- *U-Net*: The U-Net architecture³⁰ trained to maximize SSIM between the reconstructed image and the image obtained using GRAPPA on the same amount of measurements as SPICER.
- *SSDU*: A well-known self-supervised learning method¹⁷ that trains a DU network by dividing each k-space MRI acquisition into two subsets and using them as training targets for each other. For fair comparison, SSDU network use the same U-Net architecture as SPICER.
- *SSDU^{auto}*: A variant of SSDU¹⁷ that incorporates a same automatic CSM learning module as SPICER. The strategy to generate training data pairs is the same with SSDU.
- *SSDU**: A variant of SSDU¹⁷ that trained with paired training data and automatic CSM learning module as SPICER. The training data pairs are generated with Cartesian subsampling strategy (same as SPICER).
- *E2E-VarNet*: An idealized variant of method from²⁰ trained using SSIM on fully sampled reference data. E2E-VarNet shows the upper bound on performance achievable by the self-supervised variants of the method.

All the CSMs used in TV (with ESPIRiT), U-Net, and SSDU are pre-estimated using ESPIRiT.⁴ We observed that when the number of ACS lines is less than 24, reconstruction using the CSMs estimated by ESPIRiT are of poor quality. Therefore, we use 24 ACS lines for estimating CSMs with ESPIRiT to be used in TV, U-Net, and SSDU.

3.8.2 | Ablation study

We also performed an ablation study to highlight the influence of the DMBA module, the CSMs estimation module, and the CSMs smoothness regularization within SPICER. We compared the original SPICER model with three different models as follows:

- *Joint U-Net*: We replace the DMBA module of SPICER with U-Net to show the benefit of using a DMBA.
- *DMBA (ESPIRiT)*: We use the same DMBA as SPICER but use ESPIRiT on undersampled measurements to replace the CSM estimation module of SPICER.
- *SSDU**: SPICER trained without the CSMs smoothness regularization loss $\text{Loss}_{\text{smooth}}$ in (11).

3.8.3 | Evaluation of CSMs estimation

In order to evaluate the quality of the estimated CSMs, we use CSMs estimated by the proposed and ESPIRiT within the iterative TV reconstruction. We use the `fminbound` method in the `scipy.optimize` toolbox to find the optimal regularization parameters for TV.

3.9 | Evaluation metrics

For in vivo Brain Dataset, SPICER was compared with all the baselines shown in Section 3.8 at acceleration rates $R = 4, 6, 8$, and 10. For fastMRI Brain Dataset, SPICER was compared with TV, GRAPPA, SSDU, and supervised method E2E-VarNet trained on fully sampled data at the same acceleration $R = 4$ and 8. We used widely used quantitative metrics, peak signal-to-noise ratio (PSNR), measured in dB, structural similarity index (SSIM), and the normalized mean squared error (NMSE), with respective to the reference images obtained from the fully sampled data. The quantitative results were statistically analyzed by comparing SPICER with other image reconstruction methods within the targeted brain region, using the same threshold operator in ESPIRiT. We used the non-parametric Friedman's test and the post hoc test of the original FDR method of Benjamini and Hochberg.⁵⁶ The statistical analysis was performed using GraphPad Prism 9 (Version 9.3.1 for macOS, GraphPad Software). Statistical significance was defined as $p < 0.05$.

4 | RESULTS

We now present the numerical validation of SPICER on both in vivo MRI dataset and fastMRI dataset. The

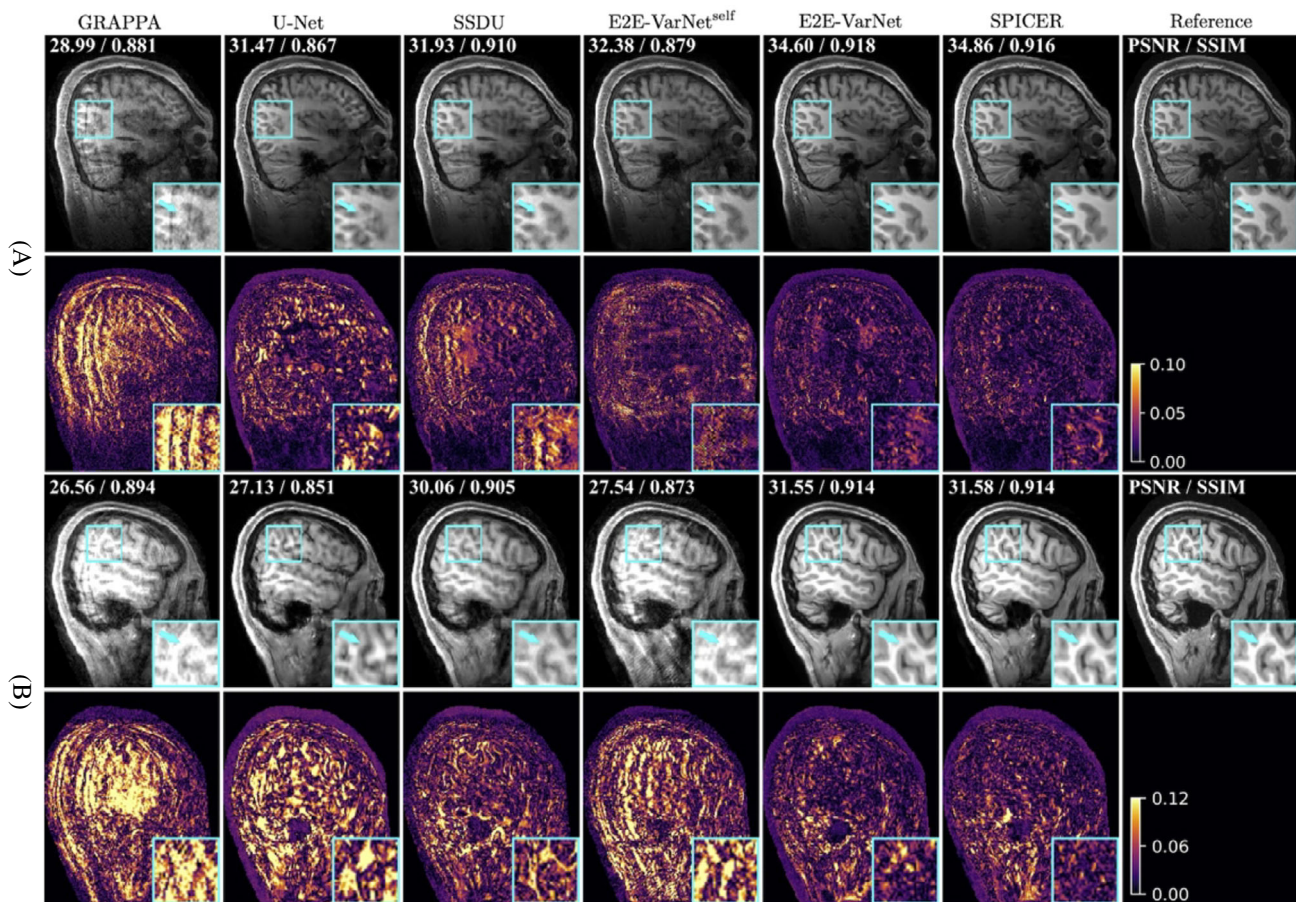


FIGURE 3 Visual and quantitative evaluation on in-vivo brain dataset corresponding to 8× and 10× acceleration rates. The top-right corner of each image provides the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) values with respect to the reference. We highlight the visually significant differences using zoom views and error maps. The visually important differences are highlighted using arrows. U-Net, SSDU^{auto}, and self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER) in the figure are based on self-supervised learning. E2E-VarNet is a supervised learning method trained using ground-truth. SPICER achieves the best performance compared to the baseline methods by jointly performing image reconstruction and CSM estimation with end-to-end self-supervised training. Note how compared to other methods, SPICER recovers sharper images and reduces artifacts, even achieving comparable performance to the supervised learning method. (A) 8× acceleration and (B) 10× acceleration.

results show that SPICER achieves state-of-the-art performance in highly accelerated acquisition settings. For better visualization, the two-dimensional visualization cases in Figures 3–6 are chosen randomly from middle 50 slides of the three-dimensional volume.

Figure 3 illustrates the results of image reconstruction on the in vivo dataset for acceleration rates $R = 8$ (top) and $R = 10$ (bottom). TV (with ESPIRiT) and GRAPPA contain blurring and ghosting artifacts, especially at higher acceleration rates. While U-Net has better performance than TV (with ESPIRiT) and GRAPPA by learning the prior from data, SSDU and SSDU^{auto} outperform it due to their model-based DL architectures. SSDU^{auto} performs joint CSMs estimation, which enables it to do better than SSDU. However, it continues to exhibit blurring, stemming from the incongruence between the training and testing phases, further intensified by the absence of paired

data utilization. This issue arises due to the undersampled measurement splitting strategy employed, highlighting the need for paired undersampled data and corresponding new splitting strategy. The supervised learning baseline E2E-VarNet and our proposed method SPICER achieve significant improvements compared to other methods. Overall, our proposed method SPICER can provide the best performance in artifact removal and sharpness compared to all of the self-supervised baseline methods.

Table 1 summarizes quantitative results of all the evaluated methods. In Table 1, SPICER achieves the highest PSNR, highest SSIM, and lowest NMSE values compared to other methods over all considered acceleration rates. Notably, SPICER outperformed all variants of the SSDU method, which highlights that the success of SPICER comes not merely from the access to paired undersampled data or a simplistic automated CSM estimation module.

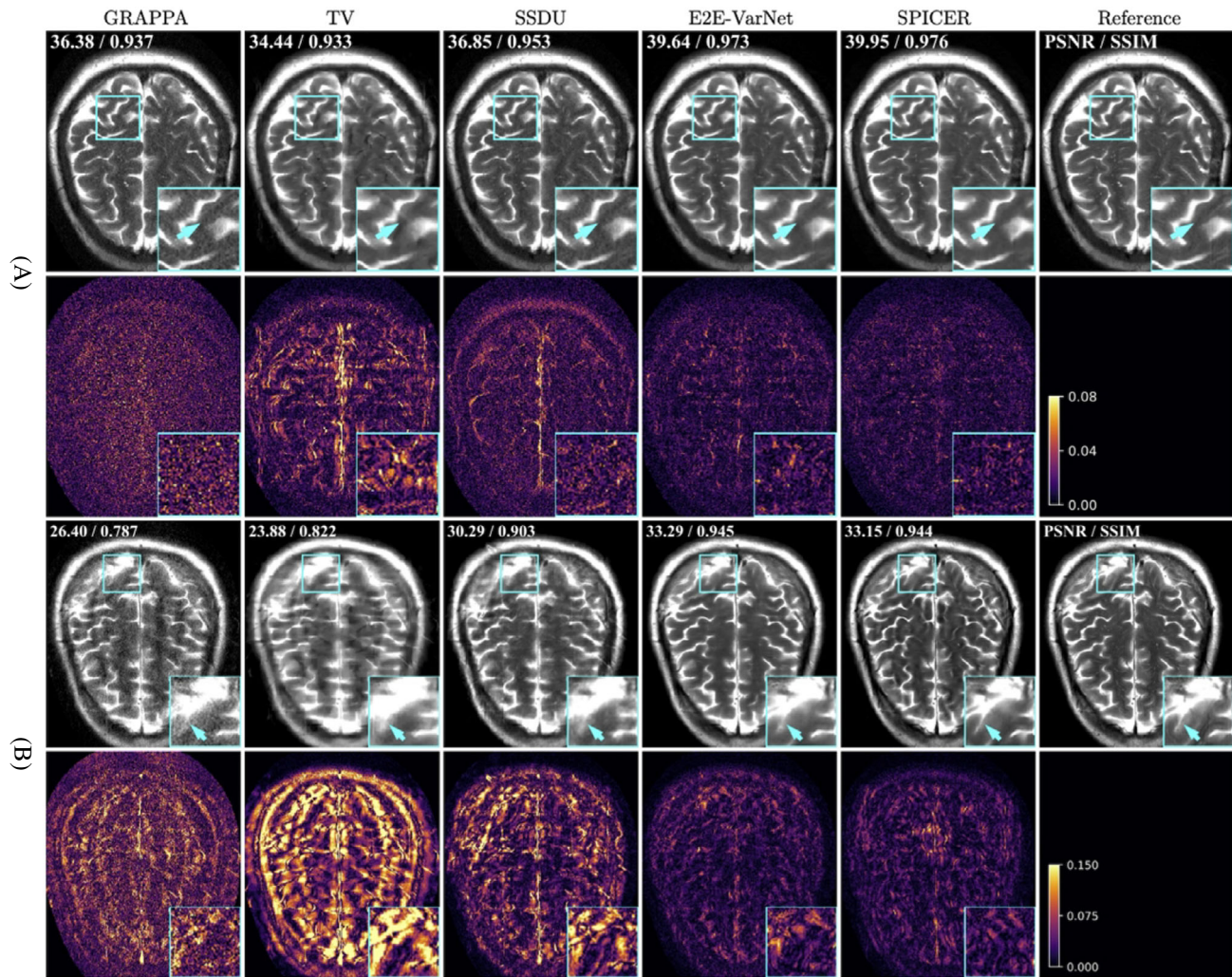


FIGURE 4 Visual and quantitative evaluation on fastMRI brain dataset corresponding to $4\times$ and $8\times$ acceleration rates. The top-right corner of each image provides the peak signal-to-noise ratio and structural similarity index values with respect to the reference. We highlight the visually significant differences using zoom views and error maps. The visually important differences are highlighted using arrows. SSDU and Self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER) in the figure are based on self-supervised learning. E2E-VarNet is a supervised learning method trained using ground-truth. SPICER achieves the best performance compared to the self-supervised baseline methods by jointly performing image reconstruction and coil sensitivity map estimation with end-to-end self-supervised training. Note how compared to other methods, SPICER recovers sharper images and reduces artifacts, even achieving comparable performance to the supervised learning method. (A) $4\times$ acceleration and (B) $8\times$ acceleration.

Moreover, our method achieves very competitive performance with a state-of-the-art supervised learning method E2E-VarNet. Note that E2E-VarNet also utilizes the joint learning of unrolling network and CSM estimation network. Statistical analysis of PSNR, SSIM, and NMSE values in Table 1 also highlights that SPICER can achieve statistically significant results compared to all the self-supervised image reconstruction baseline methods.

Figure 4 shows the results of image reconstruction on fastMRI dataset for acceleration rates $R = 4$ (top) and $R = 8$ (bottom). TV (with ESPIRiT) and GRAPPA contain artifacts and noise, especially at higher acceleration rates. SSDU successfully removes most of the artifacts and

noise, but it still suffers from some blurring, due to the use of suboptimal precalibrated CSMs. The supervised learning baseline E2E-VarNet and our proposed self-supervised method SPICER significantly outperform other methods. Overall, our proposed method SPICER can provide the best performance in artifact removal and sharpness compared to all of the self-supervised baseline methods on fastMRI dataset.

Table 2 presents the quantitative analysis of all the evaluated methods on fastMRI dataset. Table 2 shows that SPICER achieves the highest PSNR, highest SSIM, and lowest NMSE values compared to other self-supervised methods over all considered acceleration rates. Moreover,

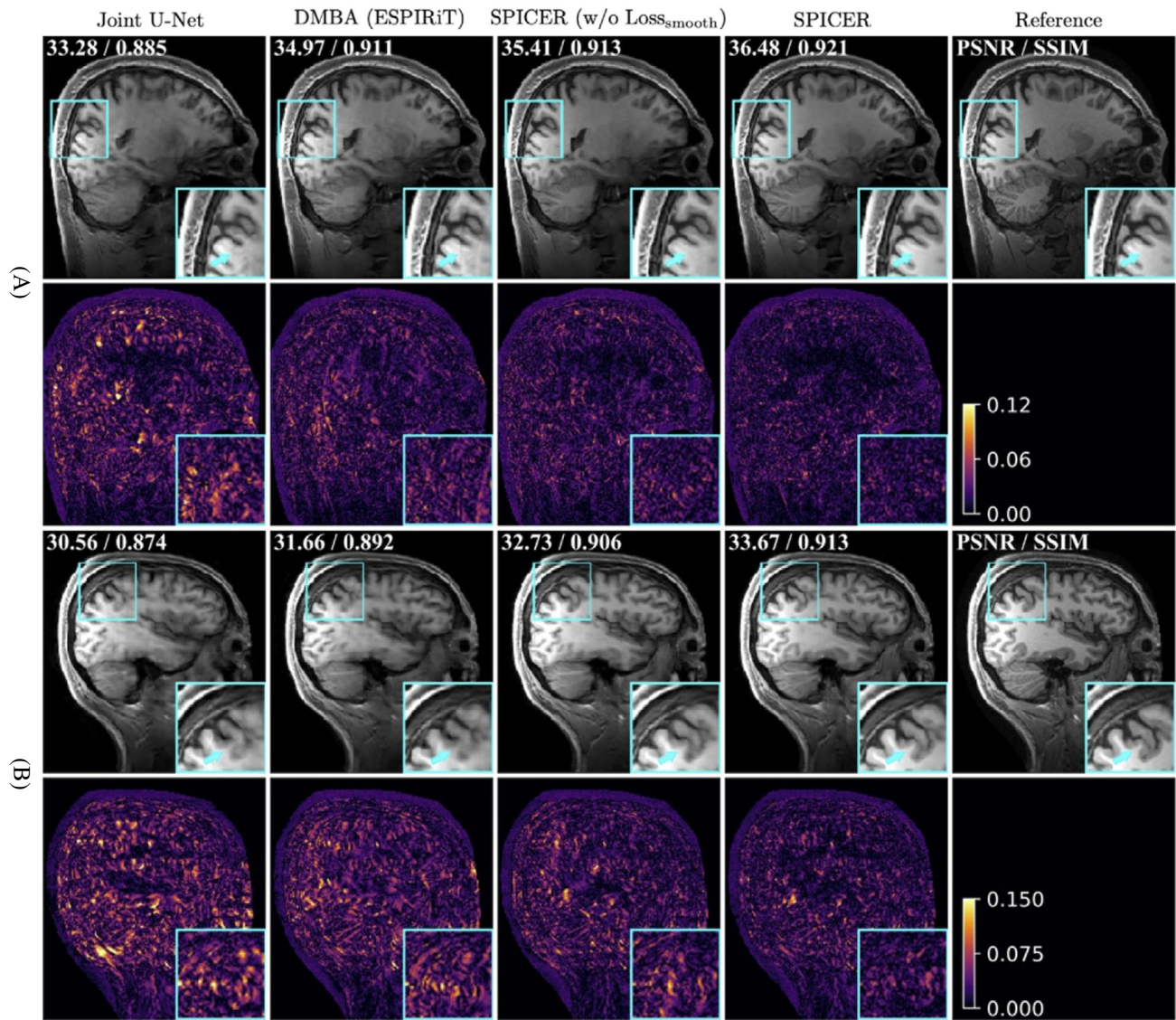


FIGURE 5 Quantitative evaluation of self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER) on in vivo brain dataset at 8× and 10× acceleration rates. The top-right corner of each image provides the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) values with respect to the reference. We highlight visually significant differences using zoom views and error maps. This figure shows that the deep model-based architecture (DMBA), coil sensitivity map (CSM) estimator, and CSM smoothness regularization of SPICER play important role in enhancing both quantitative and visual performance in terms of both artifact-removal and sharpness. (A) 8× acceleration and (B) 10× acceleration.

our method achieves competitive performance with the supervised learning method E2E-VarNet.

Figure 5 illustrates reconstruction results of the ablation study on 8× and 10× acceleration. Figure 5 shows that the DMBA model, CSM estimator, and CSM smoothness regularization of SPICER improve the imaging quality. Table 3 presents the results of an ablation study showing the influence of each module of our SPICER model and the effectiveness of Loss_{smooth} in the proposed self-supervised loss function (9). As can be seen, Joint U-Net obtains the worst performance, due to the U-Net model not using the PMRI measurement model. The

results suggest that pre-estimated CSMs generated with ESPIRiT used in DMBA lead to sub-optimal performance in all acceleration rates, especially with limited ACS lines. Note that at high acceleration rates (8× and 10×), even the CSMs pre-estimated with ESPIRiT on 4× accelerated data, the performance is still much worse than that of the joint CSM learning methods. The results of SSDU* and SPICER show that CSM smoothness regularization plays an important role in the reconstruction. Statistical analysis of quantitative values in Table 3 highlights that SPICER achieves statistically significant results compared to other methods.

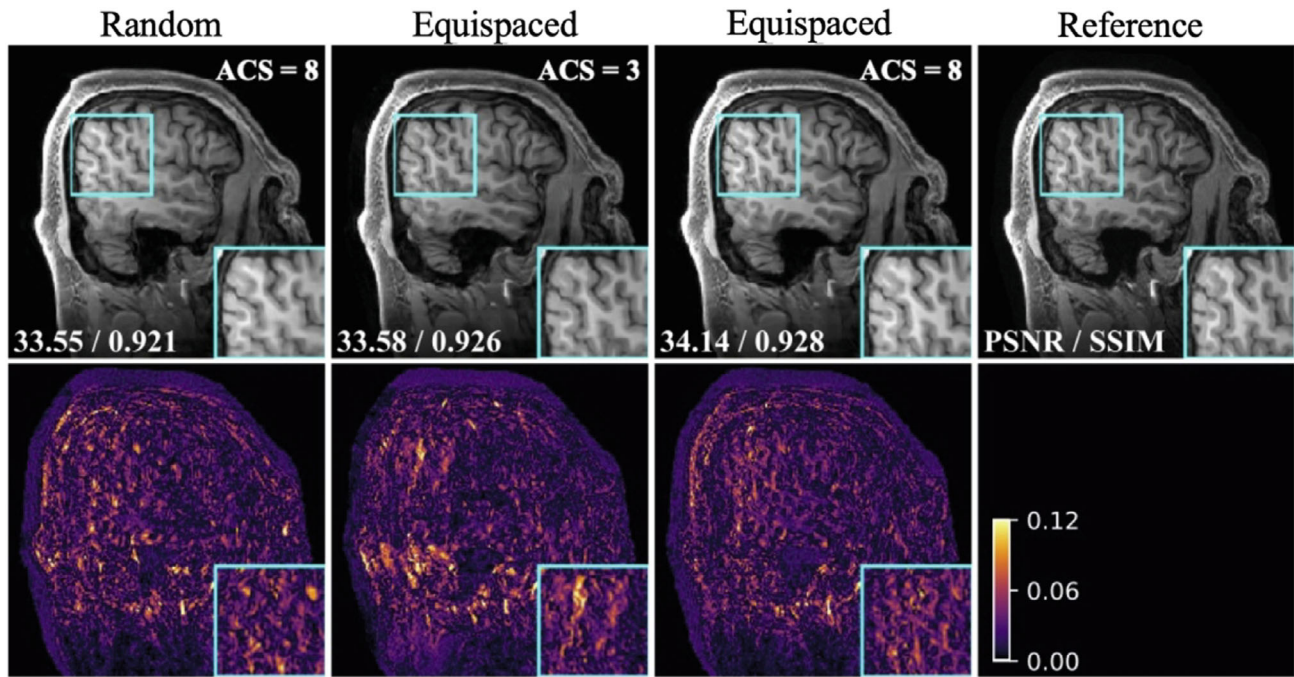


FIGURE 6 Visual and quantitative evaluation on in-vivo brain dataset corresponding to different sampling strategies for 8× acceleration rate. The first row shows the results from one-dimensional (1D) random sampling, 1D equispaced sampling with only three auto-calibration signal (ACS) lines, 1D equispaced sampling with 8 ACS lines and reference. The bottom-left corner of each image provides the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) values with respect to the reference. The top-right corner of each image provides the number of ACS lines used for each undersampling pattern. Visually significant differences are highlighted using error maps.

Figure 6 illustrates the performance of our SPICER method using various sampling patterns. We consider three distinct sampling patterns: 1D random sampling with eight ACS lines, 1D equispaced sampling with three ACS lines, and 1D equispaced sampling with eight ACS lines. The results suggest that SPICER can consistently provide high-quality reconstruction results even in challenging scenarios involving random sampling patterns and extremely limited ACS regions (only three lines).

Figure 7 illustrates the performance of the estimated CSMs. We use ESPIRiT and SPICER to estimate CSMs from data at various accelerated acquisition rates (from 4× to 10×). Estimated CSMs are then used within TV reconstruction from data corresponding to 4× accelerated acquisition rate. The reconstructed images are quantitatively evaluated using PSNR and SSIM values relative to the reference images. The top row of Figure 7 illustrates the TV reconstruction results equipped with CSMs estimated from different ACS data and different methods. The bottom row of Figure 7 illustrates the visualization of CSMs of one specific coil. Note that, even with CSMs estimated by SPICER from five ACS lines, the following TV reconstruction can achieve better performance than ESPIRiT CSMs (generated from 24 ACS lines).

5 | DISCUSSION AND CONCLUSIONS

5.1 | Discussion

In this manuscript, we proposed a self-supervised DMBA, namely SPICER, for joint MRI reconstruction and automatic coil sensitivity calibration. The key benefit of SPICER is that it is trained directly on pairs of noisy and undersampled k-space measurements of the same object without any fully sampled ground-truth data, which makes it broadly applicable when ground-truth is impossible or difficult to obtain. In addition, the automatic CSMs estimation module in SPICER shows the potential to estimate the high quality from only a few ACS lines, which can sharply increase the acceleration rate for the Cartesian sampling strategy.

Despite being trained on undersampled and noisy data, numerical results on the in-vivo Brain dataset demonstrate that SPICER can significantly outperform every self-supervised baseline while still achieving comparable reconstruction performance with the widely used supervised method E2E-VarNet. Additionally, we further validated SPICER's efficacy using data from fastMRI dataset. Both qualitatively and quantitatively, the reconstruction performances of SPICER at the acceleration rates of 4,

TABLE 1 Quantitative evaluation (mean $\pm$ SD) of self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER).

Method	PSNR $\uparrow$	SSIM $\uparrow$	NMSE $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	NMSE $\downarrow$
Acceleration factor	4 $\times$ acceleration			6 $\times$ acceleration		
Zero-Filled	18.27 $\pm$ 1.672 $\star$	0.775 $\pm$ 0.3842 $\star$	0.6212 $\pm$ 0.3460 $\star$	18.07 $\pm$ 1.694 $\star$	0.745 $\pm$ 0.0419 $\star$	0.6532 $\pm$ 0.3732 $\star$
Total variation ²⁸	37.87 $\pm$ 1.491 $\star$	0.959 $\pm$ 0.0097 $\star$	0.0043 $\pm$ 0.0019 $\star$	32.99 $\pm$ 1.305 $\star$	0.898 $\pm$ 0.0152 $\star$	0.0171 $\pm$ 0.0050 $\star$
GRAPPA ¹	38.54 $\pm$ 1.467 $\star$	0.966 $\pm$ 0.0071 $\star$	0.0039 $\pm$ 0.0012 $\star$	33.05 $\pm$ 1.205 $\star$	0.902 $\pm$ 0.0151 $\star$	0.0471 $\pm$ 0.0132 $\star$
U-Net ³⁰	35.36 $\pm$ 1.455 $\star$	0.925 $\pm$ 0.0139 $\star$	0.0111 $\pm$ 0.0034 $\star$	33.33 $\pm$ 1.425 $\star$	0.903 $\pm$ 0.0160 $\star$	0.0179 $\pm$ 0.0051 $\star$
SSDU ¹⁷	36.52 $\pm$ 1.676 $\star$	0.942 $\pm$ 0.0116 $\star$	0.0086 $\pm$ 0.0048 $\star$	33.91 $\pm$ 2.021 $\star$	0.921 $\pm$ 0.0144 $\star$	0.0176 $\pm$ 0.0191 $\star$
SSDU ^{auto}	37.07 $\pm$ 1.485 $\star$	0.943 $\pm$ 0.0125 $\star$	0.0074 $\pm$ 0.0041 $\star$	33.24 $\pm$ 1.520 $\star$	0.912 $\pm$ 0.0181 $\star$	0.0182 $\pm$ 0.0082 $\star$
SSDU*	38.21 $\pm$ 1.637 $\star$	0.945 $\pm$ 0.0129 $\star$	0.0050 $\pm$ 0.0019 $\star$	36.19 $\pm$ 1.499 $\star$	0.929 $\pm$ 0.01307 $\star$	0.0095 $\pm$ 0.0040 $\star$
SPICER	38.93 $\pm$ 1.620	0.949 $\pm$ 0.0111	0.0048 $\pm$ 0.0017	36.77 $\pm$ 1.575	0.935 $\pm$ 0.0120	0.0081 $\pm$ 0.0037
E2E-VarNet ²⁰	38.56 $\pm$ 1.768 $\star$	0.952 $\pm$ 0.0113 $\star$	0.0054 $\pm$ 0.0030 $\star$	36.09 $\pm$ 1.491 $\star$	0.938 $\pm$ 0.0136 $\star$	0.0095 $\pm$ 0.0056 $\star$
Acceleration factor	8 $\times$ acceleration			10 $\times$ acceleration		
Zero-filled	17.55 $\pm$ 1.510 $\star$	0.721 $\pm$ 0.0433 $\star$	0.7197 $\pm$ 0.3737 $\star$	16.90 $\pm$ 1.328 $\star$	0.701 $\pm$ 0.0450 $\star$	0.8209 $\pm$ 0.3865 $\star$
Total variation ²⁸	26.09 $\pm$ 1.218 $\star$	0.878 $\pm$ 0.0157 $\star$	0.1062 $\pm$ 0.0295 $\star$	25.28 $\pm$ 1.487 $\star$	0.863 $\pm$ 0.0156 $\star$	0.1331 $\pm$ 0.0519 $\star$
GRAPPA ¹	28.31 $\pm$ 1.390 $\star$	0.892 $\pm$ 0.0205 $\star$	0.0568 $\pm$ 0.0221 $\star$	25.79 $\pm$ 1.177 $\star$	0.873 $\pm$ 0.0185 $\star$	0.0924 $\pm$ 0.0354 $\star$
U-Net ³⁰	30.81 $\pm$ 1.543 $\star$	0.868 $\pm$ 0.0218 $\star$	0.0333 $\pm$ 0.0122 $\star$	29.17 $\pm$ 1.495 $\star$	0.836 $\pm$ 0.0256 $\star$	0.0489 $\pm$ 0.0207 $\star$
SSDU ¹⁷	32.58 $\pm$ 1.534 $\star$	0.910 $\pm$ 0.0154 $\star$	0.0216 $\pm$ 0.0127 $\star$	31.08 $\pm$ 1.567 $\star$	0.892 $\pm$ 0.0185 $\star$	0.0310 $\pm$ 0.0218 $\star$
SSDU ^{auto}	31.30 $\pm$ 1.634 $\star$	0.880 $\pm$ 0.0198 $\star$	0.0281 $\pm$ 0.0112 $\star$	30.45 $\pm$ 1.463 $\star$	0.878 $\pm$ 0.0204 $\star$	0.0347 $\pm$ 0.0198 $\star$
SSDU*	34.26 $\pm$ 1.539 $\star$	0.910 $\pm$ 0.0169 $\star$	0.0165 $\pm$ 0.0049 $\star$	32.14 $\pm$ 1.386 $\star$	0.894 $\pm$ 0.0208 $\star$	0.0228 $\pm$ 0.0070 $\star$
SPICER	34.66 $\pm$ 1.397	0.918 $\pm$ 0.0153	0.0128 $\pm$ 0.0036	32.94 $\pm$ 1.456	0.909 $\pm$ 0.0191	0.0176 $\pm$ 0.0067
E2E-VarNet ²⁰	34.18 $\pm$ 1.487 $\star$	0.920 $\pm$ 0.168 $\star$	0.0144 $\pm$ 0.0059 $\star$	32.91 $\pm$ 1.576 $\star$	0.919 $\pm$ 0.0141 $\star$	0.0194 $\pm$ 0.0083 $\star$

Notes: Note how SPICER achieves the best performance against other self-supervised baselines and comparable performance relative to the supervised method. Statistically significant differences compared with SPICER are marked ($p < 0.05$).

Abbreviations: NMSE, normalized mean squared error; PSNR, peak signal-to-noise ratio; SSIM, structural similarity index.

The bold value means the best results among all the self-supervised method comparison.

6, 8, and 10 consistently surpassed the state-of-the-art self-supervised method SSDU.

The proposed SPICER model achieves comparable results to the supervised learning method E2E-VarNet, primarily due to our training loss function (7) being specified in the k-space domain, which exhibits greater robustness to measurement noise. The training approach employed by E2E-VarNet relies on the SSIM loss specified on image domain, which makes it sensitive to noise and artifacts in the training data due to image formation. Consequently, SPICER demonstrates superior PSNR values compared to E2E-VarNet, albeit with slightly lower SSIM values.

We performed an ablation study to highlight the influence of the DMBA module, the CSMs estimation module, and the CSMs smoothness regularization within SPICER. As shown in Figure 5 and Table 3, the SPICER DMBA can lead to a sharp performance improvement compared with traditional U-Net model, due to the U-Net model not using the PMRI measurement model. In addition, the

joint estimation module shows the potential to estimate more accurate CSM compared to the widely used ESPIRiT method, which enables SPICER to achieve consistently better performance in all acceleration rates, especially with limited ACS lines. Additionally, the SPICER loss function performs better both qualitatively and statistically with the benefit of the CSM smoothness regularization.

We performed comparison to several variants of SSDU to emphasize that the superior performance of SPICER extends beyond our paired training data generation strategy. Table 1 illustrates that directly incorporating established self-supervised training techniques and an automatic CSM learning module resulted in minimal or negligible improvements. Notably, even when SSDU* employs the same training data splitting methodology as SPICER to generate data pairs, it demonstrates suboptimal performance. This performance gap arises from SPICER's unique training approach, which not only involves the utilization of Cartesian-subsampled data pairs but also

TABLE 2 Quantitative evaluation (Mean $\pm$ SD) of self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER) on fastMRI dataset.

Acceleration factor	4 $\times$ acceleration			8 $\times$ acceleration		
Method	PSNR $\uparrow$	SSIM $\uparrow$	NMSE $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	NMSE $\downarrow$
Zero-filled	23.61 $\pm$ 1.938 $\star$	0.774 $\pm$ 0.0417 $\star$	0.0701 $\pm$ 0.0445 $\star$	18.51 $\pm$ 1.898 $\star$	0.601 $\pm$ 0.0509 $\star$	0.244 $\pm$ 0.1612 $\star$
Total variation ²⁸	33.34 $\pm$ 1.477 $\star$	0.939 $\pm$ 0.0092 $\star$	0.0128 $\pm$ 0.0070 $\star$	24.86 $\pm$ 1.651 $\star$	0.819 $\pm$ 0.0358 $\star$	0.0935 $\pm$ 0.0570 $\star$
GRAPPA ¹	35.69 $\pm$ 1.022 $\star$	0.893 $\pm$ 0.0290 $\star$	0.0071 $\pm$ 0.0025 $\star$	26.71 $\pm$ 1.292 $\star$	0.724 $\pm$ 0.0661 $\star$	0.0584 $\pm$ 0.0292 $\star$
SSDU ¹⁷	38.79 $\pm$ 1.052 $\star$	0.953 $\pm$ 0.0104 $\star$	0.0034 $\pm$ 0.0012 $\star$	27.63 $\pm$ 2.135 $\star$	0.876 $\pm$ 0.0213 $\star$	0.0489 $\pm$ 0.0319 $\star$
SPICER	39.64 $\pm$ 0.976	0.970 $\pm$ 0.0051	0.0028 $\pm$ 0.0011	33.48 $\pm$ 1.648	0.946 $\pm$ 0.0094	0.0116 $\pm$ 0.0042
E2E-VarNet ²⁰	39.46 $\pm$ 1.128 $\star$	0.972 $\pm$ 0.0063 $\star$	0.0030 $\pm$ 0.0014 $\star$	34.70 $\pm$ 1.341 $\star$	0.950 $\pm$ 0.0084 $\star$	0.0093 $\pm$ 0.0054 $\star$

Notes: How SPICER achieves the best performance against other self-supervised baselines and comparable performance relative to the supervised method.

Statistically significant differences compared with SPICER are marked ($p\star < 0.05$).

Abbreviations: NMSE, normalized mean squared error; PSNR, peak signal-to-noise ratio; SSIM, structural similarity index.

The bold value means the best results among all the self-supervised method comparison.

TABLE 3 Quantitative evaluation (mean $\pm$ SD) using four distinct implementations of self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction (SPICER).

Acceleration factor	4 $\times$ acceleration			6 $\times$ acceleration		
Method	PSNR $\uparrow$	SSIM $\uparrow$	NMSE $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	NMSE $\downarrow$
Joint U-Net	35.77 $\pm$ 1.352 $\star$	0.950 $\pm$ 0.0119 $\star$	0.0105 $\pm$ 0.0030 $\star$	33.45 $\pm$ 1.525 $\star$	0.911 $\pm$ 0.0179 $\star$	0.0168 $\pm$ 0.0054 $\star$
DMBA (ESPIRiT)	36.80 $\pm$ 1.502 $\star$	0.959 $\pm$ 0.0098 $\star$	0.0079 $\pm$ 0.0024 $\star$	35.21 $\pm$ 1.313 $\star$	0.923 $\pm$ 0.01557 $\star$	0.0111 $\pm$ 0.0032 $\star$
SSDU*	38.21 $\pm$ 1.637 $\star$	0.945 $\pm$ 0.01299 $\star$	0.0050 $\pm$ 0.0019 $\star$	36.19 $\pm$ 1.499 $\star$	0.929 $\pm$ 0.01307 $\star$	0.0095 $\pm$ 0.0040 $\star$
SPICER	38.93 $\pm$ 1.620	0.949 $\pm$ 0.0111	0.0048 $\pm$ 0.0017	36.77 $\pm$ 1.575	0.935 $\pm$ 0.0120	0.0081 $\pm$ 0.0037

Notes: This table shows the influence of the DMBA reconstruction module, the CSM estimation module, and the smooth term in the loss function. Statistically significant differences compared with SPICER are marked ($p\star < 0.05$).

Abbreviations: CSM, coil sensitivity map; DMBA, deep model-based architecture; NMSE, normalized mean squared error; PSNR, peak signal-to-noise ratio; SSIM, structural similarity index.

The bold value means the best results among all the self-supervised method comparison.

incorporates CSMs with smoothness regularization. This integration plays a pivotal role in achieving the superior performance observed in comparison to SSDU*.

5.2 | Limitations and future directions

Several limitations exist for our proposed DL reconstruction framework. First, our numerical study is currently limited to brain images. However, the approach of jointly using self-supervised learning for CSMs and DMBA for accurate image reconstruction can be extended to a variety of imaging scenarios, provided that high-quality training data is available. Future research could explore the adaptation of this approach to other body parts, particularly in cases where body coils lack a fixed geometry, which could pose additional challenges.

We furthermore recognize that our model's current training and testing are limited to Cartesian sampling trajectories. There is growing interest in extending SPICER

to non-Cartesian sampling trajectories, such as spiral and radial trajectories. These trajectories offer the advantage of densely sampling the central k-space region, and the resulting zero-filled images can be used as inputs to the CSM estimation module, providing similar benefits to the ACS lines in Cartesian sampling. The extension of SPICER to non-Cartesian sampling presents an exciting opportunity to improve the technique's performance and applicability in various imaging scenarios that demand high spatiotemporal resolution or improved motion robustness.

Another limitation of SPICER is that it performs three-dimensional reconstruction slice-by-slice. To enhance its capabilities, we envision the formulation of a three-dimensional variant for reconstructing three-dimensional volumes in cases where undersampling may occur in two dimensions. Such a development can potentially further accelerate the MRI acquisition.

Furthermore, the current coil sensitivity estimation module in SPICER relies on ACS data for calibration

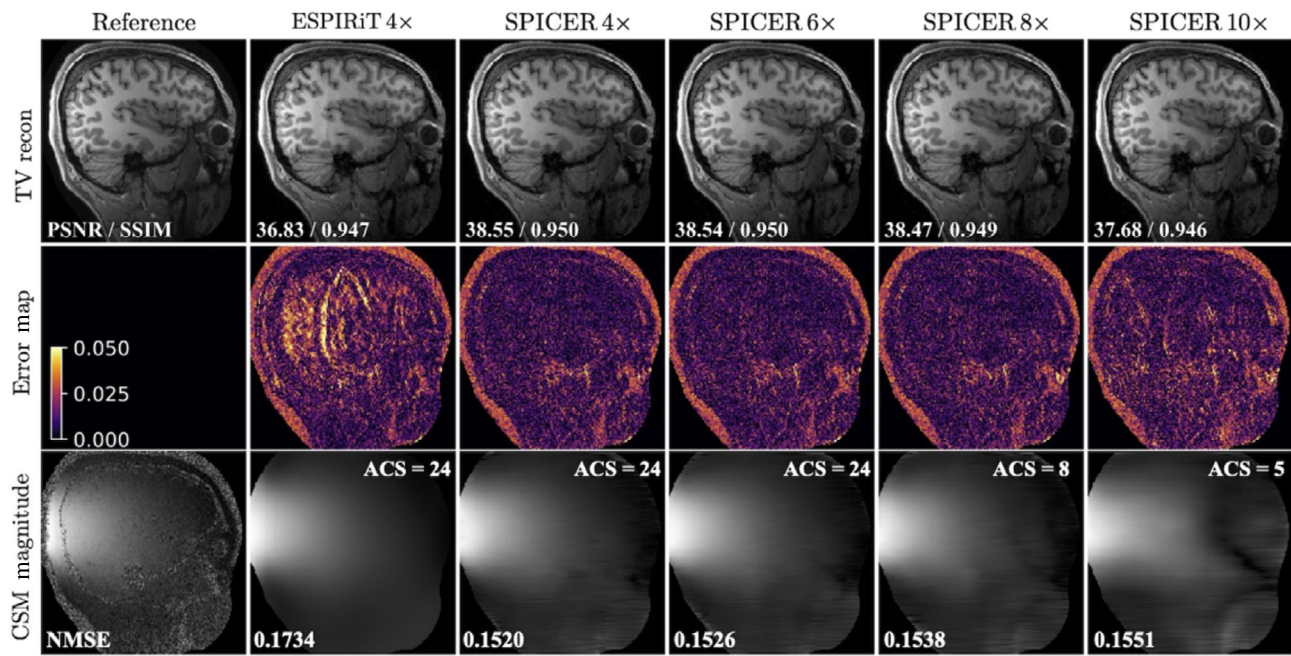


FIGURE 7 Evaluation of coil sensitivity maps (CSMs) estimated by SPICER at various acceleration rates. The first row shows the TV reconstruction results at acceleration rate $R = 4$ with the corresponding CSM shown in the third row. The bottom-left corner of each reconstruction image provides the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) values with respect to the reference. The third row shows the magnitude image of the 10th CSM. For each undersampling pattern, the number of ACS lines used is indicated in the top-right corner of each CSM image. The bottom-left corner of each CSM image includes the NMSE value relative to the reference CSM. Visually significant differences are highlighted using error maps in the second row.

of CSM. This dependency limits its utility to Echo Planar Imaging (EPI) sequences, such as fMRI and diffusion MRI. Exploring the expansion of SPICER to incorporate non-ACS data represents an intriguing and valuable direction to enhance its practicality.

6 | CONCLUSION

In conclusion, we presented SPICER, as a self-supervised MRI reconstruction method that can automatically estimate CSMs to enhance the reconstruction performance even at high undersampled rates. Our results indicate that the synergistic combination of all three SPICER modules enables it to outperform other self-supervised learning methods and match the performance of the well-known E2E-VarNet trained on fully sampled ground-truth data.

DATA AVAILABILITY STATEMENT

We performed all our experiments on a machine equipped with an Intel Xeon Gold 6130 Processor and an NVIDIA GeForce RTX 3090 GPU. The implementation details can be found in: <https://github.com/wustl-cig/SPICER>.

ORCID

Yuyang Hu <https://orcid.org/0000-0003-2349-8350>

Weijie Gan <https://orcid.org/0000-0003-3604-784X>

Chunwei Ying <https://orcid.org/0000-0003-1060-3289>

Cihat Eldeniz <https://orcid.org/0000-0002-4457-0916>

Hongyu An <https://orcid.org/0000-0001-6459-2269>

Ulugbek S. Kamilov <https://orcid.org/0000-0001-6770-3278>

TWITTER

Ulugbek S. Kamilov [ukmlv](https://twitter.com/ukmlv)

REFERENCES

1. Griswold MA, Jakob PM, Heidemann RM, et al. Generalized autocalibrating partially parallel acquisitions (GRAPPA). *Magn Reson Med*. 2002;47:1202-1210.
2. Sodickson DK, Manning WJ. Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays. *Magn Reson Med*. 1997;38:591-603.
3. Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P. SENSE: sensitivity encoding for fast MRI. *Magn Reson Med*. 1999;42:952-962.
4. Uecker M, Lai P, Murphy MJ, et al. ESPIRiT-an eigenvalue approach to autocalibrating parallel MRI: where SENSE meets GRAPPA. *Magn Reson Med*. 2014;71:990-1001.
5. Lustig M, Donoho D, Pauly JM. Sparse MRI: the application of compressed sensing for rapid MR imaging. *Magn Reson Med*. 2007;58:1182-1195.
6. Ye JC. Compressed sensing MRI: a review from signal processing perspective. *BMC Biomed Eng*. 2019;1:8.

7. Ongie G, Jalal A, Metzler CA, Baraniuk RG, Dimakis AG, Willett R. Deep learning techniques for inverse problems in imaging. *IEEE J Select Areas Inf Theory*. 2020;1:39-56.
8. Knoll F, Hammernik K, Zhang C, et al. Deep-learning methods for parallel magnetic resonance imaging reconstruction: a survey of the current approaches, trends, and issues. *IEEE Signal Process Mag*. 2020;37:128-140.
9. Montalt-Tordera J, Muthurangu V, Hauptmann A, Steeden J. Machine learning in magnetic resonance imaging: image reconstruction. *Phys Med*. 2021;83:79-87.
10. Jin KH, McCann MT, Froustey E, Unser M. Deep convolutional neural network for inverse problems in imaging. *IEEE Trans Image Process*. 2017;26:4509-4522.
11. Zhu B, Liu JZ, Cauley SF, Rosen BR, Rosen MS. Image reconstruction by domain-transform manifold learning. *Nature*. 2018;555:487-492.
12. Hammernik K, Schlemper J, Qin C, Duan J, Summers RM, Rueckert D. Systematic evaluation of iterative deep neural networks for fast parallel MRI reconstruction with sensitivity-weighted coil combination. *Magn Reson Med*. 2021;86:1859-1872.
13. Liang D, Cheng J, Ke Z, Ying L. Deep magnetic resonance image reconstruction: inverse problems meet neural networks. *IEEE Signal Process Mag*. 2020;37:141-151.
14. Duan J, Schlemper J, Qin C, et al. VS-net: variable splitting network for accelerated parallel MRI reconstruction. Paper presented at: Proceedings of Medical Image Computing and Computer-Assisted Intervention. 2019; Cham, Switzerland:713-722.
15. Aggarwal HK, Mani MP, Jacob M. MoDL: model-based deep learning architecture for inverse problems. *IEEE Trans Med Imaging*. 2019;38:394-405.
16. Hammernik K, Klatzer T, Kobler E, et al. Learning a variational network for reconstruction of accelerated MRI data. *Magn Reson Med*. 2018;79:3055-3071.
17. Yaman B, Hosseini SAH, Moeller S, Ellermann J, Uğurbil K, Akçakaya M. Self-supervised learning of physics-guided reconstruction neural networks without fully sampled reference data. *Magn Reson Med*. 2020;84:3172-3191.
18. Arvinte M, Vishwanath S, Tewfik AH, Tamir JJ. Deep J-Sense: accelerated MRI reconstruction via unrolled alternating optimization. *arXiv:2103.02087 [cs, eess]*. 2021.
19. Jun Y, Shin H, Eo T, Hwang D. Joint deep model-based MR image and coil sensitivity reconstruction network (joint-icnet) for fast MRI. Paper presented at: Proceedings of IEEE Conference on Computer Vision on Pattern Recognition. 2021; Nashville, TN:5270-5279.
20. Sriram A, Zbontar J, Murrell T, et al. End-to-end variational networks for accelerated MRI reconstruction. Paper presented at: Proceedings of Medical Image Computing and Computer-Assisted Intervention. 2020; Lima, Peru:64-73.
21. Yaman B, Hosseini SAH, Moeller S, Ellermann J, Uğurbil K, Akçakaya M. Self-supervised physics-based deep learning mri reconstruction without fully-sampled data. Paper presented at: Proceedings of International Symposium on Biomedical Imaging, Virtual, Iowa City, Iowa. 2020:921-925.
22. Akçakaya M, Yaman B, Chung H, Ye JC. Unsupervised deep learning methods for biological image reconstruction. *arXiv:2105.08040 [physics]*. 2021.
23. Lehtinen J, Munkberg J, Hasselgren J, et al. Noise2Noise: learning image restoration without clean data. Paper presented at: Proceedings of International Conference on Machine Learning. 2018; Stockholm, Sweden.
24. Eldeniz C, Gan W, Chen S, et al. Phase2Phase: respiratory motion-resolved reconstruction of free-breathing magnetic resonance imaging using deep learning without a ground truth for improved liver imaging. *Invest Radiol*. 2021;56:809-819.
25. Gan W, Sun Y, Eldeniz C, Liu J, An H, Kamilov US. Deep image reconstruction using unregistered measurements without groundtruth. Paper presented at: Proceedings of International Symposium on Biomedical Imaging; April. 2021; Nice, France:1531-1534.
26. Liu J, Sun Y, Eldeniz C, Gan W, An H, Kamilov US. RARE: image reconstruction using deep priors learned without Groundtruth. *IEEE J Select Topic Signal Process*. 2020;14:1088-1099.
27. Gan W, Hu Y, Eldeniz C, et al. Ss-jircs: self-supervised joint image reconstruction and coil sensitivity calibration in parallel mri without ground truth. Paper presented at: Proceedings of IEEE International Conference on Computer Vision Workshops. 2021; Montreal, Canada:4048-4056.
28. Block KT, Uecker M, Frahm J. Undersampled radial mri with multiple coils. Iterative image reconstruction using a total variation constraint. *Magn Reson Med*. 2007;57:1086-1098.
29. Wang G, Ye JC, De Man B. Deep learning for tomographic image reconstruction. *Nat Mach Intell*. 2020;2:737-748.
30. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. Paper presented at: Proceedings of Medical Image Computing and Computer-Assisted Intervention, Munich, Germany. 2015:234-241.
31. Lee D, Yoo J, Ye JC. Deep residual learning for compressed sensing MRI. Paper presented at: Proceedings of International Symposium on Biomedical Imaging; April. 2017; Melbourne, Australia:15-18.
32. Wang S, Su Z, Ying L, et al. Accelerating magnetic resonance imaging via deep learning. Paper presented at: Proceedings of International Symposium on Biomedical Imaging; April. 2016; Prague, Czech Republic:514-517.
33. Han Y, Sunwoo L, Ye JC. K-space deep learning for accelerated MRI. *IEEE Trans Med Imaging*. 2020;39:377-386.
34. Venkatakrishnan SV, Bouman CA, Wohlberg B. Plug-and-play priors for model based reconstruction. Paper presented at: Proceedings of IEEE Global Conference on Signal Processing and Information Processing; December. 2013; Austin, TX:945-948.
35. Kamilov US, Bouman CA, Buzzard GT, Wohlberg B. Plug-and-play methods for integrating physical and learned models in computational imaging. *IEEE Signal Process Mag*. 2023;40(1):85-97.
36. Romano Y, Elad M, Milanfar P. The little engine that could: regularization by Denoising (RED). *SIAM J Imaging Sci*. 2017;10:1804-1844.
37. Schlemper J, Caballero J, Hajnal JV, Price AN, Rueckert D. A deep Cascade of convolutional neural networks for dynamic MR image reconstruction. *IEEE Trans Med Imaging*. 2018;37:491-503.
38. Liu J, Sun Y, Gan W, Xu X, Wohlberg B, Kamilov US. SGD-net: efficient model-based deep learning with theoretical guarantees. *IEEE Trans Comput Imaging*. 2021;7:598-610.

39. Akçakaya M, Steen M, Sebastian W, Kâmil U. Scan-specific robust artificial-neural-networks for k-space interpolation (raki) reconstruction: database-free deep learning for fast imaging. *Magn Reson Med*. 2019;81:439-453.
40. Ying L, Sheng J. Joint image reconstruction and sensitivity estimation in SENSE (JSENSE). *Magn Reson Med*. 2007;57:1196-1202.
41. Uecker M, Hohage T, Block KT, Frahm J. Image reconstruction by regularized nonlinear inversion-joint estimation of coil sensitivities and image content. *Magn Reson Med*. 2008;60:674-682.
42. Peng X, Sutton BP, Lam F, Liang Z. DeepSense: learning coil sensitivity functions for sense reconstruction using deep learning. *Magn Reson Med*. 2022;87:1894-1902.
43. Yiasemis G, Sonke J, Sánchez C, Teuwen J. Recurrent variational network: a deep learning inverse problem solver applied to the task of accelerated mri reconstruction. Paper presented at: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2022; New Orleans, LA:732-741.
44. Blumenthal M, Luo G, Schilling M, Haltmeier M, Uecker M. Nlinv-net: self-supervised end-2-end learning for reconstructing undersampled radial cardiac real-time data. Paper presented at: ISMRM Annual Meeting. 2022; London, UK. Abstract 0499.
45. Tang L, Zhao Y, Li Y, et al. Jsense-pro: joint sensitivity estimation and image reconstruction in parallel imaging using pre-learned subspaces of coil sensitivity functions. *Magn Reson Med*. 2023;89:1531-1542.
46. Meng N, Yang Y, Xu Z, Sun J. A prior learning network for joint image and sensitivity estimation in parallel mr imaging. Paper presented at: Proc. Medical Image Computing and Computer-Assisted Intervention. 2019; Shenzhen, China:732-740.
47. Zeng G, Guo Y, Zhan J, et al. A review on deep learning mri reconstruction without fully sampled k-space. *BMC Med Imaging*. 2021;21:1-11.
48. Akçakaya M, Yaman B, Chung H, Ye JC. Unsupervised deep learning methods for biological image reconstruction and enhancement: An overview from a signal processing perspective. *IEEE Signal Process Mag*. 2022;39:28-44.
49. Tachella J, Chen D, Davies M. Sampling theorems for unsupervised learning in linear inverse problems. *arXiv:2203.12513* 2022.
50. Krull A, Buchholz TO, Jug F. Noise2Void – learning denoising from single noisy images. Paper presented at: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition; June. 2019:2124-2132.
51. Ulyanov D, Vedaldi A, Lempitsky V. Deep image prior. Paper presented at: Proc. IEEE Conference on Computer Vision and Pattern Recognition. 2018; Salt Lake City, UT:9446-9454.
52. Bora A, Price E, Dimakis AG. Ambientgan: generative models from lossy measurements. Paper presented at: International Conference on Learning Representation. 2018; Vancouver, Canada.
53. Chen D, Tachella J, Davies ME. Equivariant imaging: learning beyond the range space. Paper presented at: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 2021; Nashville, TN:4379-4388.
54. Huang P, Zhang C, Zhang X, Li X, Dong L, Ying L. Self-supervised deep unrolled reconstruction using regularization by denoising. *IEEE Trans Med Imaging*. 2024;43(3):1203-1213.
55. Paszke A, Gross S, Massa F, et al. Pytorch: an imperative style, high-performance deep learning library. Paper presented at: Proceedings of Annual Conference on Neural Information Processing Systems. 2019; Vancouver, Canada.
56. Benjamini Y, Hochberg Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J R Stat Soc B Methodol*. 1995;57:289-300.

How to cite this article: Hu Y, Gan W, Ying C, et al. SPICER: Self-supervised learning for MRI with automatic coil sensitivity estimation and reconstruction. *Magn Reson Med*. 2024;92:1048-1063. doi: 10.1002/mrm.30121