An end-to-end-trainable iterative network architecture for accelerated radial multi-coil 2D cine MR image reconstruction

Andreas Kofler[a)]
Physikalisch-Technische Bundesanstalt (PTB), Braunschweig, Berlin 10587, Germany

Markus Haltmeier
Department of Mathematics, University of Innsbruck, Innsbruck 6020, Austria

Tobias Schaeffter
Physikalisch-Technische Bundesanstalt (PTB), Braunschweig, Berlin 10587, Germany
School of Imaging Sciences and Biomedical Engineering, King's College London, London SE1 7EH, UK
Department of Biomedical Engineering, Technical University of Berlin, Berlin 10623, Germany

Christoph Kolbitsch
Physikalisch-Technische Bundesanstalt (PTB), Braunschweig, Berlin 10587, Germany
School of Imaging Sciences and Biomedical Engineering, King's College London, London SE1 7EH, UK

(Received 24 December 2020; revised 11 February 2021; accepted for publication 18 February 2021; published 1 April 2021)

Purpose: Iterative convolutional neural networks (CNNs) which resemble unrolled learned iterative schemes have shown to consistently deliver state-of-the-art results for image reconstruction problems across different imaging modalities. However, because these methods include the forward model in the architecture, their applicability is often restricted to either relatively small reconstruction problems or to problems with operators which are computationally cheap to compute. As a consequence, they have not been applied to dynamic non-Cartesian multi-coil reconstruction problems so far.
Methods: In this work, we propose a CNN architecture for image reconstruction of accelerated 2D radial cine MRI with multiple receiver coils. The network is based on a computationally light CNN component and a subsequent conjugate gradient (CG) method which can be jointly trained end-to-end using an efficient training strategy. We investigate the proposed training strategy and compare our method with other well-known reconstruction techniques with learned and non-learned regularization methods.
Results: Our proposed method outperforms all other methods based on non-learned regularization. Further, it performs similar or better than a CNN-based method employing a 3D U-Net and a method using adaptive dictionary learning. In addition, we empirically demonstrate that even by training the network with only iteration, it is possible to increase the length of the network at test time and further improve the results.
Conclusions: End-to-end training allows to highly reduce the number of trainable parameters of and stabilize the reconstruction network. Further, because it is possible to change the length of the network at the test time, the need to find a compromise between the complexity of the CNN-block and the number of iterations in each CG-block becomes irrelevant. [https://doi.org/10.1002/mp.14809]

Key words: deep learning, inverse problems, magnetic resonance imaging, neural networks

1. INTRODUCTION

Magnetic resonance imaging (MRI) is an important tool for the assessment of different cardiovascular diseases. Cardiac cine MRI, for example, is used to assess the cardiac function as well as left and right ventricular volumes and left ventricular mass.[1] However, MRI is also known to suffer from relatively long data acquisition times. In cardiac cine MRI, the data acquisition usually has to take place during a single breathhold of the patient in order to avoid artifacts arising from the respiratory motion. Since for patients with limited breathhold capabilities this can be challenging, undersampling techniques can be used to accelerate the measurement process. Undersampling in Fourier domain, the so-called k-space, leads to a violation of the Nyquist–Shannon sampling theorem and therefore, regularization techniques must be used to reconstruct artifact-free images.

Recently, image reconstruction has experienced a paradigm shift with the re-emergence of neural networks (NNs), and in particular, convolutional NNs (CNNs) which can be employed as regularization methods.[2] CNNs can be used in different ways as regularization methods for image reconstruction problems. A key element to categorize these methods is whether the forward operator is used in the learning process.[3] A straightforward approach is to simply post-process an initial estimate of the solution which is impaired

by noise and artifacts (see, e.g., Ref. [4–7]). Further, cross-/multi-domain methods which pre-process the *k*-space and subsequently process the initially obtained reconstruction have been investigated as well.[8] However, the post-processed image might lack data consistency, meaning that it is not clear how well the post-processed image matches the measured *k*-space data. Thus, approaches which used the output of a pretrained CNN as image-priors have been considered as well (see, e.g., Ref. [9,10]). Thereby, the CNN-priors are used in the formulation of a Tikhonov functional which is subsequently minimized to increase the data consistency of the solution. These methods have the (computational) advantage of decoupling the regularization step from increasing data consistency and are thus also applicable to arbitrary large-scale image reconstruction problems. However, the network training is typically carried out in the absence of the forward model and although this strategy was reported to be successful, image reconstruction methods based on NNs have been reported to possibly suffer from instabilities.[11] This issue could directly affect the obtained CNN-prior and thus, affect the quality of the final reconstruction, since its solution depends on the CNN-prior. Interestingly, in Ref. [11], the authors showed that the CNN-based reconstruction methods which include the forward and adjoint operators in the network architecture are the most stable with respect to small perturbations and adversarial attacks. Further, including the forward model in the network architecture has been reported to lower the maximum error bound of the CNNs.[12] Thus, it is desirable to find a way to include the physical model in the CNN architecture, even for computationally expensive/large-scale problems.

Methods in which the CNNs architectures resemble unrolled iterative schemes of finite length are referred to as variational/iterative/cascaded networks (see, e.g., Ref. [13-19,43]). Thereby, these methods typically consist of CNN-blocks and data consistency (DC)-blocks, which are given in the form of gradient steps with respect to a data-fidelity term (see, e.g., Ref. [14,19]), or as layers which implement the minimizer of a functional involving a data-fidelity term (see, e.g., Ref. [13,16,18]). While the CNN-blocks can be interpreted as learned regularizers, the DC-blocks utilize the measured undersampled *k*-space data to increase/ensure the data consistency of the intermediate CNN outputs.

Unfortunately, including the forward and adjoint operators in the CNN can at the same time represent the computational bottleneck of these methods, since the forward operator as well as its adjoint can be computationally expensive to evaluate. As a consequence, the applicability of iterative networks is currently still limited to either reconstruction problems with easy-to-compute forward operators, for example, a FFT sampled on a Cartesian grid[13,16,18,21] or to non-dynamic problems with non-Cartesian sampling schemes, see Ref. [22] for a single-coil data acquisition or Ref. [23] for a multiple-coil acquisition. For example, in Ref. [13,16,18] only a single-coil is used for the encoding operator. In,[21] multiple receiver coils were used for a dynamic 3D reconstruction problem with a Cartesian sampling grid. For non-dynamic problems, in contrast, iterative CNNs for multi-coil data acquisition on a Cartesian grid have been received more attention (see, e.g., Ref. [14,15,24]).

Acquisition protocols using non-Cartesian sampling trajectories such as radial or spiral sampling can be an attractive alternative to standard Cartesian acquisitions, especially because the arising undersampling artifacts are much more incoherent compared to the ones arising from a Cartesian acquisition.[25] However, radially acquired *k*-space data are computationally more demanding to reconstruct as the forward encoding operator involves gridding on a Cartesian grid before the fast Fourier transform can be applied, see Ref. [26] for more details. In,[22] an iterative network for a non-dynamic radial single-coil acquisition was proposed. However, the standard in the clinical routine is in fact given by multi-coil data-acquisitions[27] through which the acquisition of the *k*-space data increases in terms of computational complexity. Finally, acquiring a dynamic process is inherently computationally more demanding as for each time point, the encoding operator has to be applied to the image.

In this work, we present a novel computationally light and efficient CNN-block architecture as well as a training strategy which are tailored to image reconstruction of dynamic multi-coil radial MR data of the heart. The combination of the proposed network architecture and training strategy allows to train the entire reconstruction network in an end-to-end manner, even for dynamic problems with nonuniformly sampled data as well as multiple receiver coils. To the best of our knowledge, this is the first work which overcomes these computational difficulties and presents such an end-to-end trainable network architecture for dynamic non-Cartesian multi-coil MR reconstruction problems.

The paper is structured as follows: In Section 2, we formally introduce the reconstruction problem and the proposed method by discussing in more detail the used CNN-block as well as the training strategy. In Section 3, we show extensive experiments to validate the efficacy of our approach and compare with to different state-of-the-art methods for dynamic cardiac radial MRI. We then discuss the main advantages and limitations of our work in Section 4 and conclude the work in Section 5.

2. MATERIALS AND METHODS

2.A. Problem formulation

Let $\mathbf{x} \in \mathbb{C}^N$ with $N = N_x \times N_y \times N_t$ denote the vector representation of a complex-valued cine MR image, that is, $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_{N_t}]^{\mathrm{T}}$. The forward operator $\mathbf{A}$ maps the dynamic cine MR image to its corresponding *k*-space. In this work, we focus on a 2D radial encoding operator using multiple receiver coils. More precisely, the operator $\mathbf{A}$ is given by.

$$\mathbf{A} := (\mathbf{I}_{N_c} \otimes \mathbf{E})\mathbf{C}, \tag{1}$$

where $\mathbf{I}_{N_c}$ denotes the identity operator and $\mathbf{C}$ consists of the concatenation of the $N_c$different coil-sensitivity maps which are multiplied with the cine MR image, that is,

$\mathbf{C} = [\mathbf{C}_1, \ldots, \mathbf{C}_{N_c}]^{\mathrm{T}}$, with $\mathbf{C}_j = \mathrm{diag}(\mathbf{c}_j, \mathbf{c}_j, \ldots, \mathbf{c}_j) \in \mathbb{C}^{N \times N}$ and $\mathbf{c}_j \in \mathbb{C}^{N_x \times N_y}$. The operator $\mathbf{E} = \mathrm{diag}(\mathbf{E}_1, \ldots, \mathbf{E}_{N_t})$ denotes a composition of 2D radial encoding operators $\mathbf{E}_t$ which for each temporal point $t \in \{1, \cdots, N_t\}$ sample a 2D image $\mathbf{x}_t \in \mathbb{C}^{N_x \times N_y}$ along a non-Cartesian grid in the Fourier domain. In particular, for this work, we consider trajectories given by the radial golden-angle method[28] but point out that other trajectories can be considered as well. By $J = \{1, \ldots, N_{\mathrm{rad}}\}$, we denote the set of indices the k-space coefficients which are needed to sample a 2D image $\mathbf{x}_t$at Nyquist limit. In order to accelerate the image acquisition process, the set $J$is typically constructed by reducing the number of radial lines. We denote by $\mathbf{A}_I$the radial Fourier encoding operator which samples a complex-valued 2D cine MR image $\mathbf{x}$at a radial grid given by the indices in the set $I = I_1 \cup \ldots \cup I_{N_t}$ with $I_t \subset J$ for all $t = 1, \ldots, N_t$. Thus, the considered reconstruction problem is given by

$$\mathbf{A}_I \mathbf{x} + \mathbf{e} = \mathbf{y}_I, \tag{2}$$

where $\mathbf{y}_I = [\mathbf{y}_I^1, \ldots, \mathbf{y}_I^{N_c}]^{\mathrm{T}}$ with $\mathbf{y}_I^c \in \mathbb{C}^{N_t \cdot m_{\mathrm{rad}}}$, for $c = 1, \cdots, N_c$ denotes the measured k-space data for the dynamic 2D cine MR image $\mathbf{x}$and $\mathbf{e} \in \mathbb{C}^{N_k}$with $N_k = N_t \cdot m_{\mathrm{rad}} \cdot N_c$denotes random noise. Multiple receiver coils are typically used in order to achieve $m_{\mathrm{rad}} > N_{\mathrm{rad}}$such that problem (2) is overdetermined and can be solved by considering the normal equations

$$\mathbf{A}_I^{\mathrm{H}} \mathbf{A}_I \mathbf{x} = \mathbf{A}_I^{\mathrm{H}} \mathbf{y}_I. \tag{3}$$

System (3) could in principle be solved by conjugate gradient (CG)-like methods yielding an approximation of the solution given by.

$$\mathbf{x}^* = (\mathbf{A}_I^{\mathrm{H}} \mathbf{A}_I)^{-1} \mathbf{A}_I^{\mathrm{H}} \mathbf{y}_I, \tag{4}$$

where the configuration of the receiver coils ensures that the operator $\mathbf{A}_I^{\mathrm{H}} \mathbf{A}_I$ is invertible. However, for non-Cartesian trajectories, problem (3) is ill-conditioned. Thus, methods used to approximate $\mathbf{x}^*$ can exhibit a semi-convergence behavior and lead to undesired noise amplification.[29] Hence, regularization techniques must be used to stabilize the inversion process.

In this work, we propose a reconstruction network based on a fixed-point iteration which has the form.

$$u_\Theta^M = \underbrace{\left(f_\lambda^{\mathrm{DC}} \circ u_\Theta\right) \circ \ldots \circ \left(f_\lambda^{\mathrm{DC}} \circ u_\Theta\right)}_{M \text{ times}} = \left(f_\lambda^{\mathrm{DC}} \circ u_\Theta\right)^M, \tag{5}$$

where u_Θ is a CNN-block which reduces undersampling artifacts, and noise and f_λ^{DC} is a module which increases data consistency of the intermediate outputs of the CNN-blocks. Limits of Eq. (5) simultaneously satisfy data consistency induced by the DC module and regularity induced by the CNN-block.

For example, in the simplest case where f_λ^{DC} is defined as the minimizer of a penalized least squares functional [see Eq. (14)] and where u_Θ defines a linear projection, the fixed-point iteration Eq. (5) can be shown to converge to the minimizer of the functional.

$$F_{\lambda, \mathbf{y}_I}(\mathbf{x}) = \| \mathbf{A}_I \mathbf{x} - \mathbf{y}_I \|_2^2 + \lambda \| \mathbf{x} - u_\Theta(\mathbf{x}) \|_2^2. \tag{6}$$

Note that, similar to[17] but in contrast to,[13,14,16] the set of parameters Θ is the same for each CNN-block and therefore allows a repeated application of the iterative network u_Θ^M in general. Further, the proposed CNN-block differs from the one in,[17] as we shall discuss later. Figure 1 shows an illustration of the described reconstruction network.

In the following, we describe the CNN-block u_Θ and the DC-module f_λ^{DC} in more detail.

2.B. Proposed CNN-block

In order to keep the notation as simple as possible, we neglect the indices referring to the iteration within the network. We illustrate the CNN-block for the first iteration of the network, that is, when the input of the CNN-block is the initial nonuniform FFT (NUFFT)-reconstruction. The process described below is employed within every CNN block in the proposed network architecture. Let $\mathbf{W}$ denote a diagonal operator which contains the entries of the density compensation function in k-space. By $\mathbf{x}_I := \mathbf{A}_I^{\#} \mathbf{y}_I := \mathbf{A}_I^{\mathrm{H}} \mathbf{W} \mathbf{y}_I$, we denote the initial NUFFT-reconstruction which is obtained by re-gridding the density-compensated measured k-space coefficients onto a Cartesian grid and applying the inverse FFT (IFFT). First, we compute a temporal average of the input over all cardiac phases, that is,

$$\boldsymbol{\nu}_I = \frac{1}{N_t} \sum_{t=1}^{N_t} \mathbf{x}_{I,t} \in \mathbb{C}^{N_x \times N_y} \tag{7}$$

and stack it along the temporal axis, that is, $\boldsymbol{\mu}_I = [\boldsymbol{\nu}_I, \cdots, \boldsymbol{\nu}_I] \in \mathbb{C}^{N_x \times N_y \times N_t}$. Then, we subtract $\boldsymbol{\mu}_I$ from the initial NUFFT-reconstruction $\mathbf{x}_I$ and apply a temporal Fourier transform $\mathbf{F}_t$, that is, we obtain $\mathbf{z} = \mathbf{F}_t(\mathbf{x}_I - \boldsymbol{\mu}_I)$. Subtracting the temporal average from image frames is a well-known and established approach to sparsify image sequences, for example in video-compression (see, e.g., Ref. [30]). Further, applying a temporal FFT provides an even sparser representation and has been used for example in Refs. [18,31] for dynamic MR image reconstruction. Thus, the CNN-block learns to reduce the present undersampling artifacts and noise in a sparse domain. We define the operators $\mathbf{R}^{xt}$ and $\mathbf{R}^{yt}$ which rotate $\mathbf{z}$ by appropriately permuting the relevant axes of the images such that

$$\mathbf{z}^{xt} = \mathbf{R}^{xt} \mathbf{z} \in \mathbb{C}^{N_y \times N_x \times N_t} \tag{8}$$

$$z^{yt} = \mathbf{R}^{yt} \mathbf{z} \mathbb{C}^{(N_x \times N_y \times N_t)}. \tag{9}$$

At this point, $\mathbf{z}^{xt}$ and $\mathbf{z}^{yt}$ can be interpreted as N_y complex-valued images of shape $N_x \times N_t$ and N_x complex-valued images of shape $N_y \times N_t$, respectively. Then, a simple 2D U-Net,[32] which we denote by c_Θ is applied both to $\mathbf{z}^{xt}$ and $\mathbf{z}^{yt}$. Note that we use the same U-Net for both xt- and yt-branches, that is, the weights are shared among the two. This suggestion relies on the assumption of an isotropic resolution in x- and y-direction and the fact that for a radial sampling, the artifacts in xt- and yt-direction do not differ

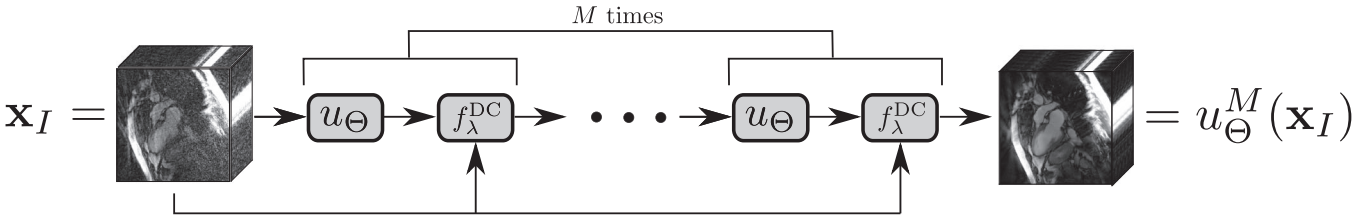

FIG. 1. The proposed reconstruction network alternates between the application of CNN-blocks u_Θ and and data onsistency blocks f_λ^{DC} which increase the consistency of the outputs of the intermediate CNN-blocks by making use of the measured data which are implicitly given by the initial reconstruction $\mathbf{x}_I$.

across the branches. For a Cartesian acquisition, where the latter assumption is clearly not valid, one could simply assign two different CNNs $c_{\Theta_{xt}}$ and $c_{\Theta_{yt}}$ to the two different branches. Also, note that because the U-Net only consists of convolutional and max-pooling layers and has no fully connected layers, the approach can be used for the case $N_x \neq N_y$ as well. Then, we obtain.

$$\mathbf{z}_{\mathrm{CNN}}^{xt} = c_\Theta(\mathbf{z}^{xt}) \tag{10}$$

$$\mathbf{z}_{\mathrm{CNN}}^{yt} = c_\Theta(\mathbf{z}^{yt}). \tag{11}$$

and calculate the estimate $\mathbf{z}_{\mathrm{CNN}}$ by reassembling the processed spatiotemporal slices, that is,

$$\mathbf{z}_{\mathrm{CNN}} = \frac{1}{2}\left((\mathbf{R}^{xt})^{\mathrm{T}}\mathbf{z}_{\mathrm{CNN}}^{xt} + (\mathbf{R}^{yt})^{\mathrm{T}}\mathbf{z}_{\mathrm{CNN}}^{yt}\right). \tag{12}$$

The factor $1/2$ in Eq. (12) is needed because every pixel is used twice — once in the xt-branch and once in the yt-branch. Finally, the estimate $\mathbf{x}_{\mathrm{CNN}}$ is given by applying $\mathbf{F}_t^{\mathrm{H}}$ and adopting a residual connection, that is,

$$\mathbf{x}_{\mathrm{CNN}} = \mathbf{F}_t^{\mathrm{H}}\mathbf{z}_{\mathrm{CNN}} + \boldsymbol{\mu}_I. \tag{13}$$

Figure 2 shows an illustration of the just described procedure. The 2D U-Net employed in our proposed network consists of three encoding stages which are interleaved by max-pooling layers. Each stage consists of a block of 2D convolutional layers with Leaky ReLU as activation function. The number of initially extracted feature maps which is doubled after each max-pooling layer is $n_f = 16$ and is on purpose chosen to be relatively small in order to keep the model's complexity as low as possible. The reason for that is that the 2D U-Net will have to be used together with a (computationally expensive) CG-block, which we describe in the next subsection. In the decoding path, bilinear upsampling followed by a 3×3 convolutional layer with no activation function is used to upsample the images. Our 2D U-Net also involves skip connections from the last to the first layers of the corresponding stages in the encoding and the decoding path. The residual connection is performed in image domain rather than in Fourier domain.

2.C. DC-module

We illustrate the DC-module for the first iteration of the network, that is, where the incoming input of the CNN-block is given by the initial reconstruction. For later iterations, the construction is analogous.

Let us fix $\mathbf{x}_{\mathrm{CNN}} = u_\Theta(\mathbf{x}_I)$ and consider the functional.

$$F_{\lambda,\mathbf{y}_I}(\mathbf{x}) = \| \mathbf{W}^{1/2}(\mathbf{A}_I\mathbf{x} - \mathbf{y}_I) \|_2^2 + \lambda \| \mathbf{x} - \mathbf{x}_{\mathrm{CNN}} \|_2^2, \tag{14}$$

where $\mathbf{W}^{1/2}$ denotes the square root of the diagonal operator which contains the entries of the density compensation function. For the special case where the operator $\mathbf{A}$ is an isometry, problem (14) has a simple closed-form solution (see, e.g., Ref. [13,16]). The involvement of $\mathbf{W}$ in functional (14) can be motivated as a weighting factor which is used for preconditioning the linear system which needs to be solved when minimizing Eq. (14).[33] This aspect is more clearly visible later in Eq. (15). A simple example of $\mathbf{A}$ being an isometry is given by a single-coil acquisition on a Cartesian grid using a simple FFT. In this case, the solution of Eq. (14) can be obtained by performing a linear combination of the measured k-space data $\mathbf{y}_I$ and the one estimated by applying $\mathbf{A}_I$ to $\mathbf{x}_{\mathrm{CNN}}$. See, for example, Ref. [13], for more details. However, in the more general case, a minimizer of Eq. (6) is obtained by solving a system of linear equations. By setting the derivative of Eq. (14) with respect to $\mathbf{x}$ to zero, it can be easily seen that the system one needs to solve is given by $\mathbf{H}\mathbf{x} = \mathbf{b}$ with

$$\begin{aligned} \mathbf{H} &= \mathbf{A}_I^{\#}\mathbf{A}_I + \lambda\mathbf{I}_N, \\ \mathbf{b} &= \mathbf{A}_I^{\#}\mathbf{y}_I + \lambda\mathbf{x}_{\mathrm{CNN}}. \end{aligned} \tag{15}$$

System (15) can be solved by means of any iterative algorithm and, since the operator $\mathbf{H}$ is symmetric, an appropriate choice is the conjugate gradient (CG) method.[34] Note that functional Eq. (14) is linear in $\mathbf{x}$ and therefore, due to strong convexity, solving Eq. (15) leads to the unique minimizer of Eq. (14). In practice, as we discuss later in the implementation details, the DC-module is an implementation of a finite number of iterations of the CG method. We denote the number of such iterations by n_{CG}. Note that in the CG method, the operator $\mathbf{H}$ has to be applied at each iteration. In addition, note that in general, the application of $\mathbf{A}_I$ as well as $\mathbf{A}_I^{\#}$ can be computationally way more expensive than for a simple FFT, for example, if the gridding of the k-space coefficients is part of the operators as in our case. Thus, it is desirable to have a CNN-block with as few trainable parameters as possible such that end-to-end training of the entire network is possible in a reasonable amount of time.

2.D. Training scheme

Because the solution of Eq. (15) has to be approximated using an iterative scheme which employs the

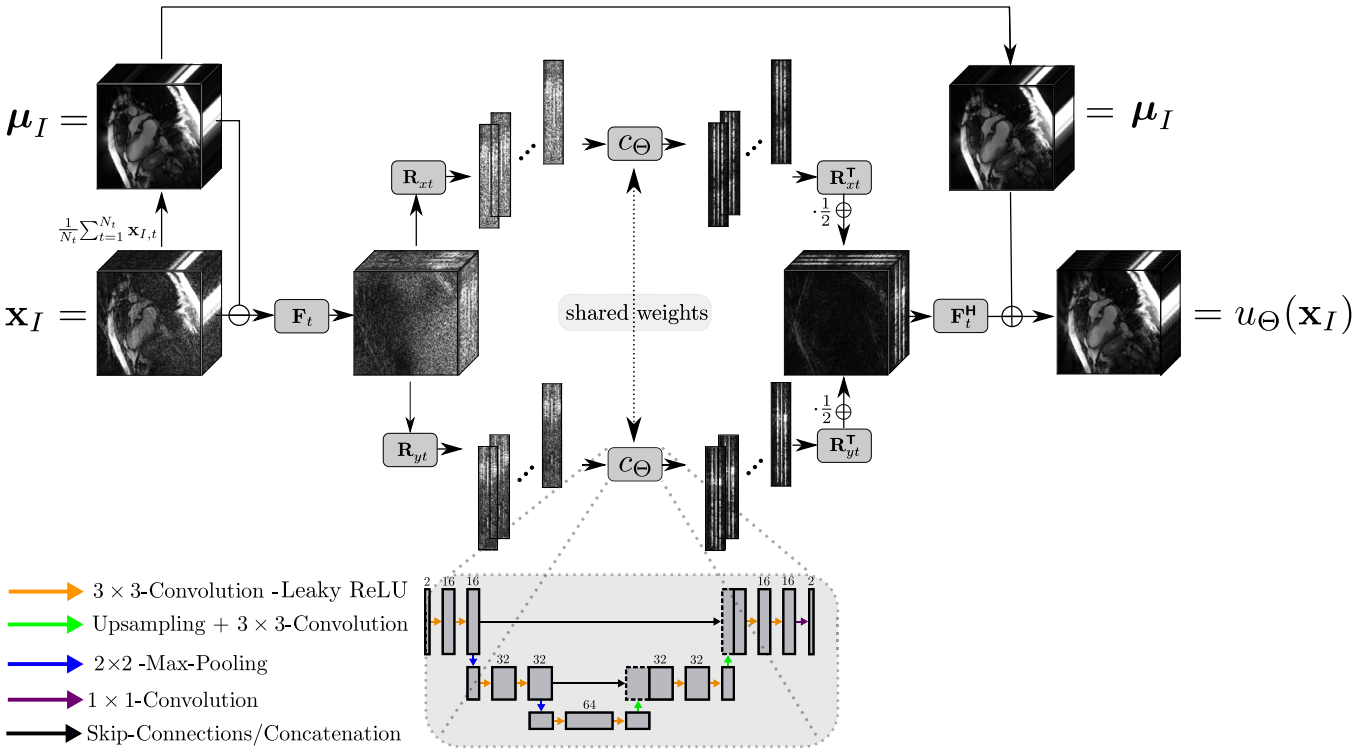

FIG. 2. The proposed CNN-block. First, the input image $\mathbf{x}_I$ is Fourier transformed along the temporal axis, then the Fourier-transformed image is reshaped into the *xt*- and *yt*-domain. The same 2D U-Net u_Θ is applied to each of the 2D slices. Then, the estimate in the temporal Fourier domain is calculated by reassembling the processed slices and the output is obtained by applying the inverse temporal Fourier transform and summing it up to the input image. [Color figure can be viewed at wileyonlinelibrary.com]

(computationally expensive) application of the operator $\mathbf{H}$ at each iteration, end-to-end training of the entire reconstruction network from scratch would be time-consuming. Thus, we circumvent this issue by the following more efficient training strategy.

First, in a pretraining step, we only train a single CNN-block on image pairs as is typically done in deep learning-based post-processing methods. More precisely, we minimize the L_2-norm of the error between the output of the CNN-block which was predicted from the initial reconstruction $\mathbf{x}_I$ and its corresponding label. Then, in a second training stage, we construct a network as in Eq. (5) and initialize each of the CNN-blocks by the previously obtained parameter set Θ and perform a further fine-tuning of the entire network by end-to-end training. Further, the regularization parameter λ contained in each of the CG-blocks f_λ^{DC} can be included in the set of trainable parameters as well and is trained to find the optimal strength of the contribution of the regularization term. This means that it implicitly learns to estimate the noise level present in the measured *k*-space data for the whole dataset.

2.E. Experiments with in-vivo data

To evaluate the efficacy of our proposed approach, we performed different experiments. First, we investigated the effect of our proposed training strategy. We also compared the proposed network with different configurations of hyper-parameters M and n_{CG}. Finally, we further compared our proposed approach to several other methods for non-Cartesian cardiac cine MR image reconstruction. In the following, we provide the reader with information about the used dataset, the methods of comparison and some details on the implementation of the proposed method.

2.F. Dataset

We used a dataset of cine MR images of $n = 19$ subjects (15 healthy volunteers + 4 patients). For each healthy volunteer as well as for two patients, $N_z = 12$ different orientations of cine MR images were acquired. For the resting two patients, only $N_z = 6$ slices were acquired due to limited breathhold capabilities. Thus, we have a total of 216 complex-valued cine MR images. We split the dataset in 144/36/36 images used for training, validation, and testing, where for the test set, we use the 36 cine MR images of the patients in order to be able to qualitatively assess the images with respect to clinically relevant features.

All images were acquired using a balanced steady-state free precession (bSSFP) sequence on a 1.5 T MR scanner during a single breathhold of 10 s (repetition time/echo time = 3.0/1.5ms, flip angle 60°). The images used as ground-truth data for the retrospective undersampling were reconstructed from the *k*-space data which were sampled along $N_\theta = 3400$ radial spokes with *kt*-SENSE.[35] The coil-sensitivity maps, which were calculated from low-resolution images obtained from the central part of the radial k-space data, were used to combine the different images to a single

image after the NUFFT-reconstruction. The images have a field of view of $N_x \times N_y = 320 \times 320$ mm^2 with an in-plane resolution of 2 mm^2 and a slice thickness of 8 mm. The number of acquired cardiac phases was $N_t = 30$. A 32-channel cardiac-phased array coil was used for signal reception and parallel image reconstruction. From these images, we retrospectively simulated k-space data by sampling $N_\theta = 560$ and $N_\theta = 1130$ radial spokes. Since sampling along $N_\theta = 3400$ already corresponds to an acceleration factor of approximately ~3 (with respect to the Nyquist limit), which was needed to perform the scan during one breathhold, $N_\theta = 560$ and $N_\theta = 1130$ correspond to acceleration factors of ~18 and ~9, respectively.

Further, the k-space data were corrupted by normally distributed noise with standard variation $\sigma = 0.02$ which was added to each the real and imaginary parts of $\mathbf{y}_I^c$ for $c = 1, \cdots, 12$ after having centered them. The calculation of the density compensation function for a fixed set of trajectories was based on partial Voronoi diagrams.[36]

2.G. Quantitative measures

We evaluated the performance of our proposed network architecture in terms of different error- and image-similarity-based measures. The peak signal-to-noise ratio (PSNR) and the normalized root mean squared error (NRMSE) are used as error-based measures. Further, we report a variety of different similarity-based measures: the structural similarity index measure (SSIM),[37] the multi-scale SSIM (MS-SSIM),[38] the universal image quality index (UQI),[39] the visual information quality measure (VIQM),[40] and the Haar wavelet-based similarity index measure.[41]

We calculate all measures by comparing the 2D complex-valued images in the xy-plane for each time point. This means that our test set consists of 1080 2D images. For the calculation of the similarity measures, the real and imaginary part of the images are treated as channels. The similarity-based measures are calculated for each channel separately and then averaged across the two channels. The statistics were calculated over a region of interest of 160×160 pixels in order to discard background noise. Further, we segmented the patients in the images of the test set in order for the statistics to reflect the achieved performance on regions of interest.

2.H. Implementation details

The architecture was implemented in PyTorch. Complex-valued images were stored as two-channeled images. The forward and the adjoint operators $\mathbf{A}_I$ and $\mathbf{A}_I^{\mathrm{H}}$ were implemented using the publicly available library Torch KBNUFFT[42,43] which also allows to perform back-propagation across the forward and adjoint operators. During training, the k-space trajectories, the coil-sensitivity maps, and the density compensation functions were stored as tensors for the implementation of $\mathbf{A}_I$ and $\mathbf{A}_I^{\mathrm{H}}$. Note that we make the regularization parameter λ trainable such that a trade-off between the measured k-space and the output of the CNN-blocks is learned. In order to constrain λ to be strictly positive, during the fine-tuning of the model, we apply a Softplus activation function, that is, we set $\lambda := \mathrm{SoftPlus}(\tilde{\lambda}) = \frac{1}{\beta}(\log(1 + \exp(\beta\tilde{\lambda}))$, which maps $\tilde{\lambda}$ to the interval $(0, \infty)$. We used the default parameter $\beta = 1$. Note that a CG scheme is usually stopped after a certain stopping criterion is met. A commonly used stopping criterion is if the norm of the newly calculated residual $\mathbf{r}_k = \mathbf{H}\mathbf{x}_k - \mathbf{b}_k$ is small enough, that is, $\| \mathbf{r}_k \|_2 \leq \mathrm{TOL} \| \mathbf{b} \|_2$ for a tolerance TOL chosen by the user. During fine-tuning the iterative network, we fix the number of CG iterations n_{CG} but when testing the network on unseen data, we can choose to use the number of iterations the CNN was fine-tuned with or set an own stopping criterion. All experiments were performed on an NVIDIA GeForce RTX 2080 with 11 GB memory.

2.I. Comparison with other methods

We compared our proposed approach to the following methods which employ recently published learned and well-established non-learned regularization methods. As well-established reconstruction methods, we applied iterative SENSE,[44] a Total Variation (TV)-minimization method,[45] and kt-SENSE.[31] Further, we compared our proposed method to a method based on dictionary learning (DL) and sparse coding (SC)[46,47] using adaptive DL and adaptive SC[48] and a method which employs CNN-based regularization in the form of previously obtained image priors,[9,10] where we used a previously trained 3D U-Net[6] for obtaining the prior.

Note that on purpose we did not compare our proposed approach with other CNN-based methods involving generative adversarial networks (GANs). The reason is that we are mainly interested in the performance of the combination of the proposed CNN-block in terms of artifacts-reduction as well as the trade-off between employing a CG-module or not. In addition, note that if the hardware allows it, it is always possible to add different components based on GANs to regularize the output of the CNN-blocks. Further, note that although there exist several other state-of-the-art methods using cascaded/iterative networks for dynamic cine MRI, see for example Ref. [13,16,18] the underlying reconstruction problem is a different one (single-coil and Cartesian vs. multi-coil and radial) and thus these methods are not directly applicable as originally published.

3. RESULTS

In the following, we report the obtained results concerning the training behavior and the reconstruction performance of our proposed method.

3.A. Computational complexity of the forward and adjoint operators

Here, we evaluated the computational complexity of the proposed network architecture in terms of required GPU memory as well as training times which can be estimated for

the end-to-end training stage. Here, we fixed the number of iterations of the network to be $M = 1$ and the CNN-block was fixed to be the identity $c_\Theta = \mathbf{I}_N$, that is, it contains no trainable parameters. Thus, the allocated memory can be mainly attributed to the tensors needed for the radial trajectories, the density compensation, the coil-sensitivity maps which define the operator, and the considered input image.

Figure 3 shows the allocated GPU memory as well as the required time to perform one weights-update by performing a forward- and a backward-pass through the entire reconstruction network. By N_θ we denote the total number of radial spokes which are acquired for each of the N_t cardiac phases. The first row of Fig. 3 shows the average allocated GPU memory as well the required time to perform one weights-update depending on the spatial image size. This is shown for $n_{\mathrm{CG}} = 1$ and different numbers of radial spokes N_θ. It can be seen that already for $n_{\mathrm{CG}} = 1$, the required GPU memory amounts to approximately 512–1024 GB and performing one step of back-propagation is in the range of 4 s. The second row of Fig. 3 shows the same quantities for fixed $N_\theta = 560$ and different n_{CG}. Here, we also see that employing a relatively high number of CG iterations, say $n_{\mathrm{CG}} = 12$, almost requires 2 GB of GPU memory and, more importantly, requires more than 30 s. Training the entire network in this configuration for 100 epochs would, for example, already require 5 days. By that, one can estimate that training the entire network from scratch in an end-to-end manner could easily amount to weeks or months. Further, the required time does not vary much for different image sizes, meaning that even for relatively small reconstruction problems, the application of $\mathbf{A}_I$ and $\mathbf{A}_I^{\mathrm{H}}$ is inherently computationally demanding.

These computational aspects highlight the importance of the efficacy of the CNN-block in terms of having a small number of parameters to ensure fast convergence during training and at the same time a good performance in terms of undersampling artifacts-reduction.

3.B. Efficacy of the training scheme

Here, we show the impact of including the forward and the adjoint operators $\mathbf{A}_I$ and $\mathbf{A}_I^{\mathrm{H}}$ in the network architecture during the learning process. Figure 4 shows the training and validation error of the proposed CNN-module during the pretraining stage. In the first pretraining stage, only one single CNN-block was trained to minimize the L_2-error between the output estimated by the CNN-block u_Θ and its corresponding label. In the fine-tuning stage, a CG-module with $n_{\mathrm{CG}} = 8$ iterations was attached to the CNN-block. As can be seen, in the pretraining stage, after about 65 000 weight updates (which corresponds to approximately 450 epochs using a mini-batch size of one), training and validation error start to stagnate between 0.025 and 0.30, respectively. After

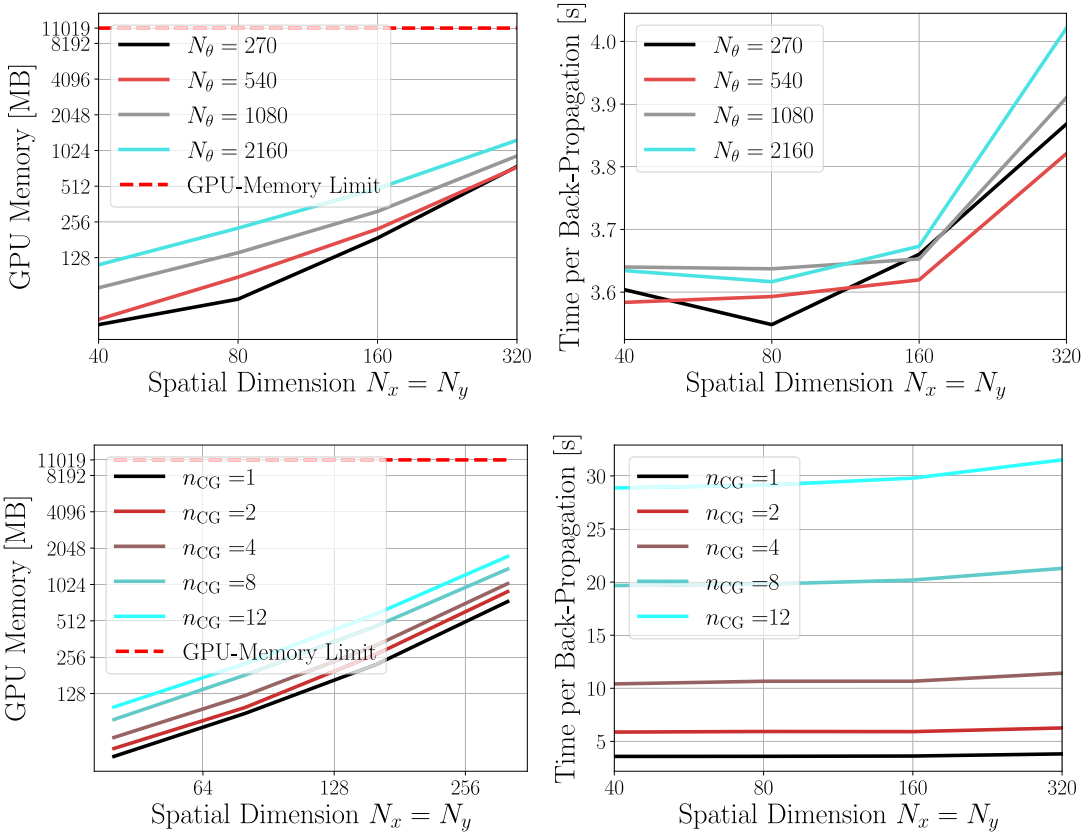

FIG. 3. Computational complexity and the time required for one backward pass for the operator $\mathbf{H}$ which is employed in the DC-module for different spatial image sizes $N_x \times N_y$. The number of temporal points was set to $N_t = 30$ and the number of coils was $N_c = 12$ [Color figure can be viewed at wileyonlinelibrary.com]

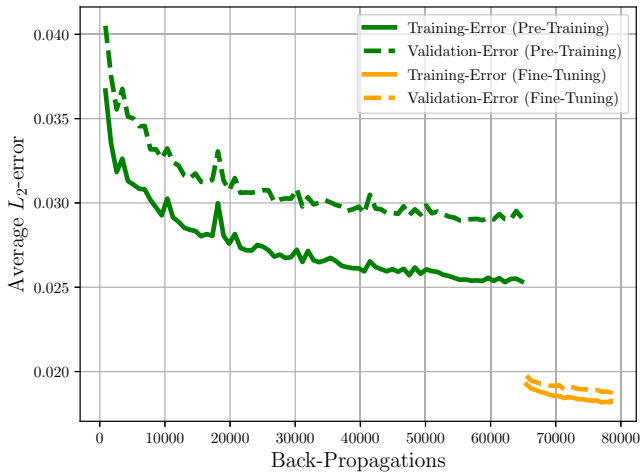

FIG. 4. Training behavior of the proposed network architecture. The lines indicate the L_2-error on the training data (solid lines) as well as on the validation data (dashed lines) during pretraining of the CNN-block (green lines) and fine-tuning of the entire network (orange lines). As can be seen, in the fine-tuning stage, where the physical models $\mathbf{A}_I$ and $\mathbf{A}_I^{\mathrm{H}}$ are included in the network architecture, the error is further reduced on both the training and the validation data. Further, the gap between training and validation errors becomes smaller. [Color figure can be viewed at wileyonlinelibrary.com]

pretraining, the parameter set Θ was stored and the fine-tuning of the entire network was carried out by initializing the set of parameters Θ as the one obtained after pretraining. As can be seen from the orange curves, including the GC-module which employs the encoding operators in the CNN architecture allowed to further reduce the training as well as the validation error and therefore to obtain a more suitable parameter set Θ.

Note that since the fine-tuning stage is much more computationally demanding than the pretraining stage, we only trained for 150 epochs and tested on the whole training and validation dataset only 12 times instead of 75 times as in the pretraining stage. The time for pretraining the CNN-module amounted to approximately 6 hr, while fine-tuning the entire network took approximately 5 days.

Figure 5 shows the effect of the proposed training strategy on the obtained results. Figure 5(e) shows the initial NUFFT-reconstruction which is directly obtained from the undersampled k-space data. After having pretrained the single block u_Θ, the image in Fig. 5(a) is obtained by passing the input to the CNN-block. Further, proceeding in the reconstruction network with the CG-bock yields the image shown in Fig. 5(b) for which the point-wise error is lower than for (a). Figure 5(c) shows the output of the CNN-block after having fine-tuned the entire network architecture, that is, a CNN-block with attached CG-block, by end-to-end training. By comparing (c) to (a), we see that the quality of the output of the CNN-block has clearly increased as the point-wise error has been further reduced. Further, proceeding with the CG-module in the network further reduces the point-wise errors as can be seen in (d). Again, by comparing the final reconstructions (d) and (b), we see that the quality of the reconstruction has further increased as the point-wise error has decreased. Note that by taking in consideration the stagnation of the training and validation error shown in Fig. 4, we can indeed attribute the increase in performance of the reconstruction method to the fact that we included the forward and adjoint operators in the training process and rather than to the additionally performed weight-updates. Further, we have experimentally confirmed the efficacy of the proposed training procedure.

Table I lists the quantitative measures obtained on the test set for the intermediate output of the CNN-block as well as for the final reconstruction before and after fine-tuning the entire network for two different acceleration factors. The table well-reflects the visual results in the sense that the intermediate output of the CNN as well as the final reconstruction after the fine-tuning stage surpass their corresponding image estimates after the sole pretraining in terms of all reported measures.

3.C. Variation of the hyper-parameters M and n_{CG}

As already mentioned, at test time, one does not necessarily need to stick to the configuration of the network in terms of length M and number of CG-iterations n_{CG} which were used for fine-tuning the network. In particular, for the fine-tuning stage, the choice of M and n_{CG} is mainly driven by factors as training times and hardware constraints which do not play a role at test time.

Thus, we performed a parameter study where at test time, we varied the length of the network M as well as the number of CG-iterations n_{CG}. We repeated these experiment for two different configurations of our proposed network. We fine-tuned one network with $M=3$ and $n_{\mathrm{CG}}=2$ and another one with $M=1$ and $n_{\mathrm{CG}}=8$. Due to hardware constraints, the networks also differ in terms of number of trainable parameters of the CNN-block.

At test time, we fixed $n_{\mathrm{CG}}=4$ and varied M from $M=2,4,\cdots,12$. With this configuration, the noise and the artifacts are gradually reduced and data consistency of the solution is increased several times during the whole reconstruction process. In Tables S1 and S2 in the supplementary material, one can see that increasing M consistently further improved the results for both networks. Further, the comparison of the two networks in Fig. S1 in the supplementary material shows that the network containing more trainable parameters (i.e., the one which was fine-tuned with $M=1$ and $n_{\mathrm{CG}}=8$) surpasses the other one with respect to all measures. This observation motivates the choice of the final reconstruction network used for comparison with other methods in the next subsection.

3.D. Reconstruction results

Here, we compare our reconstruction method with the image reconstruction methods previously introduced. As discussed in Section 3.C we chose to use $M=12$ and $n_{\mathrm{CG}}=4$ although the network was trained with $M=1$ and $n_{\mathrm{CG}}=8$. Note that, since the same strategy seemed not to be consistently useful for the 3D U-Net which was trained without the integration of the encoding operators, we report here the

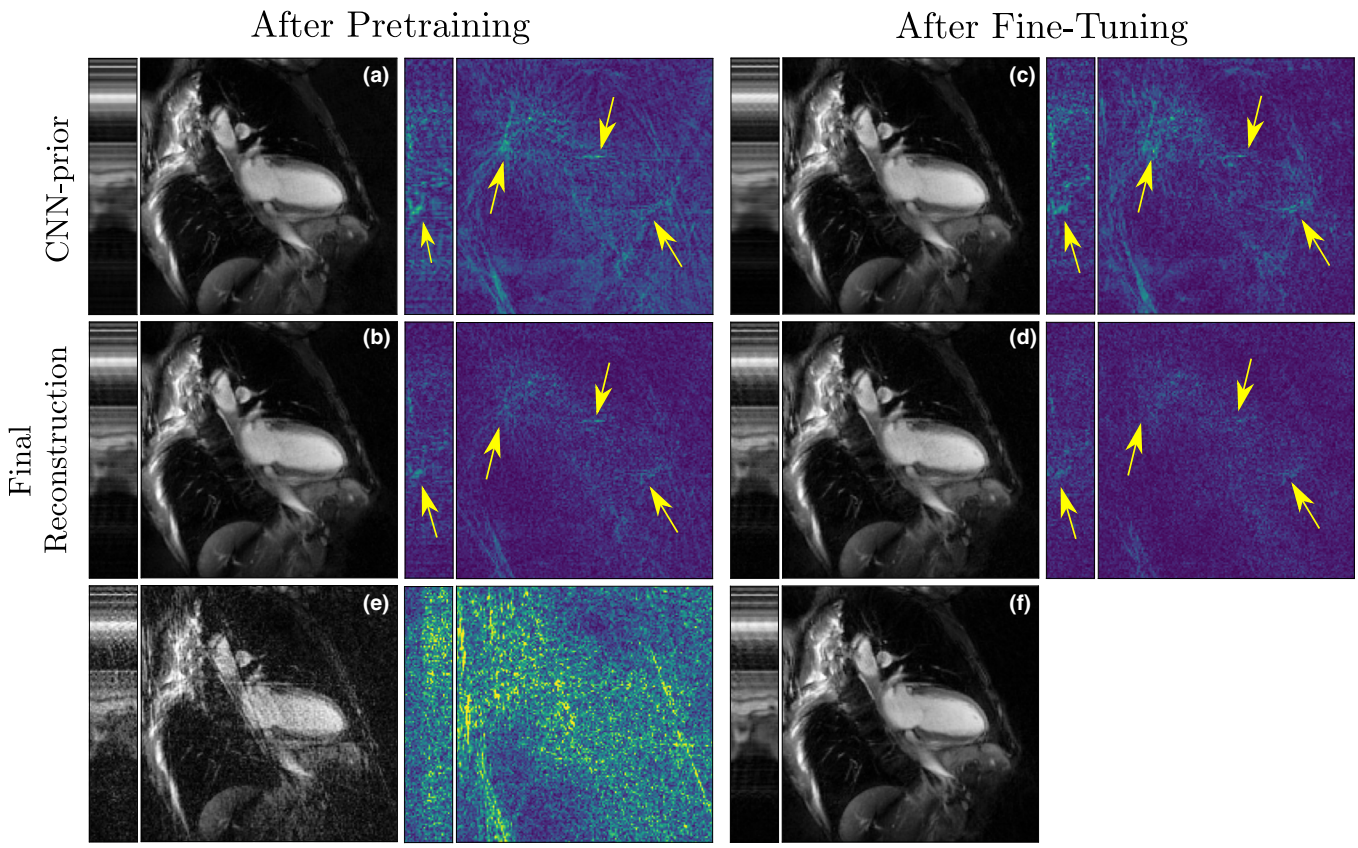

FIG. 5. Intermediate results and point-wise error images of our reconstruction method after pretraining and after fine-tuning our proposed method with $M = 1$ and $n_{\mathrm{CG}} = 8$. The output of the CNN-block after pretraining (a), the final reconstruction (b) after pretraining where (a) was the output of the CNN-block, the output of the CNN-block after fine-tuning the entire cascade in an end-to-end manner (c) and the final reconstruction after fine-tuning (d) where (c) was the output of the CNN-block, the initial NUFFT-reconstruction $\mathbf{x}_I$ obtained from $N_\theta = 560$ radial spokes (e) and the corresponding ground-truth image obtained by *kt*-SENSE with $N_\theta = 3400$ radial spokes (f). We see that after having fine-tuned the entire network, the point-wise error of the final reconstruction is the smallest. Further, the cardiac motion is much better preserved as is pointed out by the yellow arrows. [Color figure can be viewed at wileyonlinelibrary.com]

values for $M = 1$ and $n_{\mathrm{CG}} = 12$ which are also the ones used in Ref 10. In the supplementary material, the results for the variation of M and n_{CG} can be found as well. The regularization parameter for the dictionary learning method[48] and the CNN-based regularization method[10] was chosen as $\lambda = 1$.

Figure 6 shows an example of the results obtained with the previously described methods of comparison and our proposed approach. As can be seen, the total variation-minimization method as well as *kt*-SENSE successfully removed the undersampling artifacts but also led to a loss of image details as is indicated by the yellow arrows. In contrast, all learning-based method yielded a good reconstruction performance in terms of preservation of the cardiac motion. We see that our proposed method is in addition the one which best reduced residual image noise in the images. Table II shows the results achieved in terms of the chosen measures. The best achieved results are highlighted as bold numbers. Again, the experiments were repeated for two different undersampling (i.e., acceleration) factors, given by sampling the *k*-space along $N_\theta = 560$ and $N_\theta = 1130$ radial spokes, respectively.

The numbers well-reflect the observations from Fig. 6 and we see that all methods based on regularization methods employing learning-based methods consistently outperform the methods using hand-crafted regularization methods with respect to all measures. All three reported methods using machine learning-based regularization achieve competitive results, where we see that our proposed method yields substantially better results compared to the dictionary learning (DL)-based method and the other CNN-based method in terms of error-based measures. In terms of image similarity-based measures, the difference between the dictionary learning-based method and ours becomes less prominent except for VIQP. Interestingly, the 3D-U-Net-based method is consistently surpassed either by the dictionary learning-based method or our proposed one. All observations are consistent among both acceleration factors with $N_\theta = 560$ and $N_\theta = 1130$. In addition, note that while the obtained results for DL, the 3D U-Net-based iterative reconstruction and our proposed approach are similar, the average reconstruction time using DL amounts to approximately 2000 s, where the most computationally intensive part is the repeated sparse coding of the image patches at each iteration. However, the implementation of the dictionary learning and sparse coding algorithms aITKrM and aOMP is currently only available for running on the CPU and thus, a further acceleration could be expected. In contrast, for the CNN-based methods, the

TABLE I. Intermediate and final reconstruction results after pretraining of the CNN-block and after fine-tuning of the entire network architecture with $M = 1$ and $n_{\mathrm{CG}}. = 8$. The results are shown for $= N_\theta 560$ and $= N_\theta 1130$ radial spokes.

	After pretraining		After fine-tuning	
	CNN-prior	Final reconstruction	CNN-prior	Final reconstruction
Number of Radial Spokes:$N_\theta = 560$				
PSNR	42.4541	45.4826	44.484	**46.0036**
NRMSE	0.1252	0.0874	0.0963	**0.0810**
SSIM	0.9576	0.9796	0.9833	**0.9872**
MS-SSIM	0.9916	0.9966	0.9965	**0.9977**
UQI	0.8027	0.8776	0.9064	**0.9275**
VIQP	0.9431	0.9506	0.9429	**0.9434**
HPSI	0.9889	0.9941	0.9901	**0.9943**
Number of Radial Spokes:$N_\theta = 1130$				
PSNR	43.9893	47.0087	46.2383	**47.7545**
NRMSE	0.1044	0.0735	0.0808	**0.0673**
SSIM	0.9698	0.9854	0.9873	**0.9903**
MS-SSIM	0.9945	0.9977	0.9974	**0.9984**
UQI	0.8324	0.8998	0.9209	**0.9408**
VIQP	0.9574	0.9642	0.9604	**0.9626**
HPSI	0.9916	0.9961	0.9930	**0.9962**

reconstruction time amounts to approximately 12 and 48 s which are mainly coming from the CG-module. Since one iteration in the CG-block takes approximately 1 s mainly because of the evaluation of the operator $\mathbf{H} = \mathbf{A}_I^{\#}\mathbf{A}_I + \lambda \mathbf{I}_N$, the overall reconstruction time of our proposed method can be estimated as $t_{\mathrm{REC}} \approx M \cdot n_{\mathrm{CG}}$. Further, note that our proposed method quite consistently surpasses the 3D U-Net-based reconstruction method even if it only contains around 9.1% of the trainable parameters.

4. DISCUSSION

In this work, we have proposed the first end-to-end trainable iterative reconstruction network for dynamic multi-coil MR image reconstruction using nonuniform sampling schemes. As we have seen, the proposed end-to-end trainable reconstruction network provides a competitive method for image reconstruction for 2D cine MR image reconstruction with non-Cartesian multi-coil encoding operators. In the following, we highlight several advantages and limitations of our proposed approach and put it in relation to other works.

4.A. End-to-end trainability

Since the considered forward model is computationally demanding, methods as[9,10] can be used as an alternative, where the generation of the CNN output and the step of increasing data consistency are decoupled from each other. However, the major advantage of our proposed network over methods similar to Refs. [9] or [10], is that the reconstruction network is trained in and-end-to-end manner. As can be seen in Fig. 4 in Section 3.B, including the DC-module given by the CG method in the network architecture is highly beneficial since it further reduces the training and validation error and also reduces the gap between the both, leading to a better generalization power. This result experimentally confirms the theory on the achievable performance of the reconstruction network derived in.[12] Further, we demonstrated that the proposed training strategy is a viable option for training the entire network in an end-to-end manner in a relatively short amount of time while achieving good convergence properties of the network parameters.

4.B. The choice of the CNN-block

As we have seen in Fig. 4, the proposed CNN-block's trainable parameters converge relatively fast (approximately 6 hr) to a set of suitable trainable parameters in the pretraining stage. Training the 3D U-Net, in contrast, took approximately 3 days. Because the proposed CNN-block is trained relatively fast, fine-tuning the entire network is possible by investing approximately 5 days of further training.

Note that our proposed approach transfers the learning of the artifact-reduction mapping to from 3D to 2D by the application of 2D convolutional layers in the temporally Fourier-transformed spatiotemporal domain. Thus, it inherits all benefits method presented in Ref. [7]. The most important property is that due to the change of perspective on the data, for each single cine MR image, $N_x + N_y$ samples are actually considered to train the CNN-block, which is essential for training. Since only a relatively small amount of training data in terms of number of subjects already suffices for a successful training, the end-to-end-training, which is particularly computationally expensive during the fine-tuning stage, can be carried out in a reasonable amount of time without the need to additionally augment the dataset in order to prevent overfitting.

Note that, of course, if enough training data are available and the hardware constraints allow it, training the network with computationally heavier CNN-blocks is possible as well. Thus, the proposed method can also be seen as a general method for training an end-to-end reconstruction network for large-scale image reconstruction problems with computationally demanding forward operators.

4.C. The trade-off between the hyper-parameters M and n_{CG}

From the formulation of the network architecture in Eq. (5), we can identify two main hyper-parameters which can be varied and determine the nature of the proposed reconstruction algorithm. The overall number of iterations M defines the length of the unrolled iterative scheme. Further, because in general, minimizing functional (14) requires solving a linear system using an iterative solver, the number of iterations to approximate the solution of Eq. (14), here named n_{CG}, has

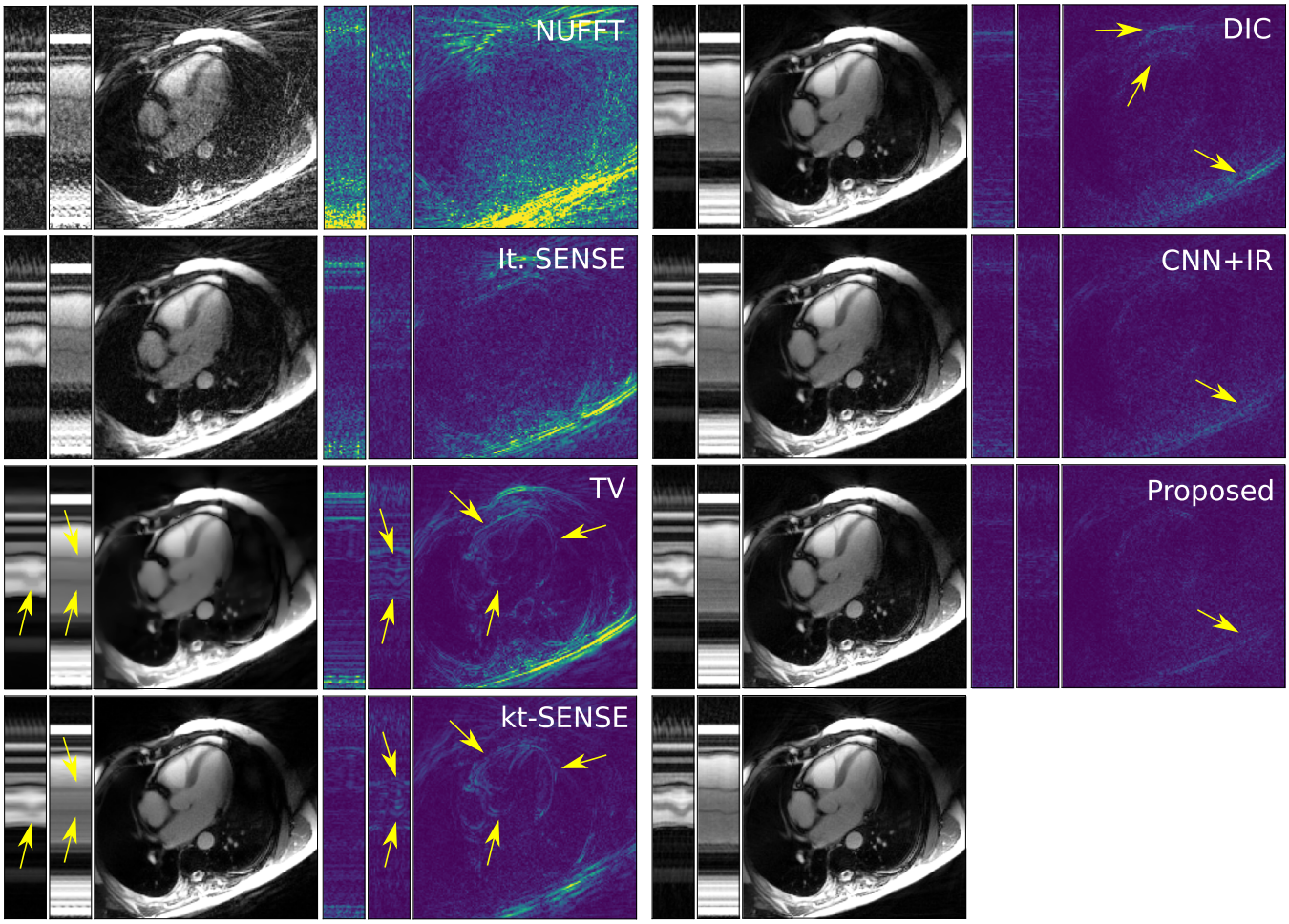

FIG. 6. Comparison of images and point-wise errors for different reconstruction methods with learned and non-learned regularization for $N_\theta = 560$. Left column: Classical iterative reconstruction methods with — from top to bottom — direct NUFFT-reconstruction, iterative SENSE,[44] total variation-minimization (TV),[45] *kt*-SENSE.[31] Right column: Learning-based regularization methods with — from top to bottom — adaptive dictionary learning and sparse coding (DIC),[48] CNN-based iterative reconstruction (CNN + IR)[10] using a 3D U-Net,[6] and our proposed approach. The yellow arrows in the images of the left column point at errors occurring at regions of the heart which might affect the assessment of the cardiac motion while the ones in the right column point at residual image noise. [Color figure can be viewed at wileyonlinelibrary.com]

to be chosen as well. By setting $M = 1$ and n_{CG} relatively "high," say $n_{\mathrm{CG}} = 12$, one aims at constructing an end-to-end trainable network which conceptually resembles methods like.[9,10] There, the CNN-block is only applied once to obtain a CNN-based image-prior and functional (14) is minimized until convergence of the iteration. In contrast, one can also set $M > 1$, but because of hardware constraints, one necessarily also has to either lower n_{CG} in each CG-block, reduce the complexity of the CNN-blocks, or in the worst case, both. Lowering the number of CG-iterations n_{CG} causes the solution of Eq. (14) most probably to only be poorly approximated and lowering the CNN-blocks complexity can be expected to deliver poorer intermediate outputs of the different CNN-blocks in terms of artifacts-reduction.

Interestingly, we have observed that even fine-tuning the entire network with $M = 1$ and $n_{\mathrm{CG}} = 8$ allowed to change the hyper-parameters at test time by further obtaining a boost in performance, see the supplementary material for more details. We also trained an iterative network architecture with our proposed CNN-block for $M = 3$ and $n_{\mathrm{CG}} = 2$. Due to hardware constraints, we employed smaller 2D U-Nets as CNN-blocks, where, different to before, we set the initial number of applied filters to $n_f = 4$. The so-constructed network consisted of only 5908 trainable parameters, that is, only about 0.57% of the 3D U-Net.

In the supplementary material, one can see a comparison of our method fine-tuned with $M = 1$ and $n_{\mathrm{CG}} = 8$ against $M = 3$ and $n_{\mathrm{CG}} = 2$ which were evaluated with different configurations of M and n_{CG} at test time. The network which was fine-tuned with $M = 1$ and $n_{\mathrm{CG}} = 8$ consistently outperforms the other with respect to all measures. Altough the comparison is not entirely fair since the number of trainable parameters highly differs from one CNN-block to the other, this result is important because of the following reason. It suggests that sacrificing expressiveness of the CNN-block in terms of trainable parameters in order be able to fine-tune with $M > 1$ seems not to be necessary since also for the network fine-tuned with $M = 1$ and $n_{\mathrm{CG}} = 8$, different M and n_{CG} can be used at test time and further improve the results. Interestingly, we were not able to observe this phenomenon for the 3D U-Net as can be seen from Table S3 in the

TABLE II. Quantitative results for $N_\theta = 560 N_\theta = 1130$ radial spokes which corresponds to an acceleration factor of ~9 and ~18, respectively.

	Non-learned regularization				Learned regularization		
	NUFFT	It SENSE	TV	*kt*-SENSE	DL	3D U-Net + IR	Proposed
Number of radial spokes:$N_\theta = 560$							
PSNR	34.9542	40.5610	41.8778	44.2316	45.6085	46.0845	**47.4396**
NRMSE	0.2913	0.1535	0.1313	0.0986	0.0844	0.0800	**0.0697**
SSIM	0.8843	0.9686	0.9786	0.9820	0.9885	0.9880	**0.9895**
MS-SSIM	0.9762	0.9935	0.9950	0.9958	0.9980	0.9979	**0.9982**
UQI	0.7574	0.8827	0.8889	0.8887	0.9347	0.9288	**0.9388**
VIQP	0.8339	0.8379	0.8234	0.8931	0.9256	0.9340	**0.9620**
HPSI	0.9456	0.9811	0.9849	0.9904	0.9950	0.9949	**0.9961**
Number of radial spokes:$N_\theta = 1130$							
PSNR	39.0289	43.8077	44.1841	46.0096	47.4373	47.5094	**48.6761**
NRMSE	0.1864	0.1067	0.0998	0.0817	0.0682	0.0681	**0.0606**
SSIM	0.9378	0.9778	0.9841	0.9875	0.9913	0.9906	**0.9916**
MS-SSIM	0.9892	0.9965	0.9965	0.9976	**0.9986**	0.9985	**0.9986**
UQI	0.8363	0.9156	0.9028	0.9224	**0.9484**	0.9420	0.9480
VIQP	0.9228	0.9513	0.8832	0.9498	0.9521	0.9547	**0.9695**
HPSI	0.9775	0.9923	0.9903	0.9943	**0.9972**	0.9966	**0.9972**
Parameters	–	–	–	–	**9 664**	1 024 224	93 617
Backend	GPU	CPU	CPU	CPU	CPU/GPU	GPU	GPU

supplementary material. Thus, we attribute this property to the fine-tuning stage in which the encoding operator is included in the network architecture which again highlights the importance of employing a CNN-block which allows end-to-end training of the entire network in a reasonable amount of time.

4.D. Limitations

For the proposed method, training times tend to be quite long, amounting to several days. This makes the algorithmic development challenging in terms of hyper-parameter tuning and limits the possibility to draw general conclusions because repeating experiments for different parameter configurations is prohibitive. Nevertheless, the proposed training scheme shows that it is easily possible to outperform post-processing methods as in[6,7] or the so-called model-agnostic approaches in Refs. [9,10] where training of the CNN-block is decoupled from the subsequent minimization of a CNN-regularized functional.

Further, although we have seen that it is possible to increase M at test time and observe an increase in performance in terms of reconstruction results, from a theoretical point of view, it remains somewhat unclear why this is exactly possible. We believe that the reason for this lies in the end-to-end training the entire network but leave a rigorous theoretical investigation and a convergence analysis as future work.

4.E. Differences and similarities to other works

Our presented approach shares similarities across different works. First, for the case $M = 1$, it can be seen as an extension of the approaches presented in Refs. [7,9,49] in the sense that we only generate only one CNN-based image-prior which is used in a Tikhonov functional. However, because of the proposed end-to-end training strategy, the obtained CNN-based image-prior tends to be a much better estimate of the ground-truth image compared to the cases when the training of the CNN-block is decoupled. Further, for $M > 1$, the structure of the network is similar to,[13,17] with the difference that first of all, the considered inverse problem is different (radial multi-coil instead of Cartesian single-coil) and thus the DC-module is a CG-block instead of the implementation of a closed-form solution to Eq. (14). Second, our CNN-block consists of a spatiotemporal 2D U-Net which is applied in the Fourier-transformed spatiotemporal domain.

In our work, the 2D U-Nets are applied in the temporally Fourier-transformed spatiotemporal domain and thus use the same change of perspective on the data as in.[7] However, in Ref. [7] the slices extraction and reassembling process is not part of the network architecture and thus does not allow end-to-end training. Further, we apply the U-net after having performed a temporal FFT, similar to Ref. [18].

In Ref. [50], where a non-Cartesian multi-coil dynamic acquisition is considered, the measured k-space data are first interpolated onto a Cartesian grid. After this interpolation, a simple FFT can be used as forward operator and thus facilitates the construction of an iterative network. However, because the k-space data interpolation is decoupled from the network, the network cannot learn to compensate for the interpolation errors. Thus, in order to really utilize the measured k-space data, applying the actual encoding operator (which involves the gridding-step) which is associated with the considered image reconstruction problem is unavoidable.

In contrast, in our approach, the gridding of the measured k-space is part of the network architecture. This important difference is the main motivation of the work since due to the

computational difficulties linked with the integration of the nonuniform FFT in the network architecture, a computationally light CNN-block has to be used.

Although the method shares similar features to other works, this is — to the best our knowledge — the first work to combine several components to construct a network architecture which is trainable in an end-to-end manner for a dynamic nonuniform multi-coil MR image reconstruction problem by actually using the radially acquired data and not the one interpolated onto a Cartesian grid. In fact, we believe that the large scale of the considered problem is the reason for the lack of end-to-end trainable reconstruction networks for dynamic non-Cartesian data acquisition protocols using multiple receiver coils.

5. CONCLUSION

In this work, we have proposed a new end-to-end trainable data-consistent reconstruction network for accelerated 2D dynamic MR image reconstruction with nonuniformly sampled data using multiple receiver coils. Further, since end-to-end training is computationally expensive because of the forward and the adjoint operators are included in the network, we have proposed and investigated an efficient training strategy to circumvent this issue. In addition, we have compared our method to other well-established iterative reconstruction methods as well as several methods based on learned regularization. Our proposed method surpassed all methods using non-learned regularization methods and achieved competitive results compared to a dictionary learning-based method and a method based on CNN-based image-priors. Although the method was presented for 2D radial cine MRI, we expect the training strategy to be applicable to general image reconstruction problems with computationally expensive operators as well. Further, we expect the proposed CNN-block to be applicable to arbitrary reconstruction problems with a time component and temporal correlation.

CONFLICT OF INTEREST

The authors have no conflict to disclose.

DATA AVAILABILITY STATEMENT

The implementation of the encoding operators $\mathbf{A}_I$, $\mathbf{A}_I^{\mathrm{H}}$, as well as the proposed CNN-block u_Θ and the entire reconstruction network u_Θ^M are available on https://github.com/koflera/DynamicRadCineMRI/.

[a]Author to whom correspondence should be addressed. Electronic mail: andreas.kofler@ptb.de

REFERENCES

1. Puntmann VO, Valbuena S, Hinojar R, et al. Society for Cardiovascular Magnetic Resonance (SCMR) expert consensus for CMR imaging endpoints in clinical research: Part I-analytical validation and clinical qualification. *J Cardiovasc Magn Reson*. 2018;20:67.
2. Wang G, Ye JC, Mueller K, Fessler JA. Image reconstruction is a new frontier of machine learning. *IEEE Trans Med Imaging*. 2018;37:1289–1296.
3. Ongie G, Jalal A, Metzler CA, Baraniuk RG, Dimakis AG, Willett R. Deep learning techniques for inverse problems in imaging. *IEEE J Select Areas Inform Theory*. 2020;1:39–56
4. 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.
5. Sandino CM, Dixit N, Cheng JY, Vasanawala SS. Deep convolutional neural networks for accelerated dynamic magnetic resonance imaging, Proceedings of 31st Conference of Neural Information Processing Systems (NIPS), Medical Imaigng meets NIPS Workshop; 2017. Online. Available at http://www.doc.ic.ac.uk/bglocker/public/mednips2017/mednips_2017_paper_19.pdf).
6. Hauptmann A, Arridge S, Lucka F, Muthurangu V, Steeden JA. Real-time cardiovascular MR with spatio-temporal artifact suppression using deep learning–proof of concept in congenital heart disease. *Magn Reson Med*. 2019;81:1143–1156.
7. Kofler A, Dewey M, Schaeffter T, Wald C, Kolbitsch C. Spatio-temporal deep learning-based undersampling artefact reduction for 2D radial cine MRI with limited training data. *IEEE Trans Med Imaging*. 2020;39:703–717.
8. El-Rewaidy H, Fahmy AS, Pashakhanloo F, et al. Multi-domain convolutional neural network (MD-CNN) for radial reconstruction of dynamic cardiac MRI. *Magn Reson Med*. 2020;85:1195–1208.
9. Hyun CM, Kim HP, Lee SM, Lee S, Seo JK. Deep learning for undersampled MRI reconstruction. *Phys Med Biol*. 2018;63:135007.
10. Kofler A, Haltmeier M, Schaeffter T, et al. Neural networks-based regularization for large-scale medical image reconstruction. *Phys Med Biol*. 2020;65:135003.
11. Antun V, Renna F, Poon C, Adcock B, Hansen AC. On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proc Nat Acad Sci*. 2020;117:30088–30095.
12. Maier AK, Syben C, Stimpel B, et al. Learning with known operators reduces maximum error bounds. *Nature Mach Int*. 2019;1:373–380.
13. 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.
14. 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.
15. Kobler E, Klatzer T, Hammernik K, Pock T. Variational networks: Connecting variational methods and deep learning. In: German conference on pattern recognition. Springer; 2017:281–293.
16. Qin C, Schlemper J, Caballero J, Price AN, Hajnal JV, Rueckert D. Convolutional recurrent neural networks for dynamic MR image reconstruction. *IEEE Trans Med Imaging*. 2018;38:280–290.
17. Aggarwal HK, Mani MP, Jacob M. Modl: Model-based deep learning architecture for inverse problems. *IEEE Trans Med Imaging*. 2018;38:394–405.
18. Qin C, Schlemper J, Duan J, et al. k-t NEXT: Dynamic MR Image Reconstruction Exploiting Spatio-Temporal Correlations. In: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer; 2019:505–513.
19. Gilton D, Ongie G, Willett R. Neumann networks for linear inverse problems in imaging. *IEEE Trans Comput Imaging*. 2019;6:328–343.
20. Kofler A, Haltmeier M, Kolbitsch C, Kachelrieß M, Dewey M. A U-nets cascade for sparse view computed tomography, In: International Workshop on Machine Learning for Medical Image Reconstruction, Springer; 2018:91–99.
21. Küstner T, Fuin N, Hammernik K., et al. CINENet: Deep learning-based 3D cardiac CINE MRI reconstruction with multi-coil complex-valued 4D spatio-temporal convolutions. *Sci Rep*. 2020;10:1–13.
22. Schlemper J, Salehi SSM, Kundu P, et al. Nonuniform Variational Network: Deep Learning for Accelerated Nonuniform MR Image Reconstruction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2019:57–64.
23. Malavé MO, Baron CA, Koundinyan SP, et al. Reconstruction of undersampled 3D non-Cartesian image-based navigators for coronary MRA

using an unrolled deep learning model. *Magn Reson Med*. 2020;84:800–812.
24. Duan J, Schlemper J, Qin C, et al. VS-Net: Variable splitting network for accelerated parallel MRI reconstruction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2019:713–722.
25. Lustig M, Donoho DL, Santos JM, Pauly JM. Compressed sensing MRI. *IEEE Signal Process Mag*. 2008;25:72–82.
26. Smith DS, Sengupta S, Smith SA, Welch EB, Trajectory optimized NUFFT: Faster non-Cartesian MRI reconstruction through prior knowledge and parallel architectures. *Magn Reson Med*. 2019;81:2064–2071.
27. 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.
28. Winkelmann S, Schaeffter T, Koehler T, Eggers H, Doessel O. An optimal radial profile order based on the Golden Ratio for time-resolved MRI. *IEEE Trans Med Imaging*. 2006;26:68–76.
29. Qu P, Zhong K, Zhang B, Wang J, Shen GX. Convergence behavior of iterative SENSE reconstruction with non-Cartesian trajectories. *Magn Reson Med*. 2005;54:1040–1045.
30. Le Gall D. MPEG: A video compression standard for multimedia applications. *Commun ACM*. 1991;34:46–58.
31. Tsao J, Boesiger P, Pruessmann KP. k-t BLAST and k-t SENSE: dynamic MRI with high frame rate exploiting spatiotemporal correlations. *Magn Reson Med*. 2003;50:1031–1042.
32. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation, In: International Conference on Medical image computing and computer-assisted intervention. Springer; 2015:234–241.
33. Ong F, Uecker M, Lustig M. Accelerating non-Cartesian MRI reconstruction convergence using k-space preconditioning. *IEEE Trans Med Imaging*. 2019;39:1646–1654.
34. Hestenes MR, Stiefel E, et al. Methods of conjugate gradients for solving linear systems. *J Res Nat Bureau Stand*. 1952;49:409–436.
35. Feng L, Srichai MB, Lim RP, et al. Highly accelerated real-time cardiac cine MRI using k-t SPARSE-SENSE. *Magn Reson Imaging*. 2012.
36. Malik WQ, Khan HA, Edwards DJ, Stevens CJ. A gridding algorithm for efficient density compensation of arbitrarily sampled Fourier-domain data. In: IEEE/Sarnoff Symposium on Advances in Wired and Wireless Communication, 2005, IEEE; 2005:125–128.
37. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: From error visibility to structural similarity. *IEEE Trans Image Process*. 2004;13:600–612.
38. Wang Z, Simoncelli EP, Bovik AC. Multiscale structural similarity for image quality assessment. In: The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003; IEEE; 2003;2:1398–1402.
39. Wang Z, Bovik AC. A universal image quality index. *IEEE Signal Process Lett*. 2002;9:81–84.
40. Sheikh HR, Bovik AC. Image information and visual quality. *IEEE Trans Image Process*. 2006;15:430–444.
41. Reisenhofer R, Bosse S, Kutyniok G, Wiegand T. A Haar wavelet-based perceptual similarity index for image quality assessment. *Signal Process*. 2018;61:33–43.
42. MEA Muckley, Torch KB-NUFFT. https://github.com/mmuckley/torchkbnufft, 2019.
43. Muckley MJ, Stern R, Murrell T, Knoll F, TorchKbNufft: A High-level, hardware-agnostic non-uniform fast fourier transform. ISMRM Workshop on Data Sampling & Image Reconstruction. 2020.
44. Pruessmann KP, Weiger M, Börnert P, Boesiger P. Advances in sensitivity encoding with arbitrary k-space trajectories. *Magn Reson Med*. 2001;46:638–651.
45. 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.
46. Wang Y, Ying L. Compressed sensing dynamic cardiac cine MRI using learned spatiotemporal dictionary. *IEEE Trans Biomed Eng*. 2014;61:1109–1120.
47. Caballero J, Price AN, Rueckert D, Hajnal JV. Dictionary learning and time sparsity for dynamic MR data reconstruction. *IEEE Trans Med Imaging*. 2014;33:979–994.
48. Pali MC, Schaeffter T, Kolbitsch C, Kofler A. Adaptive sparsity level and dictionary size estimation for image reconstruction in accelerated 2D radial cine MRI. *Med Phys*. 2020;48:178–192.
49. Schwab J, Antholzer S, Haltmeier M. Deep null space learning for inverse problems: Convergence analysis and rates. *Inverse Prob*. 2019;35:025008.
50. Biswas S, Aggarwal HK, Jacob M. Dynamic MRI using model-based deep learning and SToRM priors: MoDL-SToRM. *Magn Reson Med*. 2019;82:485–494.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Data S1. The supplementary material contains the results obtained by variying the hyper-parameters M and n_{CG} for our proposed method as well as for the method using the 3D U-Net. Further, a comparison of our proposed method with two different configurations of hyper-parameters is reported.