Assessment of data consistency through cascades of independently recurrent inference machines for fast and robust accelerated MRI reconstruction.

Karkalousos, D.; Noteboom, S.; Hulst, H.E.; Vos, F.M.; Caan, M.

Physics in Medicine & Biology

PAPER • **OPEN ACCESS**

Assessment of data consistency through cascades of independently recurrent inference machines for fast and robust accelerated MRI reconstruction

To cite this article: D Karkalousos *et al* 2022 *Phys. Med. Biol.* **67** 124001

IOP Publishing Phys. Med. Biol. 67 (2022) 124001 https://doi.org/10.1088/1361-6560/ac6cc2

Physics in Medicine & Biology

IPEM Institute of Physics and Engineering in Medicine

OPEN ACCESS

RECEIVED
30 November 2021

REVISED
12 April 2022

ACCEPTED FOR PUBLICATION
4 May 2022

PUBLISHED
8 June 2022

PAPER

Assessment of data consistency through cascades of independently recurrent inference machines for fast and robust accelerated MRI reconstruction

D Karkalousos[1], **S Noteboom**[2], **H E Hulst**[2,3], **F M Vos**[4] and **M W A Caan**[1]

[1] Department of Biomedical Engineering & Physics, Amsterdam UMC, University of Amsterdam, Amsterdam, The Netherlands
[2] Department of Anatomy & Neurosciences, MS Center Amsterdam, Amsterdam Neuroscience, Amsterdam UMC, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
[3] Department of Medical, Health and Neuropsychology, Leiden University, Leiden, The Netherlands
[4] Department of Imaging Physics, Delft University of Technology, Delft, The Netherlands

E-mail: d.karkalousos@amsterdamumc.nl

Keywords: MRI, image reconstruction, machine learning, data consistency, compressed sensing

Supplementary material for this article is available online

Abstract

Objective. Machine Learning methods can learn how to reconstruct magnetic resonance images (MRI) and thereby accelerate acquisition, which is of paramount importance to the clinical workflow. Physics-informed networks incorporate the forward model of accelerated MRI reconstruction in the learning process. With increasing network complexity, robustness is not ensured when reconstructing data unseen during training. We aim to embed data consistency (DC) in deep networks while balancing the degree of network complexity. While doing so, we will assess whether either explicit or implicit enforcement of DC in varying network architectures is preferred to optimize performance. *Approach.* We propose a scheme called Cascades of Independently Recurrent Inference Machines (CIRIM) to assess DC through unrolled optimization. Herein we assess DC both implicitly by gradient descent and explicitly by a designed term. Extensive comparison of the CIRIM to compressed sensing as well as other Machine Learning methods is performed: the End-to-End Variational Network (E2EVN), CascadeNet, KIKINet, LPDNet, RIM, IRIM, and UNet. Models were trained and evaluated on T_1-weighted and FLAIR contrast brain data, and T_2-weighted knee data. Both 1D and 2D undersampling patterns were evaluated. Robustness was tested by reconstructing 7.5× prospectively undersampled 3D FLAIR MRI data of multiple sclerosis (MS) patients with white matter lesions. *Main results.* The CIRIM performed best when implicitly enforcing DC, while the E2EVN required an explicit DC formulation. Through its cascades, the CIRIM was able to score higher on structural similarity and PSNR compared to other methods, in particular under heterogeneous imaging conditions. In reconstructing MS patient data, prospectively acquired with a sampling pattern unseen during model training, the CIRIM maintained lesion contrast while efficiently denoising the images. *Significance.* The CIRIM showed highly promising generalization capabilities maintaining a very fair trade-off between reconstructed image quality and fast reconstruction times, which is crucial in the clinical workflow.

1. Introduction

Magnetic resonance imaging (MRI) non-invasively images the anatomy of the human body. It is important to note that data are acquired in the frequency domain, known as k-space. Conventionally, the measured signals need to adhere to the Nyquist-criterion to allow for inverse Fourier transforming them to the image domain without aliasing. Due to hardware limitations and physical constraints, however, sampling the full k-space leads

to long scanning times. Almost 25 years ago, parallel-imaging (PI) (Sodickson and Manning 1997) was introduced to reduce acquisition times, overcoming hardware and software limitations by applying multiple receiver coil arrays. Each coil has a distinct sensitivity profile which can be exploited in reconstructing undersampled data. With sensitivity encoding (SENSE) the multicoil data are transformed to the image domain through the inverse Fourier Transform, after which a reconstruction algorithm dealiases the images based on the coil sensitivity maps (Pruessmann *et al* 1999). The combination of PI with compressed sensing (CS) (Lustig *et al* 2007, 2008) is now standardly applied in clinical settings, allowing for high acceleration factors by utilizing the constrained reconstruction through a sparsifying transform.

Machine Learning (ML) methods can learn how to reconstruct images by training them on acquired data for which a reference reconstruction is available. As such the reconstruction times can be reduced, which is of paramount importance to the clinical workflow. The UNet (Ronneberger *et al* 2015) may be the most popular network in the field and the base for numerous other methods, as elaborated upon below. Its unique architecture, with the down- and up-sampling operators and the large number of features on the output, has made it a cornerstone approach in image reconstruction today. Although such a network architecture can perform well, its performance is still limited due to operating only in image space without any MR physics knowledge incorporated.

Physics-informed networks were therefore introduced, incorporating the forward model of accelerated MRI reconstruction in the learning process. The variational network (VN) (Hammernik *et al* 2018) and the recurrent inference machines (RIM) (Putzky and Welling 2017, Lønning *et al* 2019, Putzky *et al* 2019) proposed to solve the inverse problem of accelerated MRI reconstruction through a Bayesian estimation. Alternatively, scan-specific techniques were used to restore missing k-space from fully-sampled autocalibration data (Akçakaya *et al* 2019, Kim *et al* 2019, Arefeen *et al* 2022). Furthermore, dual-domain networks were proposed to leverage the k-space information and perform corrections both in the frequency domain and the image domain. The Learned Primal-Dual reconstruction technique (LPDNet) (Adler and Öktem 2018) replaced the proximal operators in the Primal-Dual Hybrid Gradient algorithm (Chambolle and Pock 2011) with learned operators, yielding a learning scheme combined with model-based reconstruction. The KIKI-net (Eo *et al* 2018) introduced a sequence of convolutional neural networks (CNN) performed in k-space (K) and image space (I). Later, concatenations of UNets were applied to replace the sequence of CNNs in the KIKI-net (Souza *et al* 2020). Finally, the Model-Based Deep Learning technique (Aggarwal *et al* 2019) proposed a learned model-based reconstruction scheme involving a data consistency term.

With increasing network complexity, however, robustness is not ensured when reconstructing data unseen during training. This especially concerns clinical data with pathology for which fully sampled reference data cannot be obtained. This was understood in recent MRI reconstruction challenges (Beauferris *et al* 2020, Knoll *et al* 2020, Muckley *et al* 2021), in which deep end-to-end schemes, such as the End-to-End Variational Network (E2EVN) (Sriram *et al* 2020a), the XPDNet (Ramzi *et al* 2020a), and the Joint-ICNet (Jun *et al* 2021) allowed for higher image quality at increased acceleration factors but not necessarily for generalization to out-of-distribution data containing pathologies. Recurrent neural networks (RNNs), i.e. the RIM and the pyramid convolutional RNN (Chen *et al* 2019), appeared to generalize well on out-of-distribution data due to their nature of maintaining a notion of memory (Pascanu *et al* 2013). However, they scored lower on the trained data compared to the previously mentioned networks, possibly due to a limited number of iterations required to avoid gradient instabilities. Such methods would potentially benefit of increased network complexity as can be achieved using a number of cascades of networks (Schlemper *et al* 2018, Qin *et al* 2019, Souza *et al* 2019). The cascades can be considered as stacked networks targeting to resolve aliasing artifacts and to enhance denoising by iteratively evaluating the reconstruction, but without sharing parameters through backpropagation. Unfortunately, a solution may no longer be consistent with the acquired data with increasing network complexity. This raises a need for embedding data consistency in deep networks while balancing the degree of network complexity.

Data consistency (DC) can be embedded into the learning scheme in several ways, such as through gradient descent (Hammernik *et al* 2018, Lønning *et al* 2019, Schlemper *et al* 2019, Sriram *et al* 2020a), an iterative energy minimization process, namely variable-splitting (Duan *et al* 2019), generative adversarial networks (Yang *et al* 2018, Cole *et al* 2020, Dar *et al* 2020a, Sim *et al* 2020), adversarial transformers (Korkmaz *et al*), complex-valued networks (Wang *et al* 2020, Cole *et al* 2021, Xiao *et al* 2022), transfer learning (Dar *et al* 2020b), manifold approximation (Zhu *et al* 2018), or through sparsity (Yang *et al* 2017, Quan *et al* 2018, Sriram *et al* 2020b, Zhang *et al* 2020, Pezzotti *et al* 2020). Recent work evaluated enforcing DC in three ways, by gradient descent, by proximal mapping, and by variable-splitting (Hammernik *et al* 2021). It was shown that the training set could be reduced in size by doing so. The best results were obtained when train and test domains were aligned. However, it remains unknown whether either explicit or implicit enforcement of DC in varying network architectures is the best approach to optimize performance.

This work proposes a scheme called Cascades of Independently Recurrent Inference Machines (CIRIM). The CIRIM comprise RIM blocks sequentially connected through cascades and the efficient Independently Recurrent Neural Network (IndRNN) (Li *et al* 2018) as recurrent unit. The cascades allow us to train a deep but balanced RNN for improved de-aliasing and denoising, while maintaining stable gradient calculations. The enforcement of DC in an implicit or explicit manner will be assessed by comparison to the E2EVN. The networks are further compared to the CascadeNet (Schlemper *et al* 2018), the KIKINet (Eo *et al* 2018), the LPDNet (Adler and Öktem 2018), the RIM (Lønning *et al* 2019), the RIM built with the IndRNN, the UNet (Ronneberger *et al* 2015), and conventional Compressed Sensing reconstruction (Lustig *et al* 2008). The performance is evaluated on multi-modal MRI datasets applying different undersampling strategies. As a clinical application, we focused on reconstructing (out-of-training distribution) FLAIR data of multiple sclerosis patients. Finally, reconstruction times are also assessed as a critical aspect of improving clinical workflow.

2. Methods

In this section, first in 2.1, the MRI acquisition process is introduced. In 2.2, the background on solving the inverse problem of accelerated MRI reconstruction through a Bayesian approach is set. In 2.2.1 and 2.2.2, unrolled optimization by gradient descent is reviewed via the Recurrent Inference Machines (RIM) and the End-to-End Variational Network (E2EVN). The Cascades of Independently Recurrent Inference Machines (CIRIM) is then proposed in 2.2.3, to expand further de-aliasing capabilities of a deep trainable RNN. Furthermore, assessment of data consistency (DC) is performed in 2.2.2 and 2.2.3 to evaluate to what extent the performance of networks depends on the cascades or the DC formulation, or both. In 2.2.4, the loss function is explained with respect to the network's architecture. In 2.2.3, the experiments are described, i.e. the used datasets, the computed evaluation metrics, and the hyperparameters to be optimized.

2.1. Accelerated MRI acquisition

The process of accelerating the MRI acquisition can be described through a forward model. Let the true image be denoted by $\boldsymbol{x} \in \mathbb{C}^n$, with $n = n_x \times n_y$, and let $\boldsymbol{y} \in \mathbb{C}^m$, with $m \ll n$, be the set of the sparsely sampled data in k-space. The forward model describes how the measured data are obtained from an underlying reference image. For the $\boldsymbol{i}$th coil of $\boldsymbol{c}$ receiver coils, the forward model is formulated as:

$$y_i = A(x) + \sigma_i, \quad i = 1,\ldots,c, \tag{1}$$

in which $\boldsymbol{A}$: $\mathbb{C}^n \mapsto \mathbb{C}^{n \times n_c}$ denotes the linear forward operator modeling the sub-sampling process of multicoil data and $\boldsymbol{\sigma_i} \in \mathbb{C}^n$ denotes the measured noise for the $\boldsymbol{i}$th coil. $\boldsymbol{A}$ is given by

$$A = P \circ \mathcal{F} \circ \varepsilon. \tag{2}$$

Here, $\boldsymbol{P}$ is a sub-sampling mask selecting a fraction of samples to reduce scan time. $\boldsymbol{\mathcal{F}}$ denotes the Fourier transform, projecting the image onto k-space. $\boldsymbol{\varepsilon}$: $\mathbb{C}^n \times \mathbb{C}^{n \times n_c} \mapsto \mathbb{C}^{n \times n_c}$ denotes the expand operator, transforming $\boldsymbol{x}$ into $\boldsymbol{x_c}$ multicoil images and is given by

$$\varepsilon(x) = (S_0 \circ x,\ldots,S_c \circ x) = (x_0,\ldots,x_c), \tag{3}$$

where $\boldsymbol{S_i}$ are the coil sensitivity maps, a diagonal matrix representing the spatial sensitivities that scale every pixel of the reference image by a complex number

The adjoint backward operator of $\boldsymbol{A}$ in (2), projecting $\boldsymbol{y}$ onto image space, is given by

$$A^* = \rho \circ \mathcal{F}^{-1} \circ P^T, \tag{4}$$

where $\boldsymbol{\mathcal{F}^{-1}}$ denotes the inverse Fourier transform, and $\boldsymbol{\rho}$: $\mathbb{C}^{n \times n_c} \times \mathbb{C}^{n \times n_c} \mapsto \mathbb{C}^n$ denotes the reduce operator that serves for combining the multicoil images $\boldsymbol{x_c}$ into $\boldsymbol{x}$. $\boldsymbol{\rho}$ is given by

$$\rho(x_0,\ldots,x_c) = \sum_{i=1}^{c} S_i^H \circ x_i, \tag{5}$$

with $\boldsymbol{H}$ representing Hermitian complex conjugation.

2.2. The inverse problem of accelerated MRI reconstruction

The objective when solving the inverse problem of accelerated MRI reconstruction (figure 1) is to map the sparsely sampled k-space measurements to an unaliased, highly accurate image. The inverse transformation of restoring the true image from the set of the sparsely sampled measurements can be found through the *Maximum A Posteriori* (MAP) estimator, given by

$$x_{\text{MAP}} = argmax_x \{log\ p\ (y|x) + log\ p\ (x)\}, \tag{6}$$

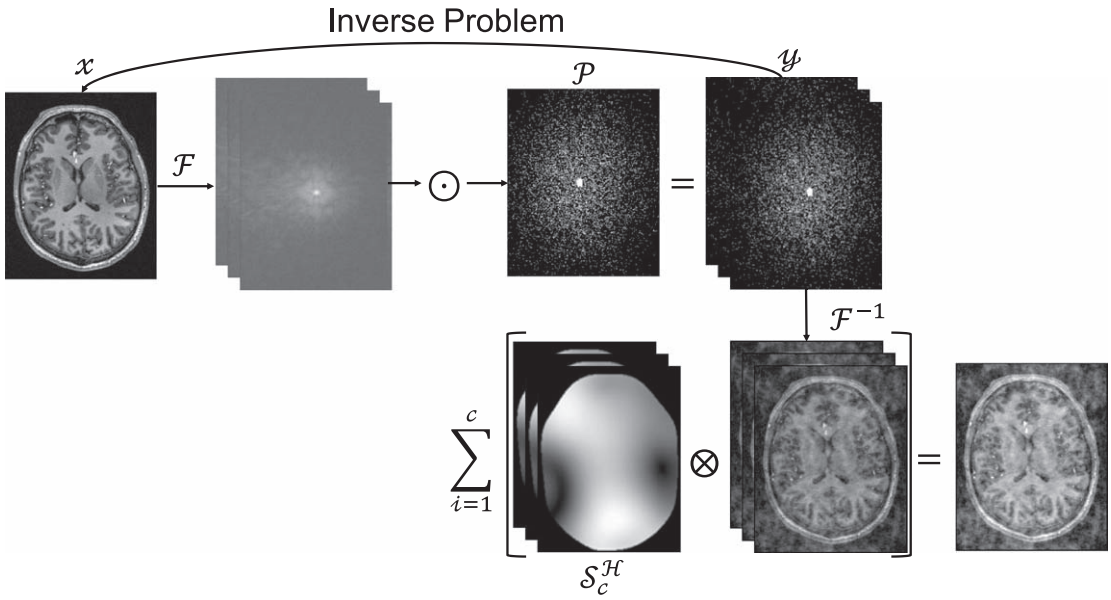

Figure 1. The objective in accelerated MRI reconstruction is to solve the inverse problem of recovering an unaliased image (x) from a set of sparsely sampled measurements (y). A forward model starts from the true image representation (x) (top-first), measured over multiple receiver coils (S) (bottom-first image). It is Fourier transformed to k-space (top-second) and sub-sampled using a mask (P) (top-third) to obtain sparsely sampled measurements (y) (top-fourth). Through the inverse Fourier transform (bottom-second) and after combining with coil sensitivity maps (bottom-first), an aliased image is obtained (bottom-third).

which is the maximization of the sum of the log-likelihood and the log-prior distribution of $\boldsymbol{y}$ and $\boldsymbol{x}$, respectively. The log-likelihood expresses the log probability that k-space data $\boldsymbol{y}$ are obtained given an image $\boldsymbol{x}$, yielding a data fidelity term derived from the posterior $p(y|x)$. The log-prior distribution regularizes the solution by representing an MR-image's most likely appearance.

Conventionally, equation (6) is reformulated as the following optimization problem

$$x_{MAP} = argmin_x \left\{ \sum_{i=1}^{c} d(y_i, A(x)) + \lambda R(x) \right\}, \tag{7}$$

where d ensures data consistency between the reconstruction and the measurements, reflecting the error distribution given by the log-likelihood distribution in equation (6). R is a regularizer weighted by λ, which constrains the solution space by incorporating prior knowledge over x.

Assuming Gaussian distributed data and ignoring the regularization term in equation (7), the negative log-likelihood is:

$$log\ p(y|x) = \frac{1}{\sigma^2} \sum_{i=1}^{c} \|A(x) - y_i\|_2^2. \tag{8}$$

2.2.1. Recurrent inference machines (RIM)

The RIM (Lønning *et al* 2019) were originally proposed as a general inverse problem solver. The RIM targets iterative optimization of a model with a complex-valued parametrization, requiring taking derivatives with respect to a complex variable. This can be achieved using the Wirtinger- or $\mathbb{CR}$-calculus (Amin *et al* 2011, Sarroff *et al* 2015, Zhang and Mandic 2016). Gradient descent is performed by us using the Wirtinger derivative, to yield a real-valued cost function of complex values. The unrolled scheme for generating updates is presented in figure 2.

Non-convex optimization can be performed based on the approach by Andrychowicz *et al* (2016). The update rules are learned by the optimizer $\boldsymbol{h}$, which has its own set of parameters ϕ. Formulating equation (6) accordingly, resulting updates are of the form

$$x_{\tau+1} = x_\tau + h_\phi(\nabla_{y|x_\tau}, x_\tau), \tag{9}$$

at iteration τ and for a (*a priori* set) total number of iterations $\boldsymbol{T}$.

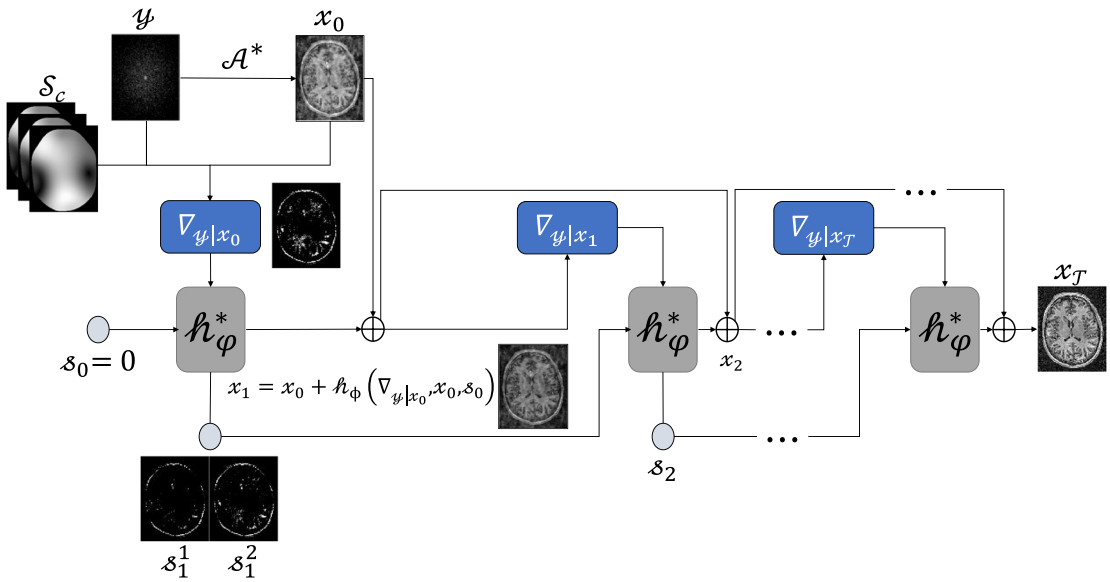

Figure 2. The recurrent inference machines unrolled over two iterations. The inputs to the model are the set of sparsely sampled measurements (y) (top, second image), the coil sensitivities maps (S_c) (top, first image), and the initial estimation (x_0) (top, third image) for the estimation of the log-likelihood gradient (llg) ($\nabla_{y|x_0}$). The llg is passed through a network to produce updates; the network maintains hidden states initialized by $s_0 = 0$, $s_1 = 0$. At each iteration (τ) the network updates itself and after total (T) iterations produces the final estimation (x_T) (rightmost).

The gradient of the log-likelihood function is given by

$$\nabla_{y|x} := \frac{1}{\sigma^2} A^*(A(x) - y). \tag{10}$$

The advantage of the RIM is the explicit modeling of the update rule $\boldsymbol{h}_\phi$ using a recurrent neural network (RNN). In addition to the gradient information, the model is aware of the position of the estimation in variable space equation (9).

By inserting equation (10) into (9), the update equations are obtained, given by

$$s_0 = 0, \qquad x_0 = A^*(y),$$

$$s_{\tau+1} = h^*_\phi(\nabla_{y|x_\tau}, x_\tau, s_\tau), \quad x_{\tau+1} = x_\tau + h_\phi(\nabla_{y|x_\tau}, x_\tau, s_{\tau+1}). \tag{11}$$

where $\boldsymbol{h}^*_\phi$ is the updated model for state variable $\boldsymbol{s}$. Equation (11) reflects that not the prior is explicitly evaluated, but instead its gradient when performing updates. The step size is learned implicitly in combination with the prior. Therefore, $\boldsymbol{h}_\phi$ also acts as regularizer $\boldsymbol{R}$ in equation (7). Observe that the RIM contains latent (hidden) states, representing the recurrent aspects of the network.

2.2.2. End-to-end variational network (E2EVN)

The variational network (VN) (Hammernik *et al* 2018) introduces a mapping to real-valued numbers, going from mapping $\mathbb{C}^n \mapsto \mathbb{C}^m$ to mapping $\mathbb{R}^{2n} \mapsto \mathbb{R}^{2m}$. $\boldsymbol{x}$ can be computed by least-squares minimization in equation (8). As originally proposed in Chen *et al* (2015) and adapted by the VN, the idea is to perform gradient descent through the iterative Landweber algorithm. By defining a regularizer $\boldsymbol{R}$, equation (7) can be formulated as a trainable gradient scheme with time-varying parameters.

The End-to-End Variational Network (E2EVN) (Sriram *et al* 2020a) uses a UNet as regularizer (R_{UNet}), whose parameters are learned from the data. Unrolled optimization of the regularized problem in equation (7) is performed through cascades, given by

$$\hat{x}_{k+1} = \lambda R_{UNet\,k}(A^*(y_k)), \tag{12}$$

for cascade k, with $1 \leqslant k \leqslant K$ for a total number of K cascades. Next, an explicitly formulated data consistency step applies k-space corrections. This step is given by

$$y_{k+1} = y_k - d(y_k - y) - A(\hat{x}_{k+1}), \tag{13}$$

where $d(y_k - y)$ is a soft DC term, with a weighting factor d. The optimization is initialized with the (sparsely sampled) measurement data, $y_{k=1} = y$. The eventual image is obtained via the adjoint operator $x_K = A^*(y_K)$.

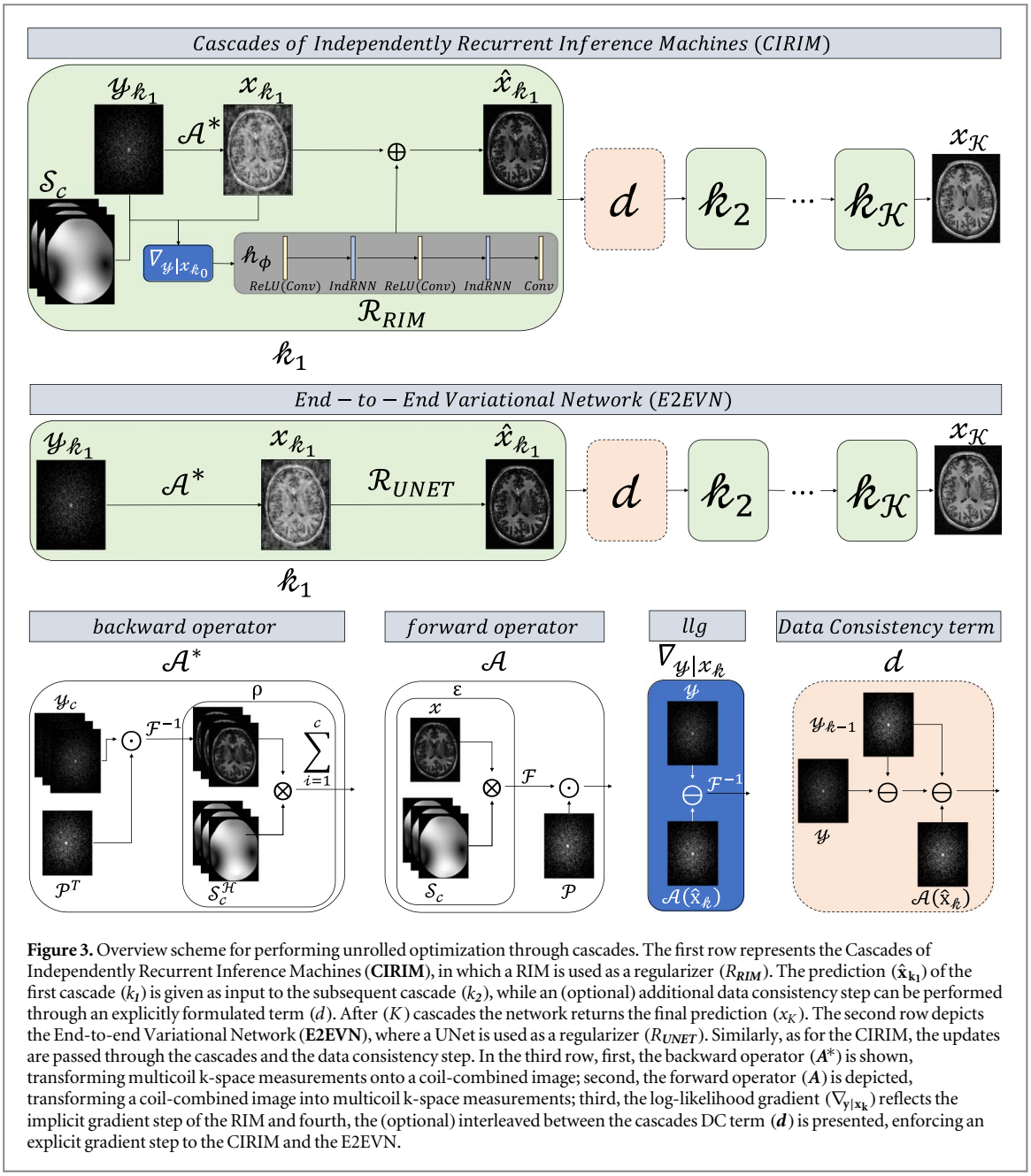

Figure 3. Overview scheme for performing unrolled optimization through cascades. The first row represents the Cascades of Independently Recurrent Inference Machines (**CIRIM**), in which a RIM is used as a regularizer (R_{RIM}). The prediction ($\hat{\mathbf{x}}_{\mathbf{k_1}}$) of the first cascade ($k_1$) is given as input to the subsequent cascade (k_2), while an (optional) additional data consistency step can be performed through an explicitly formulated term (d). After (K) cascades the network returns the final prediction (x_K). The second row depicts the End-to-end Variational Network (**E2EVN**), where a UNet is used as a regularizer (R_{UNET}). Similarly, as for the CIRIM, the updates are passed through the cascades and the data consistency step. In the third row, first, the backward operator ($\boldsymbol{A}^*$) is shown, transforming multicoil k-space measurements onto a coil-combined image; second, the forward operator ($\boldsymbol{A}$) is depicted, transforming a coil-combined image into multicoil k-space measurements; third, the log-likelihood gradient ($\nabla_{\mathbf{y}|\mathbf{x_k}}$) reflects the implicit gradient step of the RIM and fourth, the (optional) interleaved between the cascades DC term ($\boldsymbol{d}$) is presented, enforcing an explicit gradient step to the CIRIM and the E2EVN.

In this paper, we test omitting the DC step, in equation (13), and evaluate if the network's performance is more dependent on the cascades or the gradient step. In that case, updates are given by equation (12). Note that the cascades effectively yield sequentially connected VN blocks, targeting de-aliasing (figure 3).

The complex-valued image to complex-valued image mapping is performed in image space by concatenating the real and imaginary parts along the coil dimension. After the regularizer's update in equation (12), the image is reshaped to have the real and imaginary parts stacked to a complex (last) dimension.

2.2.3. Cascades of independently recurrent inference machines (CIRIM)

We now propose CIRIM, consisting of sequentially connected RIM blocks (figure 3). The cascades allow building a deep RNN without vanishing or exploding gradients issues and further evaluate equation (9) through K cascades. As such the RIM acts as regularizer (R_{RIM}), while the updates to the CIRIM are given by

$$\hat{x}_{k+1} = x_k + \lambda R_{RIM\,k}(x_k), \tag{14}$$

for cascade k, with $1 \leqslant k \leqslant K$.

In previous work (Putzky and Welling 2017, Lønning *et al* 2019), the gated recurrent unit (GRU) (Cho *et al* 2014) was used as recurrent unit for the RIM. A key novelty of our approach is to include the Independently Recurrent Neural Networks (IndRNN) (Li *et al* 2018) as a more efficient unit for balancing the network's complexity while increasing the number of trainable parameters through the cascades.

Through the cascades the network's size has increased, but it is unclear whether either implicitly evaluating data consistency through the log-likelihood gradient in equation (10) is adequate, or an additional learned gradient step is needed to constrain the solution space further. In a similar manner as in equation (13), we assess enforcing DC explicitly and interleaved between the cascades. By doing so, we aim to understand to what extent the network's performance and de-aliasing capabilities depend on the cascades or the formulation of the DC.

Then, the updated prediction of the model is given by

$$\hat{x}_{k+1} = A^*(y_k - d(y_k - y) - A(\hat{x}_{k+1})), \tag{15}$$

with $x_K = A^*(y_K)$. If this DC step is omitted, updates to the CIRIM are given by equation (14). Implementation notation for the recurrent units can be found in the appendix.

2.2.4. Loss function

For calculating the loss, we compare magnitude images derived from the complex-valued estimations $\hat{x}$ against the fully sampled reference x. As a loss function, we choose the ℓ_1-norm. The ℓ_1-norm represents the sum of the absolute difference, given by

$$\mathrm{L}^{\ell_1}(\mathrm{x}) = |\,\hat{\mathrm{x}} - \mathrm{x}\,|. \tag{16}$$

For the E2EVN and other trained models, the loss is given by equation (16). For the CIRIM, the loss is weighted depending on the number of iterations and averaged over the K cascades, to emphasize the predictions of the later iterations. The loss is then formulated as

$$L^{\ell_1}(\hat{x}) = \frac{\sum_{i=1}^{c}\left(\frac{1}{qT}\sum_{\tau=1}^{T} w_\tau\,|\,\hat{x}_\tau - x\,|\right)}{K}, \tag{17}$$

where q is the total number of pixels of the image and w_τ is a weighting vector of length T prioritizing the loss at later time-steps. The weights are calculated by setting $w_\tau = 10^{-\frac{T-\tau}{T-1}}$.

2.3. Experiments

For our experiments, we used multiple datasets as described in 2.3.1. Scanning parameters of these datasets are given in table 1.

Our experiments focused on assessment of the following aspects:

A. Training and validation in fully sampled and retrospectively undersampled data. The undersampling strategy is described in 2.3.2.

B. Independent evaluation in prospectively undersampled data of Multiple Sclerosis patients containing white matter lesions.

We trained and compared the CIRIM and the E2VN to the LPDNet, the KIKINet, and the CascadeNet (Adler and Öktem 2018, Eo *et al* 2018, Schlemper *et al* 2018), the hyperparameters of which are described in 2.3.3. For comparing the performance of the methods regarding assessment (A) we chose the structural similarity index measure (SSIM) (Wang *et al* 2004) and the peak signal-to-noise ratio (PSNR). For assessment (B), we calculated the contrast resolution (CR), the noise in the white matter (WMN), the noise in the background (BGN), and a resulted weighted average (WA). The metrics are described in 2.3.4.

2.3.1. Datasets

For assessment (A), three fully sampled raw complex-valued multi-coil datasets were obtained. The first dataset was acquired in-house. To this end, eleven healthy subjects were included, from whom written informed consent (under an institutionally approved protocol) was obtained beforehand. The ethics board of Amsterdam UMC declared that this study was exempt from IRB approval. All eleven subjects were scanned by performing 3D T_1-weighted brain imaging on a 3.0 T Philips Ingenia scanner (Philips Healthcare, Best, The Netherlands) in Amsterdam UMC. The data were visually checked to ascertain that they were not affected by motion artifacts. After scanning, raw data were exported and stored for offline reconstruction experiments. The training set was composed of ten subjects (approximately 3000 slices) and the validation set of one subject (approximately 300 slices).

The second dataset consisted of 451 2D multislice FLAIR scans, publicly available through the fastMRI brains dataset (Muckley *et al* 2020). The training set consisted of 344 scans (approximately 5000 slices) and the validation set of 107 scans (approximately 500 slices). The number of coils varied from 2 to 24. The data were

Table 1. Scan parameters of each dataset used for different experiments. Target anatomy, contrast, scan, and field strength are given, with resolution (res), field-of-view (FOV), time in minutes (with acceleration factor), number of coils (ncoils) and other scan parameters.

Scan/sequence	Field strength	Res (mm)	FOV (mm)	Time (acc)	Ncoils	Parameters
			Training, Validation			
T_1-Brain/T_1 3D MPRAGE	3 T	1.0 × 1.0 × 1.0	256 × 256 × 240	10.8 (1x)	32	FA 9°, TFE-factor 150, TI = 900 ms
T_2-Knee/T_2 TSE	3 T	0.5 × 0.5 × 0.6	160 × 160 × 154	15.3 (1x)	8	FA 90°, TR = 1550 ms, TE = 25 ms
FLAIR-Brain/2D FLAIR	1.5 T/3 T	0.7 × 0.7 × 5	220 × 220	- (1x)	2–24	FA 150°, TR = 9000 ms, TE = 78–126 ms
			Pathology study			
MS FLAIR-Brain/3D FLAIR	3 T	1.0 × 1.0 × 1.1	224 × 224 × 190	4.5 (7.5x)	32	TR = 4800 ms, TE = 350 ms, TI = 16500 ms

cropped in the image domain to 320 for the readout direction by the size of the phase encoding direction (varied from 213 to 320). The cropped images were visually evaluated to not crop any tissue (only air).

The third dataset was composed of 3D knee scans of 20 subjects, available on a public repository (Epperson *et al* 2013). From these data, two subjects were discarded due to observed motion artifacts. The training set consisted of 17 subjects (approximately 12 000 slices) and the validation set of one subject (approximately 700 slices).

For all datasets, coil sensitivities were estimated using an autocalibration procedure called ecalib from the BART toolbox (Uecker *et al* 2015), which leverages the ESPIRiT algorithm (Uecker *et al* 2014). For training and validation, slices were randomly selected by setting a random seed to enable deterministic behavior for all methods and ensure reproducibility. Note that the validation set was only used to calculate the loss at the end of each epoch and not included into the training set. Finally, all volumes were normalized to the maximum magnitude.

For assessment (B), testing the methods' ability to reconstruct unseen pathology, a dataset of 3D FLAIR data of multiple sclerosis patients with known white matter lesions was obtained. Data were prospectively undersampled with a factor of 7.5x based on a Variable-Density Poisson disk distribution. Originally these data were acquired on a 3.0 T Philips Ingenia scanner (Philips Healthcare, Best, The Netherlands) in Amsterdam UMC, within the scope of a larger, ongoing study. The local ethics review board approved this study and patients provided informed consent prior to imaging. A fully-sampled reference scan was also acquired and used to estimate coil sensitivity maps using the caldir method of the BART toolbox (Uecker *et al* 2015). The data were visually checked after which all subjects with motion artifacts were discarded, ending up including 18 patients (approximately 4000 slices).

2.3.2. Undersampling

The masks for retrospective undersampling in assessment (A) were initially defined in 2D. As such the models trained on all modalities could also be used later for reconstructing high-resolution isotropic FLAIR data for assessment (B). The 3D datasets were first Fourier transformed along the frequency encoding axis and used as separate slices along the two-phase encoding axes. The 2D multislice FLAIR dataset was initially Fourier transformed along the frequency encoding axis and undersampled per slice in 2D, to train a model on an identical contrast as in assessment (B), while also having pathology present in the data.

All data were retrospectively undersampled in 2D by sampling k-space points from a Gaussian distribution with a full width at half maximum (FWHM) of 0.7, relative to the k-space dimensions. Hereby the sampling of low frequencies is prioritized whereas incoherent noise is created due to the random sampling. Note that in this way, we abide by the compressed sensing (CS) requirement of processing incoherently sampled data (Lustig *et al* 2007). For autocalibration purposes, data points near the k-space center were fully sampled within an ellipse of which the half-axes were set to 2% of the fully sampled region. Acceleration factors of $4\times$, $6\times$, $8\times$, and $10\times$ were used by randomly generating a sampling mask (P) with according sampling density (both during training and validation).

To abide to the underlying sampling protocol, and to test the model's ability to reconstruct undersampled data in 1D, we performed an additional experiment with retrospective undersampling in just one dimension. Equidistant k-space points were sampled in the phase encoding direction (Uecker *et al* 2014). The acceleration factor was set to four, while the central region was densely sampled retaining eight percent of the fully-sampled k-space.

2.3.3. Hyperparameters

For the CIRIM models, hyperparameters were selected as follows. The number of cascades K was set to 5, the number of channels to 64 for the recurrent and convolutional layers, and the number of iterations $\mathscr{T}$ to 8. The hyperparameter search for finding the optimal number of cascades is shown in the Supplementary Material (available online at stacks.iop.org/PMB/67/124001/mmedia). The kernel size of the first convolutional layer was set to 5×5 and to 3×3 for the second and third layers. The optimization of these hyperparameters is described elsewhere (Lønning *et al* 2019). Next, we trained models on the T_1-Brain dataset, the T_2-Knee dataset, and the FLAIR-Brain dataset to realize the DC step from equation (15).

For the E2EVN models, we chose 8 cascades, 4 pooling layers, 18 channels for the convolutional layers, and included the DC step from equation (13). The hyperparameter search for finding the optimal number of cascades, pooling layers, and number of channels, is again shown in the supplementary material. Then, for further optimization, we experimented with training models on the T_1-Brain dataset, the T_2-Knee dataset, and the FLAIR-Brain dataset while omitting the DC step. The inputs to the UNet regularizer were padded for making the inputs square, setting the padding size to 11, and the outputs were unpadded for restoring the original input size.

For the baseline UNet, the number of input and output channels was set to 2. The number of channels for the convolutional layers was set to 64, and we chose 2 pooling layers. Similar to the E2EVN, the padding size was set to 11, while no dropout was applied. The selected hyperparameters for the UNet were motivated by the configuration in Zbontar *et al* (2018).

For the LPDNet, the KIKINet, and the CascadeNet, the choice of the hyperparameters was motivated from the baseline proposed models. For the LPDNet we used the same network architecture for both the primal and the dual part, being a UNet with 16 channels, 2 pooling layers, and padding size of 11, while no dropout was applied. The number of the primals, the duals, and the number of unrolled iterations was set to 5. Similarly, for the KIKINet, we used the UNet architecture for the k-space and the image space networks. The number of channels was set to 64, the number of pooling layers to 2, and the padding size to 11, without applying any dropout. Finally, for the CascadeNet the number of cascades set to 10, using a sequence of CNNs with 64 channels and depth size of 5, without applying batch normalization.

For all models, we applied the ADAM optimizer (Kingma and Ba 2015), setting the learning rate to 1e-3, except for the CascadeNet where the learning rate was set to 1e-5. The batch size was set to 1, allowing training on various input sizes. The data type was set to complex64 for complex-valued data and to float16 for real-valued data. For training models with 2D undersampling, the loss function for the CIRIM is given by equation (17) and for all models by equation (16). For training models with 1D undersampling, we used the SSIM as loss function, motivated by (Muckley *et al* 2021), as a better option for resolving artifacts introduced by equidistant undersampling.

CS reconstructions were performed using the BART toolbox (Uecker *et al* 2015). Here we used parallel-imaging compressed sensing (PICS) with a ℓ_1-wavelet sparsity transform. The regularization parameter was set to $\alpha = 0.005$, which was heuristically determined as a trade-off between aliasing noise and blurring. The maximum number of iterations was set to 60. We tested the reconstruction times of CS on the GPU (turning the -g flag on).

All experiments were performed on an Nvidia Tesla V100 with 32GB of memory. The code was implemented in PyTorch 1.9 (Paszke *et al* 2019) and PyTorch-Lighting 1.5.5 (Falcon and Team 2019), on top of novel frameworks (Kuchaiev *et al* 2019, Yiasemis *et al* 2022), and can be found at https://github.com/wdika/mridc.

2.3.4. Evaluation metrics

For quantitative evaluation of the fully-sampled measurements, we compared normalized magnitude images derived from the complex-valued estimations x_T against the reference x and calculated SSIM and PSNR metrics.

For evaluating robustness on the 3D FLAIR MS data, we computed the contrast resolution (CR), the noise in the white matter (WMN), the noise in the background (BGN), and a resulted weighted average (WA) of those three metrics.

Since the data are not fully-sampled, the CR is an efficient metric to evaluate the signal level between the white matter and the lesions. To compute CR, lesion segmentations were performed using a pretrained multi-view convolutional neural network (MV-CNN). The MV-CNN was previously trained on combined Fast Imaging Employing Steady-state Acquisition (FIESTA), T_2-weighted and contrast-enhanced T_1-weighted data, for eye and tumor segmentation of retinoblastoma patients (Strijbis *et al* 2021). For the segmentation of the white matter, the statistical parametric mapping (SPM) toolbox was used (Penny *et al* 2007). The mean lesion intensity was compared to that of presumed homogeneous surrounding white matter. To that end, the lesion masks were dilated by four voxels and intersected with the whole brain white matter mask. The CR is then defined as the difference between the lesion signal and the signal in the surrounding white matter, divided by the summation of them, given by

$$\mathrm{CR} = \frac{s_{\mathrm{lesion}} - s_{\mathrm{WMSurroundingLesion}}}{s_{\mathrm{lesion}} + s_{\mathrm{WMSurroundingLesion}}}. \tag{18}$$

The WMN is defined as the mode of the gradient magnitude image x, given by

$$\mathrm{WMN} = \mathrm{mode}(\nabla |x/\overline{x_{\mathrm{WM}}}|), \tag{19}$$

where $\overline{x_{\mathrm{WM}}}$ is the mean WM intensity. The background noise (BGN) is computed as the 99-percentile value in the background region, being the complement of a tissue mask.

A weighted average (WA) was eventually defined as the combination of the CR, the WMN, and the BGN after scaling them to maximum value.

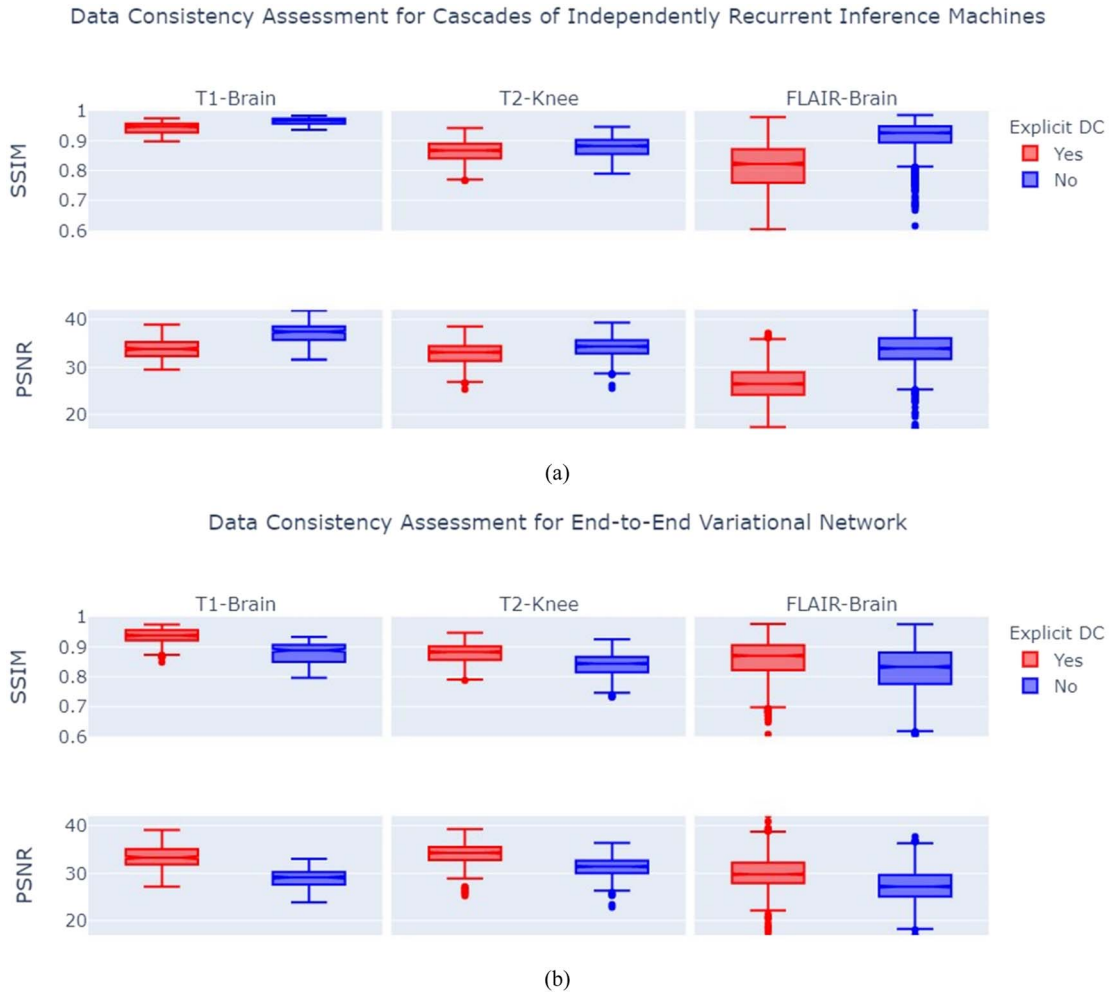

Figure 4. Data consistency (DC) assessment for (a) Cascades of Independently Recurrent Inference Machines and (b) End-to-End Variational Network. DC is enforced both explicitly (red) and implicitly (blue). The first row represents SSIM scores and the second row PSNR scores. Performance is reported for models trained on the T_1-Brain dataset (first column), the T_2-Knee dataset (second column), and the FLAIR-Brain dataset (third column).

Finally, for every scan, the signal-to-noise ratio (SNR) was calculated as follows,

$$\mathrm{SNR} = \frac{\overline{t|x|}}{\widetilde{|y|}}, \tag{20}$$

where $\overline{t|x|}$ is the mean value after thresholding the magnitude image x to discard the background, and $\widetilde{|y|}$ the median magnitude value within a square region in the periphery of k-space, which was assumed to be dominated by imaging noise. The threshold t was set using Otsu's method (Otsu 1979).

3. Results

Figure 4 shows SSIM and PSNR scores upon assessing DC explicitly and implicitly for the CIRIM (figure 4(a)) and the E2EVN (figure 4(b)). The models were trained on the T_1-Brain dataset, the T_2-Knee dataset, and the FLAIR-Brain dataset.

A qualitative evaluation of the CIRIM's and the E2EVN's performance on the trained datasets, accelerated with ten-times Gaussian 2D undersampling, is presented in figure 5. The CIRIM performed significantly better than the E2EVN on the T_1-Brain and the FLAIR-Brain dataset. On the FLAIR-Brain dataset, the E2EVN failed to accurately reconstruct the center of brain, as well as to resolve noise in the White Matter lesion. On the T_2-Knee dataset, the two models performed comparably in terms of SSIM, while the CIRIM showed a slight improvement in PSNR.

In figure 6, the CIRIM is compared to the RIM and the IRIM. SSIM and PSNR scores are reported for each model trained on the T_1-Brain dataset, the T_2-Knee dataset, and the FLAIR-Brain dataset. The IRIM performed slightly worse compared to the RIM, while the CIRIM performed best.

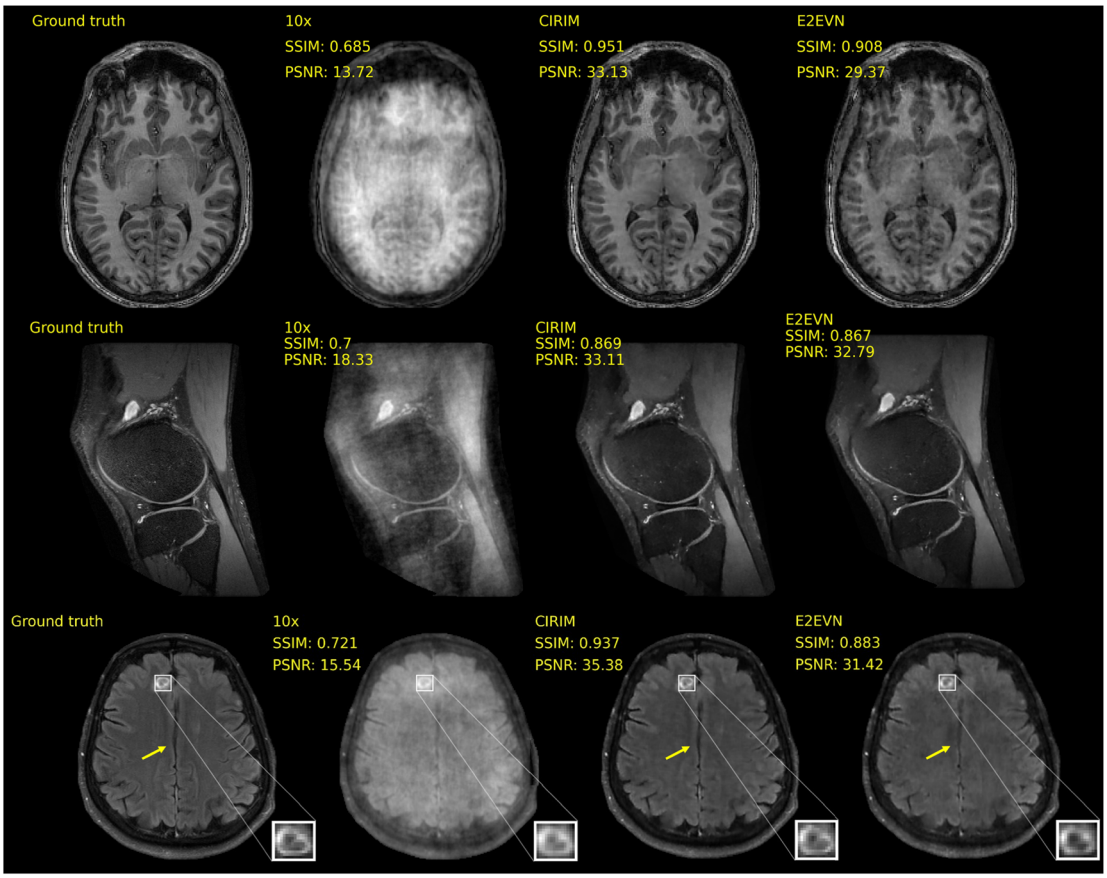

Figure 5. Comparison of the CIRIM (third column) to the E2EVN (fourth column) for reconstructing ten-times accelerated slices from the T_1-Brain dataset (first row, first and second image), the T_2-Knee dataset (second row, first and second image), and the FLAIR-Brain dataset (third row, first and second images). For the FLAIR-Brain dataset, the inset focuses on a reconstructed White Matter lesion; obtained through the fastMRI + annotations (Zhao *et al* 2022). The arrow points out to a region of interested.

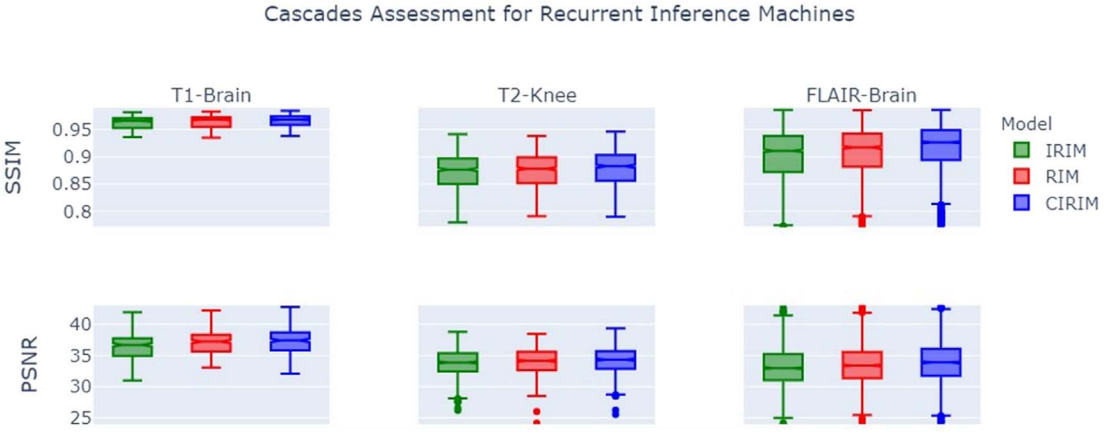

Figure 6. Comparison of the Cascades of Independently Recurrent Inference Machines (CIRIM) (blue color), to the recurrent inference machines (RIM) (red color), and the independently recurrent inference machines (IRIM) (green color). Performance is reported for SSIM (first row) and PSNR (second row), on the T_1-Brain dataset (first column), the T_2-Knee dataset (second column), and the FLAIR-Brain dataset (third column).

Table 2 collates overall performance of the methods on all training datasets (T_1-Brain, T_2-Knee, FLAIR-Brain). The methods were evaluated with ten-times accelerated data using Gaussian 2D masking, and four times accelerated equidistant 1D masking. For the FLAIR-Brain dataset we dropped the slices outside the head, containing no signal. The CIRIM performed best in all settings in terms of SSIM and PSNR, while only the E2EVN had comparable performance for the evaluation on the T_2-Knee dataset. Representative reconstructions

Table 2. SSIM and PSNR scores of all methods evaluated on the T_1-Brain dataset (third and fourth column), the T_2-Knee dataset (fifth and sixth column), and the FLAIR-Brain dataset (seventh to tenth column). For all datasets performance is reported for ten times acceleration using Gaussian 2D undersampling. For the FLAIR-Brain dataset performance is also reported for four times acceleration using equidistant 1D undersampling (ninth and tenth column). The first column reports the method's name. The second column reports the total number of trainable parameters for each model. Best performing models are highlighted in bold. Methods are sorted in alphabetical order.

		T_1-Brain Gaussian2D10x		T_2-Knee Gaussian2D10x		FLAIR-Brain Gaussian2D10x		FLAIR-Brain Equidistant1D4x	
Method	Params	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑
CascadeNet	1.1 M	0.922 ± 0.042	30.2 ± 1.5	0.859 ± 0.045	32.2 ± 2.6	0.872 ± 0.106	29.9 ± 5.0	0.913 ± 0.038	30.3 ± 4.6
CIRIM	264k	**0.966 ± 0.015**	**35.8 ± 0.5**	**0.877 ± 0.039**	**33.7 ± 2.3**	**0.906 ± 0.101**	**32.8 ± 6.2**	**0.942 ± 0.065**	**34.3 ± 3.2**
E2EVN	19.6 M	0.940 ± 0.023	31.8 ± 1.4	**0.877 ± 0.039**	33.5 ± 2.5	0.855 ± 0.108	29.1 ± 4.8	0.930 ± 0.062	31.3 ± 5.1
IRIM	53k	0.963 ± 0.017	35.3 ± 0.6	0.870 ± 0.041	33.3 ± 2.3	0.892 ± 0.107	32.0 ± 6.0	0.908 ± 0.093	32.0 ± 5.4
KIKINet	1.9 M	0.925 ± 0.040	31.1 ± 1.3	0.842 ± 0.045	32.1 ± 2.0	0.829 ± 0.113	28.4 ± 4.8	0.919 ± 0.065	30.5 ± 4.6
LPDNet	118k	0.960 ± 0.016	35.0 ± 0.4	0.873 ± 0.038	33.5 ± 2.0	0.858 ± 0.011	29.7 ± 4.7	0.938 ± 0.061	32.3 ± 5.3
PICS		0.866 ± 0.032	30.9 ± 0.7	0.729 ± 0.041	29.7 ± 4.3	0.816 ± 0.174	29.2 ± 7.6	0.876 ± 0.068	30.0 ± 4.2
RIM	94k	0.963 ± 0.017	35.3 ± 0.4	0.872 ± 0.040	33.5 ± 2.3	0.898 ± 0.103	32.3 ± 6.1	0.934 ± 0.069	33.4 ± 3.2
UNet	1.9 M	0.874 ± 0.049	26.6 ± 3.1	0.846 ± 0.048	31.4 ± 3.4	0.795 ± 0.116	26.9 ± 4.1	0.909 ± 0.064	29.4 ± 4.3
Zero-Filled		0.766 ± 0.084	17.3 ± 2.0	0.674 ± 0.031	17.3 ± 1.1	0.703 ± 0.120	16.8 ± 4.2	0.806 ± 0.062	21.5 ± 3.8

Table 3. Independent evaluation of model performance (first column) on the 3D FLAIR MS Brains dataset for different training datasets (second column). The reported figures collate: contrast resolution (CR, higher is better) of MS lesions, gradient magnitude white matter noise (WMN, lower is better), background noise (BGN, lower is better) and weighted average (WA, with negative CR and relative to maximum scores such that lower is better), respectively. For each model and dataset, the mean and standard deviation on each metric is given. The best scores are underlined and other high scores highlighted in bold. Methods are sorted in alphabetical order.

FLAIR MS Brains—Variable Density Poisson 7.5x					
Method	Trained Dataset	CR↑	WMN ↓	BGN ↓	WA ↓
CascadeNet	T_1-Brain	0.128 ± 0.028	0.135 ± 0.022	0.292 ± 0.078	1.08
	T_2-Knee	0.087 ± 0.040	0.290 ± 0.059	0.302 ± 0.083	1.43
	FLAIR-Brain	0.145 ± 0.030	0.126 ± 0.016	0.265 ± 0.071	0.96
	FLAIR-Brain 1D	0.139 ± 0.025	0.121 ± 0.016	0.309 ± 0.068	1.05
CIRIM	T_1-Brain	0.179 ± 0.025	0.145 ± 0.030	0.172 ± 0.092	0.69
	T_2-Knee	0.097 ± 0.020	0.285 ± 0.044	0.322 ± 0.053	1.42
	FLAIR-Brain	**0.183 ± 0.025**	0.131 ± 0.029	**0.104 ± 0.085**	**0.55**
	FLAIR-Brain 1D	0.173 ± 0.030	**0.110 ± 0.017**	0.137 ± 0.074	**0.62**
E2EVN	T_1-Brain	0.145 ± 0.034	0.144 ± 0.010	0.359 ± 0.095	1.13
	T_2-Knee	0.109 ± 0.028	0.301 ± 0.042	0.576 ± 0.352	1.79
	FLAIR-Brain	0.159 ± 0.041	0.116 ± 0.014	0.358 ± 0.064	1.03
	FLAIR-Brain 1D	0.134 ± 0.035	0.141 ± 0.020	0.360 ± 0.058	1.17
IRIM	T_1-Brain	0.159 ± 0.025	0.128 ± 0.027	0.200 ± 0.089	0.80
	T_2-Knee	0.078 ± 0.021	0.260 ± 0.122	0.348 ± 0.061	1.51
	FLAIR-Brain	0.169 ± 0.027	0.145 ± 0.020	0.181 ± 0.126	0.74
	FLAIR-Brain 1D	0.176 ± 0.025	0.151 ± 0.020	0.213 ± 0.081	0.77
KIKINet	T_1-Brain	0.117 ± 0.032	0.184 ± 0.042	0.432 ± 0.075	1.40
	T_2-Knee	0.149 ± 0.026	0.235 ± 0.032	0.294 ± 0.087	1.10
	FLAIR-Brain	0.105 ± 0.077	0.175 ± 0.040	0.626 ± 0.141	1.75
	FLAIR-Brain 1D	0.103 ± 0.026	0.144 ± 0.035	0.352 ± 0.060	1.29
LPDNet	T_1-Brain	0.240 ± 0.046	0.206 ± 0.029	0.210 ± 0.056	**0.56**
	T_2-Knee	0.030 ± 0.151	0.126 ± 0.031	0.204 ± 0.065	1.34
	FLAIR-Brain	0.117 ± 0.024	**0.099 ± 0.012**	0.332 ± 0.083	1.15
	FLAIR-Brain 1D	0.066 ± 0.029	0.129 ± 0.024	0.338 ± 0.070	1.40
RIM	T_1-Brain	0.178 ± 0.025	0.168 ± 0.026	0.170 ± 0.093	0.71
	T_2-Knee	0.091 ± 0.036	0.149 ± 0.030	0.251 ± 0.091	1.18
	FLAIR-Brain	**0.197 ± 0.029**	0.175 ± 0.025	**0.134 ± 0.078**	**0.58**
	FLAIR-Brain 1D	**0.183 ± 0.027**	0.158 ± 0.026	0.165 ± 0.074	0.67
UNet	T_1-Brain	0.182 ± 0.034	0.174 ± 0.022	0.276 ± 0.069	0.87
	T_2-Knee	0.125 ± 0.040	0.924 ± 0.084	0.285 ± 0.089	1.93
	FLAIR-Brain	0.087 ± 0.027	0.079 ± 0.010	0.625 ± 0.137	1.72
	FLAIR-Brain 1D	0.065 ± 0.021	0.105 ± 0.023	0.348 ± 0.070	1.40
PICS		0.178 ± 0.025	0.140 ± 0.018	**0.147 ± 0.092**	0.64
Zero-Filled		0.072 ± 0.023	0.092 ± 0.017	0.372 ± 0.064	1.39

can be found in the supplementary material, as well as further evaluation for four-, six-, and eight-times acceleration for Gaussian 2D undersampling.

The trained models on each dataset and undersampling scheme were used to evaluate performance on out-of-training distribution data, containing MS lesions. As summarized in table 3, the performance is evaluated quantitatively by measuring the CR of the reconstructed lesions, the WMN, the background noise (BGN) and a WA. A combination of high CR, low WMN and low BGN yields a low WA and reflects highly accurate reconstruction (figures 6, S7, S8), such as in the CIRIM FLAIR-Brain model and PICS. The models trained on the FLAIR-Brain, the FLAIR-Brain 1D, and the T_1-Brain datasets scored high on CR and low on WMN compared to the T_2-Knee trained models. The CIRIM and the RIM achieved the lowest BGN. The CascadeNet, the E2EVN, the KIKINet, and the UNet models reported high BGN, in general corresponding to more aliased reconstruction. The LPDNet achieved high CR and relatively low WMN and BGN, but the observed reconstruction quality was poor. This also highlighted the need for combined metrics and qualitative evaluation to evaluate performance.

Figure 7 shows reconstructions of a coronal slice from the MS FLAIR-Brain dataset. Visually, the CIRIM, PICS, RIM, and IRIM reconstructions appear similar. The E2EVN and the CascadeNet showed inhomogeneous intensities and high contrast deviations. The LPDNet showed more aliased reconstructions, with lower contrast levels. The KIKINet and the UNet seemed in our experiments not able to resolve background noise and in general resulted in more distorted images. Example reconstructions of two more subjects including axial and sagittal plane reconstructions can be found in the supplementary material.

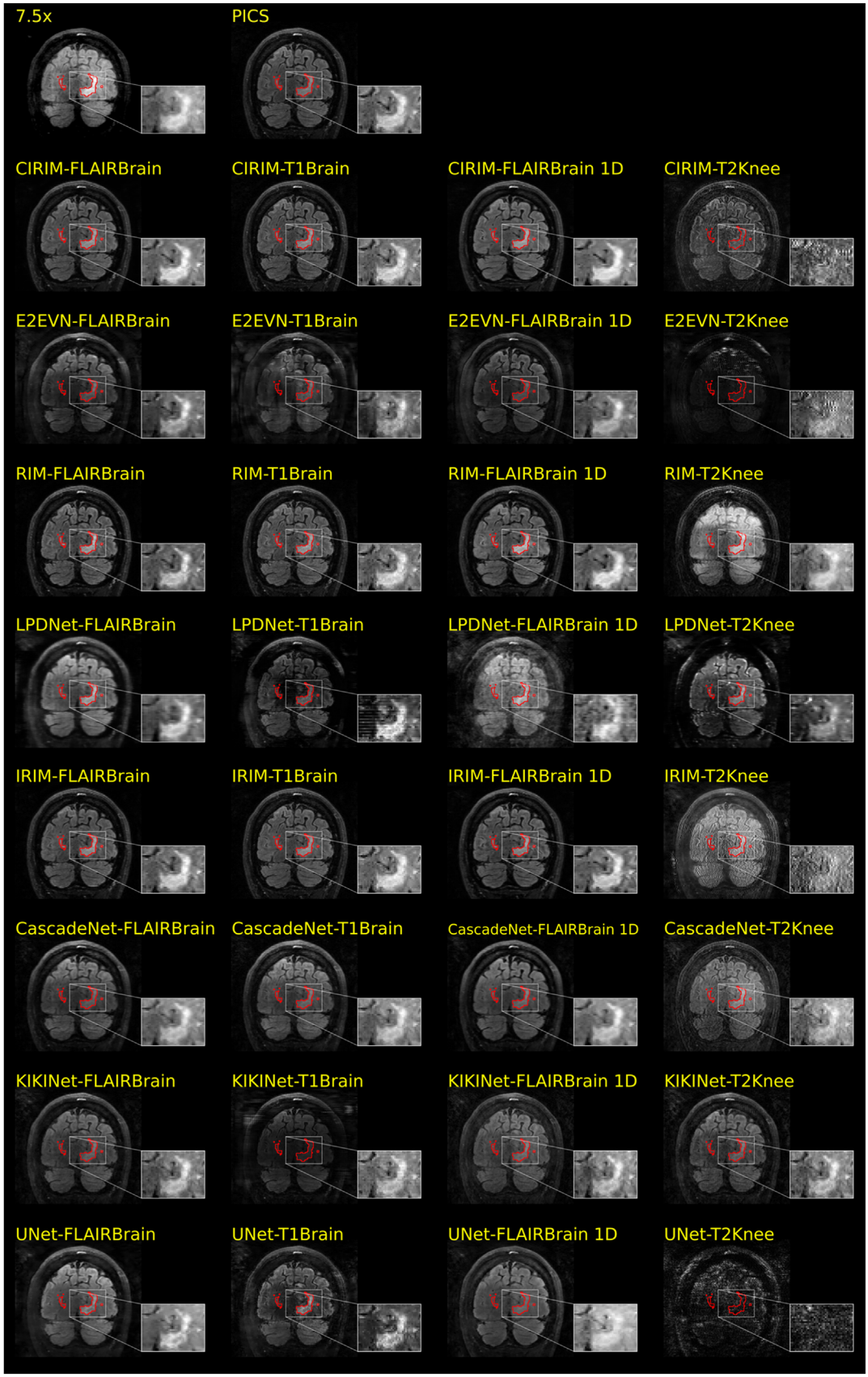

Figure 7. Reconstructions of a representative coronal slice of a 7.5x accelerated 3D FLAIR scan of a MS patient. Segmented MS lesions are depicted with red colored contours. Shown is the aliased linear reconstruction (first row-first image), PICS (first row-second image), and models' reconstructions on each trained scheme: the FLAIR-Brain dataset with Gaussian 2D undersampling (second-last row, first column), the T_1-Brain dataset with Gaussian 2D undersampling (second-last row, second column), the FLAIR-Brain dataset with equidistant 1D undersampling (second-last row, third column), and the T_2-Knee dataset with Gaussian 2D undersampling (second-last row, fourth column). The inset on the right bottom of each reconstruction focuses on a lesion region with high spatial detail.

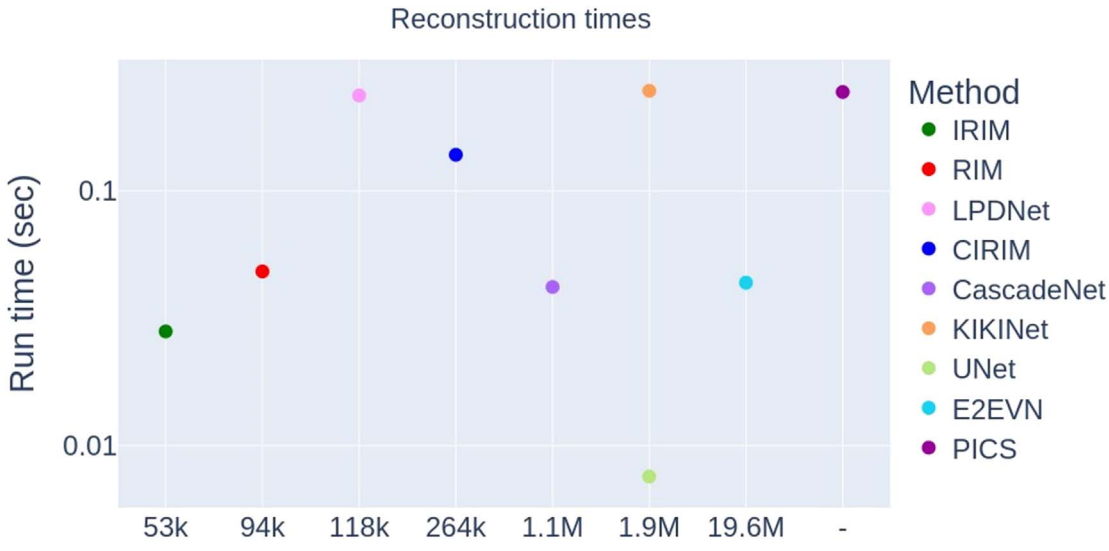

Figure 8. Inference times for reconstructing one volume from the FLAIR brains dataset using different methods. The *x*-axis represents methods' number of trainable parameters. The *y*-axis shows the run time in seconds.

Finally, in figure 8, the reconstruction times of all methods are reported. As input, one volume from the trained fastMRI FLAIR brains dataset was taken, consisting of fifteen slices cropped to a matrix size of 320×320. The KIKINet, PICS, and the LPDNet were the slowest methods, requiring 247 ms, 245 ms, and 237 ms respectively to reconstruct the volume. The CIRIM needed 139 ms, the RIM 48 ms, the E2EVN 44 ms, the CascadeNet 42 ms, the IRIM 28 ms, and the UNet 8 ms.

4. Discussion

In this paper, we proposed the CIRIM, for a balanced increase in model complexity while maintaining generalization capabilities. We assessed DC both implicitly through unrolled optimization by gradient descent and explicitly by a formulated term. Robustness was evaluated by reconstructing sparsely sampled MRI data containing unseen pathology. The CIRIM was extensively compared to another unrolled network, the E2EVN, and a range of other methods.

In experiments reconstructing brain and knee data containing different contrasts, the proposed CIRIM performed best, with promising generalization capabilities. On the T_2-knee dataset, the E2EVN performed equivalently to the CIRIM, while on the T_1-brain and the FLAIR-brain datasets for eight- and ten-times acceleration, the measured PSNR dropped by approximately 5% of what compared to what. Visually, this reflected in missing anatomical details such as vessels. The LPDNet, the RIM, and the IRIM performed comparable but slightly worse than the CIRIM. The CascadeNet and the KIKINet, dropped further in SSIM and PSNR on all trained datasets, resulting in more noisy reconstructions. PICS and the UNet showed most of the time overly smoothed results. Interestingly, for 1D undersampling the CascadeNet showed comparable performance to the CIRIM, but it was more sensitive to banding artifacts.

The RIM-based models (RIM, IRIM, CIRIM), trained on FLAIR and T_1-weighted brain data, and PICS, could accurately reconstruct Multiple Sclerosis lesions unseen during training. Spatial detail when reconstructing MS lesions was preserved with better denoised images, compared to, e.g. the CascadeNet. The E2EVN and the LPDNet did not show any significant improvement in this respect. The reason for such behavior might be that these scans, in contrast to the training data, came without a fully sampled center since a separate reference scan was acquired. This deviation from the training data could explain the lower performance of some of the models. Conditional deep priors tend to learn dealiasing of undersampled acquisitions on images that they have trained on. In such a situation, learning k-space corrections might be disadvantageous. The KIKINet and the UNet performed significantly worse than the other methods, thereby appearing to be sensitive to noisy inputs. Furthermore, the models trained on knees were inadequate in reconstructing MS lesions, indicating training anatomy preference rather than generalization. Remarkably, this was also realized by the performance of the networks trained on the FLAIR-Brain datasets with equidistant 1D undersampling. All models generalized well on reconstructing the MS data, despite the deviating undersampling scheme (variable density poisson sampling in 2D).

The SNR, the number of coils, and the size of the training dataset appeared to be important parameters that influenced performance. This is to be seen in the reported SSIM and PSNR scores. Here, the E2EVN models performed highest on the largest dataset, i.e. the T_2-weighted knee dataset, which contained approximately 12 000 slices. However, all models experienced lower performance due to lower SNR (17.1 $\pm$ 4.5) and number of coils (8), compared to the T_1-weighted brain dataset (3000 slices, SNR of 25.7 $\pm$ 5.4, and 32 coils). The FLAIR brain dataset, despite its relatively high SNR (5000 slices and SNR of 23.6 $\pm$ 4.8), did not necessarily yield high quality in reconstructed images. The deviating number of coils (from 2 to 24), field strength (1.5 T an 3 T), and matrix sizes, resulted in a challenging dataset to converge with when training a model. In this situation, the advantage of implementing cascades was most apparent, making the CIRIM being robust with all tested acceleration factors (4x, 6x, 8x—supplementary material and 10x—table 2). PICS and the UNet scored overall lower, illustrating that learning a prior with an efficient model is advantageous.

Importantly, our results show that the RIM-based models can reconstruct image details unseen during training. The RIM explicitly contains a formulation of the prior information of an MR image and acts as optimizer itself. Unrolled optimization is performed by gradient descent (Putzky and Welling 2017), such that DC is enforced implicitly. The CIRIM allows to further denoise the reconstructed images through the cascades without sharing parameters, similar to previously proposed deep cascading networks (Schlemper *et al* 2018, Huang *et al* 2019, Chen *et al* 2021). The cascades thereby allowed us to train an overall deep network of multiple connected RNNs that captures long-range dependencies while avoiding vanishing or exploding gradients. The E2EVN also performs unrolled optimization through cascades, but explicitly enforces DC with a formulated term.

Recent work has pointed out the importance of benchmarking and quantifying the performance of deep networks regarding the GPU memory required for training, the inference times, the applications, and the optimization (Wang *et al*, Ramzi *et al* 2020b, Hammernik *et al* 2021). With regard to inference times, methods such as the LPDNet and the KIKINet did not seem to improve in speed over the conventional CS algorithm, implemented on the GPU. The reason for these methods being slower is that they consist of deep feed-forward large convolutional layers. The RIM, the E2EVN, and the CascadeNet reduce reconstruction times by a factor of six compared to CS. Here, inference is performed over an iterative scheme, in which sharing of parameters is optimized either through time-steps or cascades. The IRIM and the UNet even further reduce the time by a factor of two and six, respectively. The CIRIM serves as a balanced deep network, being two times faster than the slowest methods and two times slower than the other cascading networks. The performance gain in further denoising and generalization capabilities may counterbalance the need for longer inference times.

5. Conclusion

The CIRIM implicitly enforces DC when targeting unrolled optimization through gradient descent. The comparable E2EVN performed best when DC was explicitly enforced, performing well on the training distributions. However, it appeared to be inadequate on reconstructing out-of-training distribution data without a fully sampled center. The CIRIM performed best on all training datasets, tested undersampling schemes and acceleration factors. Also, it showed robust performance on reconstructing accelerated FLAIR data containing MS lesions, achieving good lesion contrast and efficient denoising compared to PICS, the baseline RIM and the IRIM. In contrast, methods such as the CascadeNet and the LPDNet were sensitive to highly noisy untrained data, showing limited generalization capabilities. The KIKINet and the UNet tended to oversimplify the reconstructed images, performing markedly worse than rest methods. To that extent, the impression is that evaluating the forward process of accelerated MRI reconstruction, frequently through time, is of great importance for generalization in other settings. The implemented cascades and the application of the RIM to a deeper network allowed backpropagation on a smaller number of time-steps but on higher frequency for each iteration. Thus, a key advantage of the CIRIM is that it maintains a very fair trade-off between reconstructed image quality and fast reconstruction times, which is crucial in the clinical workflow.

Acknowledgments

This publication is based on the STAIRS project under the TKI-PPP program. The collaboration project is co-funded by the PPP Allowance made available by Health~Holland, Top Sector Life Sciences & Health, to stimulate public-private partnerships.

M W A Caan is shareholder of Nico.Lab International Ltd.

Appendix.

Gated recurrent unit (GRU)

The GRU has two gating units, the reset gate and the update gate. These gates control how the information flows in the network. The update gate regulates the update to a new hidden state, whereas the reset gate controls the information to forget. Both gates act in a probabilistic manner.

The activation of the reset gate $\boldsymbol{r}$ at iteration $\boldsymbol{\tau}$, for updating equation (9), is computed by

$$r_\tau = \sigma(W_r[s_{\tau-1}, x_\tau] + b_r).$$

$\boldsymbol{\sigma}$ is the logistic sigmoid function, $\boldsymbol{x_\tau}$ and $\boldsymbol{s_{\tau-1}}$ are the input and the previous hidden state, respectively. $\boldsymbol{W_r}$ and $\boldsymbol{b_r}$ are the weights matrix and the learned bias vector.

Similarly, the update gate $\boldsymbol{z}$ is computed by

$$z_\tau = \sigma(W_z[s_{\tau-1}, x_\tau] + b_z).$$

The actual activation of the next hidden state $\boldsymbol{s_\tau}$ is then computed by

$$s_\tau = (1 - z_\tau) \odot s_{\tau-1} + z_\tau \odot \tilde{s}_\tau,$$

where $\odot$ represents the Hadamard product and $\tilde{s}_\tau$ is given by

$$\widetilde{s_\tau} = \tanh(\mathrm{W}_s[r_\tau \odot s_{\tau-1}, x_\tau] + b_s).$$

Independently Recurrent Neural Network (IndRNN)

The IndRNN addresses gradient decay over iterations, following an independent neuron connectivity within a recurrent layer. The update on equation (9) and at iteration $\boldsymbol{\tau}$ is given by

$$s_\tau = \sigma(W_{x_\tau} + u \odot s_{\tau-1} + b),$$

where $\boldsymbol{W}$ is the weight for the current input, $\boldsymbol{u}$ is the weight for the recurrent input, and $\boldsymbol{b}$ is the bias vector.

ORCID iDs

D Karkalousos https://orcid.org/0000-0001-5983-0322
M W A Caan https://orcid.org/0000-0002-5162-8880

References

Adler J and Öktem O 2018 Learned primal-dual reconstruction *IEEE Trans. Med. Imaging* **37** 1322–32

Aggarwal H K, Mani M P and Jacob M 2019 MoDL: model-based deep learning architecture for inverse problems *IEEE Trans. Med. Imaging* **38** 394–405

Akçakaya M, Moeller S, Weingärtner S and Uğurbil K 2019 Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: database-free deep learning for fast imaging *Magn. Reson. Med.* **81** 439–53

Amin M F, Amin M I, Al-Nuaimi A Y H and Murase K 2011 Wirtinger calculus based gradient descent and Levenberg-Marquardt learning algorithms in complex-valued neural networks *Lecture Notes Comput. Sci.* **7062** 550–9

Andrychowicz Marcin, Denil Misha, Colmenarejo Sergio Gómez, Hoffman Matthew W, Pfau David, Schaul Tom, Shillingford Brendan and Freitas Nando 2016 Learning to learn by gradient descent by gradient descent *Proceedings of the 30th International Conference on Neural Information Processing Systems* 3988–96

Arefeen Y, Beker O, Cho J, Yu H, Adalsteinsson E and Bilgic B 2022 Scan-specific artifact reduction in k-space (SPARK) neural networks synergize with physics-based reconstruction to accelerate MRI *Magn. Reson. Med.* **87** 764–80

Beauferris Y *et al* 2020 Multi-channel MR reconstruction (MC-MRRec) challenge—comparing accelerated MR reconstruction models and assessing their genereralizability to datasets collected with different coils pp 1–31 arXiv:2011.07952

Chambolle A and Pock T 2011 A first-order primal-dual algorithm for convex problems with applications to imaging *J. Math. Imaging Vis.* **40** 120–45

Chen E Z, Wang P, Chen X, Chen T and Sun S 2019 Pyramid convolutional RNN for MRI image reconstruction arXiv:1912.00543

Chen J, Zhang P, Liu H, Xu L and Zhang H 2021 Spatio-temporal multi-task network cascade for accurate assessment of cardiac CT perfusion *Med. Image Anal.* **74** 102207

Chen Y, Yu W and Pock T 2015 On learning optimized reaction diffusion processes for effective image restoration arXiv:1503.05768

Cho K *et al* 2014 Learning phrase representations using RNN encoder-decoder for statistical machine translation *EMNLP 2014—2014 Conf. on Empirical Methods in Natural Language Processing, Proc. of the Conf.* pp 1724–34

Cole E, Cheng J, Pauly J and Vasanawala S 2021 Analysis of deep complex-valued convolutional neural networks for MRI reconstruction and phase-focused applications *Magn. Reson. Med.* **86** 1093–109

Cole E K, Pauly J M, Vasanawala S S and Ong F 2020 Unsupervised MRI Reconstruction with Generative Adversarial Networks arXiv:2008.13065

Dar S U H, Özbey M, Çatlı A B and Çukur T 2020b A transfer-learning approach for accelerated MRI using deep neural networks *Magn. Reson. Med.* **84** 663–85

Dar S U H, Yurt M, Shahdloo M, Ildiz M E, Tinaz B and Cukur T 2020a Prior-guided image reconstruction for accelerated multi-contrast mri via generative adversarial networks *IEEE J. Sel. Top. Signal Process.* **14** 1072–87
Duan J *et al* 2019 Vs-net: variable splitting network for accelerated parallel MRI reconstruction *MICCAI 2019. Lecture Notes in Computer Science()* **11767** 713–22
Eo T, Jun Y, Kim T, Jang J, Lee H J and Hwang D 2018 KIKI-net: cross-domain convolutional neural networks for reconstructing undersampled magnetic resonance images *Magn. Reson. Med.* **80** 2188–201
Epperson K *et al* 2013 Creation of fully sampled MR data repository for compressed sensing of the knee *SMRT Conf.* **2013** 1
Falcon W and Team T P L (2019) PyTorch Lightning *The lightweight PyTorch wrapper for high-performance AI research. Scale your models, not the boilerplate.* (https://doi.org/10.5281/zenodo.3828935)
Hammernik K *et al* 2018 Learning a variational network for reconstruction of accelerated MRI data *Magn. Reson. Med.* **79** 3055–71
Hammernik K, Schlemper J, Qin C, Duan J, Summers R M and Rueckert D 2021 Systematic evaluation of iterative deep neural networks for fast parallel MRI reconstruction with sensitivity-weighted coil combination *Magn. Reson. Med.* **86** 1859–72
Huang Q, Yang D, Wu P, Qu H, Yi J and Metaxas D 2019 MRI reconstruction via cascaded channel-wise attention network *2019 IEEE 16th Int. Symp. on Biomedical Imaging (ISBI 2019) (Piscataway, NJ)* (IEEE) pp 1622–6
Jun Y, Shin H, Eo T and Hwang D 2021 Joint deep model-based Mr image and coil sensitivity reconstruction network (Joint-ICNet) for fast MRI *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* pp 5266–75
Kim T H, Garg P and Haldar J P 2019 LORAKI: Autocalibrated Recurrent Neural Networks for Autoregressive MRI Reconstruction in k-Space arXiv:1904.09390
Kingma D P and Ba J L 2015 Adam: a method for stochastic optimization 1412.6980 3rd International Conference on Learning Representations (ICLR 2015)
Knoll F *et al* 2020 Advancing machine learning for MR image reconstruction with an open competition: overview of the 2019 fastMRI challenge *Magn. Reson. Med.* **84** 3054–370
Korkmaz Y, Dar S U and Yurt M 2021 Unsupervised MRI Reconstruction via Zero-Shot Learned Adversarial Transformers arXiv:2105.08059v2
Kuchaiev O *et al* 2019 NeMo: a toolkit for building AI applications using Neural Modules arXiv:1909.09577
Li S, Li W, Cook C, Zhu C and Gao Y 2018 Independently recurrent neural network (IndRNN): building a longer and deeper RNN *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* pp 5457–66
Lustig M, Donoho D and Pauly J M 2007 Sparse MRI: the application of compressed sensing for rapid MR imaging *Magn. Reson. Med.* **58** 1182–95
Lustig M, Donoho D L, Santos J M and Pauly J M 2008 Compressed sensing MRI *IEEE Signal Process Mag.* **25** 72–82
Lønning K, Putzky P, Sonke J J, Reneman L, Caan M W A and Welling M 2019 Recurrent inference machines for reconstructing heterogeneous MRI data *Med. Image Anal.* **53** 64–78
Muckley M J *et al* 2020 State-of-the-Art Machine Learning MRI Reconstruction in 2020: Results of the Second fastMRI Challenge arXiv:2012.06318
Muckley M J *et al* 2021 Results of the 2020 fastMRI challenge for machine learning MR image reconstruction *IEEE Trans. Med. Imaging* (https://doi.org/10.1109/TMI.2021.3075856)
Otsu N 1979 A threshold selection method from gray-level histograms *IEEE Trans. Syst. Man Cybern.* **9** 62–6
Pascanu R, Mikolov T and Bengio Y 2013 On the difficulty of training recurrent neural networks *30th International Conference on International Conference on Machine Learning* 328, pp III–1310–III–1318
Paszke A *et al* 2019 PyTorch: an imperative style, high-performance deep learning library arXiv:1912.01703 Adv. Neural Inf. Process. Syst.32
Penny W, Friston K, Ashburner J, Kiebel S and Nichols T 2007 *Statistical Parametric Mapping: The Analysis of Functional Brain Images* (London: Elsevier) 625–647978-0-12-372560-8
Pezzotti N *et al* 2020 An adaptive intelligence algorithm for undersampled knee MRI reconstruction *IEEE Access* **8** 204825–38
Pruessmann K P, Weiger M, Scheidegger M B and Boesiger P 1999 SENSE: sensitivity encoding for fast MRI *Magn. Reson. Med.* **42** 952–62
Putzky P, Karkalousos D, Teuwen J, Miriakov N, Bakker B, Caan M and Welling M 2019 i-RIM applied to the fastMRI challenge arXiv:1910.08952
Putzky P and Welling M. 2017 Recurrent inference machines for solving inverse problems arXiv:1706.04008
Qin C, Schlemper J, Caballero J, Price A N, Hajnal J V and Rueckert D 2019 Convolutional recurrent neural networks for dynamic MR image reconstruction *IEEE Trans. Med. Imaging* **38** 280–90
Quan T M, Nguyen-Duc T and Jeong W K 2018 Compressed sensing MRI reconstruction using a generative adversarial network with a cyclic loss *IEEE Trans. Med. Imaging* **37** 1488–97
Ramzi Z, Ciuciu P and Starck J-L 2020 XPDNet for MRI reconstruction: an application to the fastMRI 2020 brain challenge arXiv:2010.07290
Ramzi Z, Ciuciu P and Starck J L 2020b Benchmarking deep nets MRI reconstruction models on the fastmri publicly available dataset *Proc.—Int. Symp. Biomed. Imaging* **2020** 1441–5
Ronneberger O, Fischer P and Brox T 2015 U-Net: convolutional networks for biomedical image segmentation *Med. Image Comput. Comput. Assist. Interv.* **15** 234–241
Sarroff A M, Shepardson V and Casey M A 2015 Learning Representations Using Complex-Valued Nets arXiv:1511.06351
Schlemper J, Caballero J, Hajnal J V, Price A N and Rueckert D 2018 A deep cascade of convolutional neural networks for dynamic MR image reconstruction *IEEE Trans. Med. Imaging* **37** 491–503
Schlemper J, Qin C, Duan J, Summers R M and Hammernik K 2019 Σ-net: Ensembled Iterative Deep Neural Networks for Accelerated Parallel MR Image Reconstruction 1912.05480
Sim B, Oh G, Kim J, Jung C and Ye J C 2020 Optimal transport driven cycleGAN for unsupervised learning in inverse problems *SIAM J. Imag. Sci.* **13** 2281–306
Sodickson D K and Manning W J 1997 Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays *Magn. Reson. Med.* **38** 591–603
Souza R *et al* 2020 Dual-domain cascade of U-nets for multi-channel magnetic resonance image reconstruction *Magn. Reson. Imaging* **71** 140–53
Souza R, Lebel R M and Frayne R 2019 A hybrid, dual domain, cascade of convolutional neural networks for magnetic resonance image reconstruction *2nd International Conference on Medical Imaging with Deep Learning* 102, 437–46
Sriram A, Zbontar J, Murrell T, Defazio A, Zitnick C L, Yakubova N, Knoll F and Johnson P 2020 End-to-End Variational Networks for Accelerated MRI Reconstruction arXiv:2004.06688

Sriram A, Zbontar J, Murrell T, Zitnick C L, Defazio A and Sodickson D K 2020b GrappaNet: combining parallel imaging with deep learning for multi-coil MRI reconstruction *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* pp 14303–10

Strijbis V I J *et al* 2021 Multi-view convolutional neural networks for automated ocular structure and tumor segmentation in retinoblastoma *Sci. Rep.* **11** 14590

Uecker M *et al* 2014 ESPIRiT-an eigenvalue approach to autocalibrating parallel MRI: where SENSE meets GRAPPA *Magn. Reson. Med.* **71** 990–1001

Uecker Martin, Ong Frank, Tamir Jonathan I, Bahri Dara, Virtue Patrick, Cheng Joseph Y, Zhang Tao, Lustig Michael *et al* (2015) BART *Toolbox for Computational Magnetic Resonance Imaging* (https://doi.org/10.5281/zenodo.3934312)

Wang K *et al* 2021 Memory-efficient Learning for High-Dimensional MRI Reconstruction arXiv:2103.04003

Wang S *et al* 2020 DeepcomplexMRI: exploiting deep residual network for fast parallel MR imaging with complex convolution *Magn. Reson. Imaging* **68** 136–47

Wang Z, Bovik A C, Sheikh H R and Simoncelli E P 2004 Image quality assessment: from error visibility to structural similarity *IEEE Trans. Image Process.* **13** 600–12

Xiao L *et al* 2022 Partial Fourier reconstruction of complex MR images using complex-valued convolutional neural networks *Magn. Reson. Med.* **87** 999–1014

Yang G *et al* 2018 DAGAN: deep De-aliasing generative adversarial networks for fast compressed sensing MRI reconstruction *IEEE Trans. Med. Imaging* **37** 1310–21

Yang Y, Sun J, Li H and Xu Z 2017 ADMM-net: a deep learning approach for compressive sensing MRI arXiv:1705.06869

Yiasemis G, Moriakov N, Karkalousos D, Caan M and Teuwen J (2022) DIRECT *Deep Image Reconstruction Toolkit*

Zbontar J *et al* 2018 fastMRI: an open dataset and benchmarks for accelerated MRI arXiv pp 1–35

Zhang H and Mandic D P 2016 Is a complex-valued stepsize advantageous in complex-valued gradient learning algorithms? *IEEE Trans. Neural Netw. Learn. Syst.* **27** 2730–5

Zhang X, Lian Q, Yang Y and Su Y 2020 A deep unrolling network inspired by total variation for compressed sensing MRI *Digit. Signal Process.: Rev. J.* **107** 102856

Zhu B, Liu J Z, Cauley S F, Rosen B R and Rosen M S 2018 Image reconstruction by domain-transform manifold learning *Nature* **555** 487–92

Zhao R, Yaman B, Zhang Y, Stewart R, Dixon A, Knoll F, Huang Z, Lui Y W, Hansen M S and Lungren M P 2022 fastMRI+, Clinical pathology annotations for knee and brain fully sampled magnetic resonance imaging data *Scientific Data* **9** 9