Medical Image Analysis 53 (2019) 64–78

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Recurrent inference machines for reconstructing heterogeneous MRI data☆

Kai Lønning[a,b,c,*], Patrick Putzky[b,d], Jan-Jakob Sonke[c], Liesbeth Reneman[e], Matthan W.A. Caan[a,e], Max Welling[b,d]

[a] Spinoza Centre for Neuroimaging, Amsterdam 1105 BK, the Netherlands
[b] Informatics Institute at the University of Amsterdam, Amsterdam 1098 XH, the Netherlands
[c] Netherlands Cancer Institute, Amsterdam 1066 CX, the Netherlands
[d] AMLab, Amsterdam, 1098 XH, the Netherlands
[e] Amsterdam UMC, Biomedical Engineering and Physics, University of Amsterdam, Amsterdam 1105 AZ, the Netherlands

ARTICLE INFO

Article history:
Received 15 August 2018
Revised 5 January 2019
Accepted 14 January 2019
Available online 18 January 2019

Keywords:
MRI
Reconstruction
Deep learning
Inverse problems

ABSTRACT

Deep learning allows for accelerated magnetic resonance image (MRI) reconstruction, thereby shortening measurement times. Rather than using sparsifying transforms, a prerequisite in Compressed Sensing (CS), suitable MRI prior distributions are learned from data. In clinical practice, both the underlying anatomy as well as image acquisition settings vary. For this reason, deep neural networks must be able to reapply what they learn across different measurement conditions. We propose to use Recurrent Inference Machines (RIM) as a framework for accelerated MRI reconstruction. RIMs solve inverse problems in an iterative and recurrent inference procedure by repeatedly reassessing the state of their reconstruction, and subsequently making incremental adjustments to it in accordance with the forward model of accelerated MRI. RIMs learn the inferential process of reconstructing a given signal, which, in combination with the use of internal states as part of their recurrent architecture, makes them less dependent on learning the features pertaining to the source of the signal itself. This gives RIMs a low tendency to overfit, and a high capacity to generalize to unseen types of data. We demonstrate this ability with respect to anatomy by reconstructing brain and knee scans, as well as other MRI acquisition settings, by reconstructing scans of different contrast and resolution, at different field strength, subjected to varying acceleration levels. We show that RIMs outperform CS not only with respect to quality metrics, but also according to a rating given by an experienced neuroradiologist in a double blinded experiment. Finally, we show with qualitative results that our model can be applied to prospectively under-sampled raw data, as acquired by pre-installed acquisition protocols.

1. Introduction

Magnetic Resonance Imaging (MRI) is used in a wide variety of research and clinical applications measuring soft tissue in the human body. Systems at multiple field strengths deploy several measuring sequences to produce the specific contrast for the intended purpose, including T_1-, T_2-, and T_2^*-weighted images. The scanner measures data in the space of proton net-precession frequencies, known in MRI as *k*-space. Once enough samples in *k*-space are acquired to meet the Nyquist-criterion, the MR-image of tissue density can be computed through the inverse Fourier transform. However, there are physical constraints to the data acquisition process, putting a lower bound on the time it takes to fully sample k-space and produce an image (Haacke et al., 1999). As such, aspirations to reducing MR scan times amount to acquiring k-space samples below the Nyquist-criterion and reconstructing the MR-image through a dealiasing algorithm. This entails solving an inverse problem: In the context of accelerated MRI reconstruction, the forward model is a known process that describes the transformation taking the true image signal to the measured samples. What is not known, is the forward model's inverse transformation, taking the measured signal back to the true image, as this information is lost when *k*-space is sparsely sampled.

The set of possible MR-images is huge, even when restricted to a particular anatomical region, contrast mechanism or resolution. Also consider that each image spawns a large set of possible image corruptions, one for each permitted set of k-space sub-

☆ This article is part of the Special Issue on MISDL.

* Corresponding author.
E-mail address: k.lonning@nki.nl (K. Lønning).

https://doi.org/10.1016/j.media.2019.01.005
1361-8415/

samples. Using deep learning to find a function that maps each corruption back to the original signal, for all possible original signals, is a highly complex problem, requiring constraints on the solution space to be made. Traditional methods do this by careful design of features that exploit some known property inherent to the MR-images at hand. For instance, in Parallel Imaging (PI) the variations in spatial sensitivity of different signal receiver coils placed within the scanner provide redundant information, which is exploited in order to unfold aliasing artifacts caused by periodic under-sampling (Griswold et al., 2002; Pruessmann et al., 1999). Another well-established technique is Compressed Sensing (CS), which forces the reconstruction to conform to a sparse transform known to compress MR-images of a particular anatomical region (Lustig et al., 2007).

As the training of deep neural networks expands into ever more areas of applications, including medical imaging (Litjens et al., 2017), research efforts are turning away from hand-engineering features to the design of good network architectures that allow the model to learn features on its own, often leading to performance gains. We give examples of deep learning solutions to accelerated MRI reconstruction in Section 2. In this work, we apply Recurrent Inference Machines (RIM) for accelerated MRI reconstruction, which were first proposed as general inverse problem solvers in Putzky and Welling (2017). They constrain the solution space by learning an iterative process, where step-wise reassessments of the maximum a posteriori estimate lead to incremental updates that infer the inverse transform of the forward model.

Apart from breaking a complex problem into multiple subproblems, we conjecture that this iterative "meta-learning"-approach prevents the model from overfitting on the image statistics of the dataset, by shifting focus toward learning the inversion procedure itself. This should result in a model that is more invariant with respect to changes in the specific imaging settings. In this paper, we show that RIMs can accurately and efficiently reconstruct sparsely sampled MR-images at varying acceleration factors, and that their solution is robust against perturbations in subsampling points, image resolution levels, and to some extent the underlying anatomy being imaged, making them suitable candidates for distribution across different MR-acquisition set-ups.

In Section 2, a more detailed description of accelerated MRI reconstruction is given, along with an overview of previously proposed solutions. Section 3 then describes the RIM model and its implementation. Our experiments are set up to demonstrate the generalizability of RIMs and are described in Section 4, before results are presented in Section 5. Finally, we conclude with a discussion of our findings in Section 6.

2. Background and related work

2.1. The forward model

We begin by introducing the forward model of accelerated MRI reconstruction. Let $\mathbf{x} \in \mathbb{C}^n$ be the true image signal and let $\mathbf{y}_\iota \in \mathbb{C}^m$, $m \ll n$, be the set of sparsely sampled frequency signals measured in k-space by one of the scanner's c receiver coils. The measurements can then be described in terms of the true image,

$$\mathbf{y}_\iota = P\mathscr{F}S_\iota\mathbf{x} + \mathbf{n}_\iota, \quad \iota = 1, \dots, c. \tag{1}$$

Here, the MR-image is decomposed into a partial coil image through the sensitivity map S_ι, a diagonal matrix that scales every pixel by a complex number according to the spatial sensitivity of the ιth coil. This partial image is then projected onto its frequency domain through the Fourier transform $\mathscr{F}$, followed by a subsampling mask P, which reduces the dimensionality by discarding some fraction of values in k-space, thereby facilitating the acceleration in scan times. Measurements are assumed to be subjected to additive, normally distributed noise, $\mathbf{n}_\iota \sim \mathcal{N}(0, I\sigma^2) + i\mathcal{N}(0, I\sigma^2)$, $i^2 = -1$, stemming from measurement errors accumulated by the scanner. Although it is common to use a covariance matrix for the receiver coils in the forward model (Pruessmann et al., 1999), here we assume that the noise can be modeled as independent and identically distributed across coils, pixels and complex components.

The forward model is illustrated in Fig. 1, going from the true image in the top left corner, to the acquired measurements in the bottom right corner. Applying the SENSE reconstruction (Pruessmann et al., 1999) to the under-sampled k-space measurements, by taking a conjugated coil sensitivity weighted sum over the inverse Fourier transforms of the coil samples, creates the corrupted image $\sum_{\iota=1}^{c} S_\iota^H \mathscr{F}^{-1} P^T \mathbf{y}_\iota$ seen in the top right corner. This is used as a starting point in several reconstruction algorithms (Pruessmann et al., 1999; Hammernik et al., 2018; Yang et al., 2017; Hyun et al., 2018).

2.2. The maximum a posteriori solution

The goal in accelerated MRI reconstruction is to find an inverse transform of the forward model in (1), thereby mapping incomplete measurements $\mathbf{y} := \{\mathbf{y}_\iota\}_{\iota=1}^{c}$ to a high resolution image $\mathbf{x}$. A common strategy with this aim is to optimize for the maximum a posteriori (MAP) estimator from statistics, given by

$$\mathbf{x}_{MAP} = \text{argmax}_{\mathbf{x}}\{\log p(\mathbf{y}|\mathbf{x}) + \log p(\mathbf{x})\}, \tag{2}$$

which is the maximization of the sum of the log-likelihood and log-prior distributions of y and x. This is commonly reformulated as optimizing the regularized problem

$$\text{argmin}_{\mathbf{x}} \left\{ \sum_{\iota=1}^{c} d(\mathbf{y}_\iota, P\mathscr{F}S_\iota\mathbf{x}) + \lambda R(\mathbf{x}) \right\}, \tag{3}$$

where d evaluates the data consistency between the reconstruction and measurements, and R is a regularizer, with regularization factor λ, that further constrains the solution space and prevents overfitting to the data by incorporating prior knowledge about the solution.

Under the assumption of independent, identically and normally distributed measurement errors as in (1), the (negative) log-likelihood in (2), corresponding to the data consistency in (3), is given by

$$\log p(\mathbf{y}|\mathbf{x}) = \frac{1}{\sigma^2} \sum_{\iota=1}^{c} \|P\mathscr{F}S_\iota\mathbf{x} - \mathbf{y}_\iota\|_2^2 \tag{4}$$

when ignoring the normalization constant.

Whereas (4) follows explicitly from the forward model in (1), the regularizer R is a matter of model design. In CS, one takes advantage of the fact that MR-images are known to have sparse representations under a given transformation Ψ. The wavelet transform is commonly used for brain imaging, but other anatomical regions are more compressible under other transformations (Lustig et al., 2007). This leads to the regularizer $R(\mathbf{x}) = \|\Psi\mathbf{x}\|_1$, where the l_1-norm is utilized for its bias toward sparse solutions when used as a regularizer. A solution to (3) is then found through some iterative scheme. The regularization factor λ determines how much the reconstruction should prioritize constraining the solution space to signals that are sparse in Ψ's co-domain versus reconstructing an image that is consistent with the acquired data points. Setting λ too high will lead to regularization artifacts, whereas setting it too low will favor a solution too close to the sparsely sampled corrupted image.

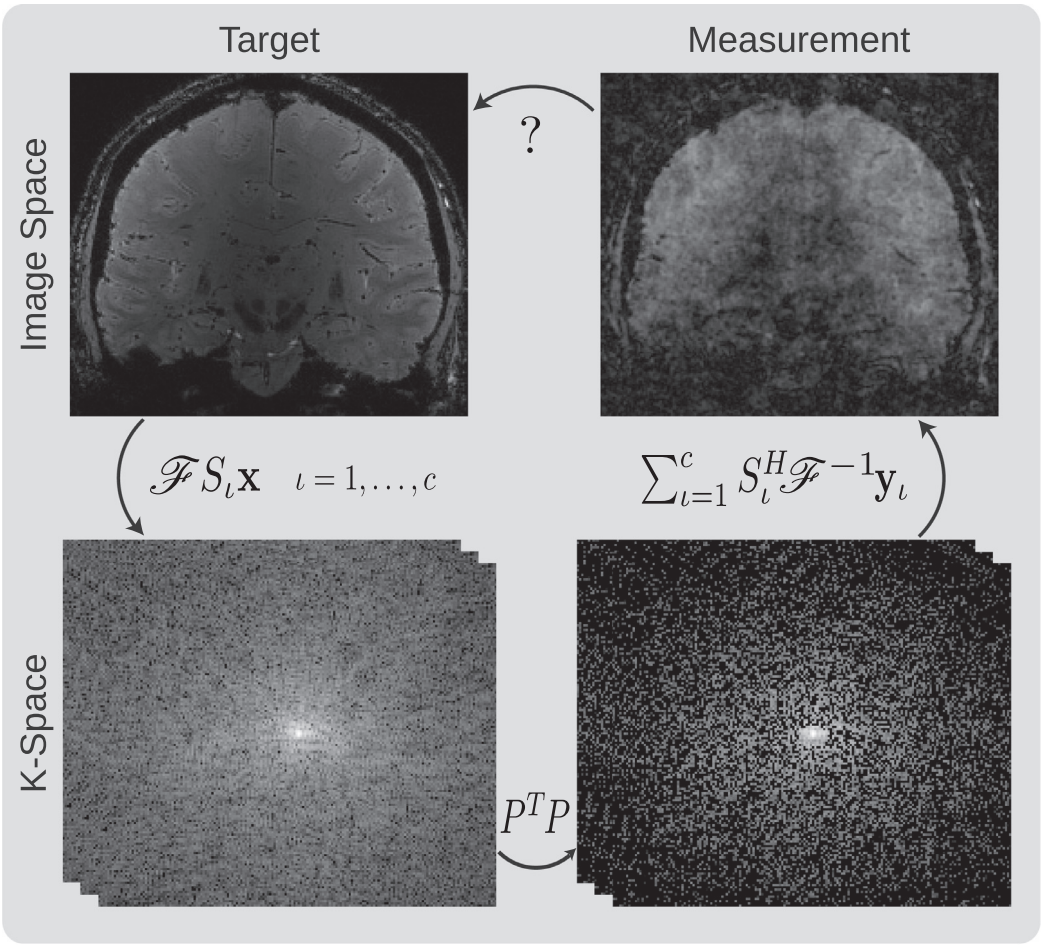


Fig. 1. The goal in MRI is to retrieve a high resolution image (top left). Measurements are done in k-space (bottom left), which is related to image space through a coil sensitivity weighted map, followed by a Fourier transform. In order to accelerate the measurement process, *k*-space is sparsely sampled (bottom right). Reconstructing the sparsely sampled k-space measurements will lead to an aliased image (top right). The goal of this work is to find a function that maps from an incomplete *k*-space (bottom right) to a high resolution image (top left).

2.3. Deep learning approaches

Deep neural networks offer the ability to learn features that capture the look and feel of a typical MR-image, thereby eliminating the need to pick a prior with some suitable sparse transformation beforehand. Another benefit is that the tuning of λ, whether explicitly included in the model or not, can be moved away from inference and into the training procedure.

Recently, there have been several deep learning proposals for accelerated MRI reconstruction. Some of these are, like CS, based on learning an iterative scheme to find (3), using the corrupted image as a starting point. Examples include the Deep ADMM-Net and the Variational Network (Yang et al., 2017; Hammernik et al., 2018). The first method uses a neural network to parametrize the alternating direction method of multipliers (ADMM), a method that uses dummy variables to solve (3) in a series of partial updates. The second method formulates (3) as a reaction-diffusion process, as described in Chen et al. (2015). Another iterative approach, the Deep Cascade of CNNs (Schlemper et al., 2018), stacks convolutional neural networks together for reconstruction, separated by k-space correction layers designed to maintain data consistency.

In the aforementioned methods, $\mathbf{x}$ is retrieved through an iterative process, where each iteration is parametrized by a separate set of network parameters. RIMs also learn an iterative process, but with a recurrent architecture where parameters are shared across iterations, using internal and external states to distinguish the task of one iterative pass from the next. We do not use the same input, but this approach bears some resemblance to the model in Andrychowicz et al. (2016), where recurrent neural networks are trained to learn gradient descent schemes by using the gradient of the objective function as the network's input for each time-step.

Not all deep learning methods for accelerated MRI reconstruction are iterative, such as the U-net architecture. Originally designed for image segmentation tasks (Ronneberger et al., 2015), the U-net has been repurposed for solving inverse problems in both CT- and MRI-settings (Jin et al., 2017; Hyun et al., 2018). Taking the corrupted image as input, U-nets consist of two parts. The first part extracts features from local patches in the input image through the standard CNN architecture of combined convolutions and max pooling layers, while steadily increasing the number of feature maps. This enables the network to extract a large number of features, but at the cost of losing global context needed for reconstruction. The second part of the U-net seeks to remedy this by using unpooling layers in order to upscale the extracted feature maps back to the original size of the input image. After each unpooling layer, previous feature maps at that resolution level are concatenated with the unpooling output, enabling the network to combine the extracted features with the spatial context that was lost due to max pooling. The U-net's non-iterative solution is great for fast reconstruction, but may come at the cost of more easily

overfitting on the training data. In this paper, we will illustrate this effect as compared to the RIM.

Finally, not all methods use $\sum_{\iota=1}^{c} S_\iota^H \mathscr{F}^{-1} P^T \mathbf{y}_\iota$ as a starting point. Instead of confining reconstruction to the image space, another non-iterative approach described in Zhu et al. (2018) learns a mapping from the sparsely sampled k-space measurements to the fully sampled image directly. A down-side to this approach is that, due to the inverse Fourier transform mapping each point in k-space to all points in image space, it requires the use of fully connected layers.

3. Recurrent inference machines

3.1. The RIM update equations

When applied to accelerated MRI reconstruction, RIMs aim to optimize (3) by learning an iterative scheme over t recurrent time-steps. Each time-step receives information on the current state of the reconstruction process as an input, yielding an incremental step $\Delta\mathbf{x}_\tau$ to take in image space as output. One of these inputs is the gradient of the log-likelihood in (4), given by

$$\nabla_{\mathbf{y}|\mathbf{x}_\tau} := \frac{1}{\sigma^2} \sum_{\iota=1}^{c} S_\iota^H \mathscr{F}^{-1} P^T (P \mathscr{F} S_\iota \mathbf{x}_\tau - \mathbf{y}_\iota) \tag{5}$$

at time-step τ. As for the problem of evaluating the gradient of the log-prior distribution in (2), this is solved by passing the current estimate, or external state, $\mathbf{x}_\tau$ as an input to the network, so that any function that would implicitly approximate the log-prior gradient can be evaluated at $\mathbf{x}_\tau$.

Let the RIM network be denoted by h, such that each pass through h produces the next incremental update $\Delta\mathbf{x}_\tau$. The RIM update equations are then given by

$$\begin{aligned} &\mathbf{s}_0 = \mathbf{0}, && \mathbf{x}_0 = \sum_{\iota=1}^{c} S_\iota^H \mathscr{F}^{-1} P^T \mathbf{y}_\iota, \\ &\boldsymbol{s}_{\tau+1} = g\left(\nabla_{\mathbf{y}|\mathbf{x}_\tau}, \mathbf{x}_\tau, \mathbf{s}_\tau\right), && \mathbf{x}_{\tau+1} = \mathbf{x}_\tau + h\left(\nabla_{\mathbf{y}|\mathbf{x}_\tau}, \mathbf{x}_\tau, \mathbf{s}_{\tau+1}\right), \end{aligned} \tag{6}$$

for $0 \le \tau < t$. g is simply the part of the network responsible for producing the next internal state $\mathbf{s}_{\tau+1}$, which the RIM needs in order to keep track of iterations and modify its behaviour based on the progression of the inference procedure.

3.2. Update function

The update function h was implemented using a sequence of alternating convolutional layers and gated recurrent unit (GRU) cells. The first two convolutional layers are followed by ReLU activation functions before the feature maps are passed to the GRUs. The GRU cells work as described in Cho et al. (2014). They are assigned the task of maintaining the internal state, meaning that in practice there are two internal states $\mathbf{s} = \{\mathbf{s}^1, \mathbf{s}^2\}$ represented by $\boldsymbol{s}$ in (6). Fig. 2 illustrates the way in which all these layers were assembled.

To produce the input of the first convolutional layer, the external state $\mathbf{x}_\tau$ is simply concatenated with the current log-likelihood gradient $\nabla_{\mathbf{y}|\mathbf{x}_\tau}$ along the channel dimension, resulting in 4 input channels due to the complex components also being given separate channels. This first layer is implemented with a kernel size of 5×5, whereas the next two convolutions have kernel sizes 3×3. All convolutions are padded to retain the same image size throughout the network.

We will take f to mean the number of features in the GRU cells' internal states and the number of feature maps produced by the convolutional layers. This hyper-parameter is kept the same through-out all internal layers, before the final convolutional layer outputs the complex-valued image update $\Delta\mathbf{x}_\tau$. Note that the GRU cells' weights are shared across image pixels, but differ across the feature maps produced by the convolutional layers, allowing the network to process images of any given size.

3.3. Loss function

We use the mean square error (MSE) as a loss function, where the estimate $\mathbf{x}_\tau$ is evaluated against the true image $\mathbf{x}$ for each time-step. The total loss to minimize is then given by the weighted sum of MSE over all time-steps.

$$L(\mathbf{x}_t) = \frac{1}{nt} \sum_{\tau=1}^{t} w_\tau \|\mathbf{x}_\tau - \mathbf{x}\|_2^2. \tag{7}$$

As before, n is the number of image pixels, and t is the number of time-steps trained on. w_τ determines the image quality emphasis to put on reconstruction τ relative to the other $t-1$ estimates.

4. Experiments

4.1. Dataset

For training, validation, and testing purposes, three different types of raw complex-valued multi-coil data were used. A sample of reconstructed fully sampled images from the test set is shown in Fig. 3.

On a 3.0T Philips Ingenia scanner (Philips Healthcare, Best, The Netherlands) equipped with a 32-channel head coil, T_1-weighted three-dimensional (3D) magnetization prepared rapid gradient echo (MPRAGE) data of the human brain were acquired with an isotropic resolution of $1.0\,\text{mm}^3$ and FOV $256 \times 240\,\text{mm}^2$, matrix size 256×240, 225 slices with sagittal slice encoding direction, TFE factor 150, shot interval 2500 ms, inversion delay 900 ms, flip angle 9°, and first order shimming. The data were fully sampled with an elliptical shutter, such that the total scanning time was 10.8 min.

On a 7.0T Philips Achieva scanner (Achieva, Philips Healthcare, Cleveland, USA) equipped with a 32-channel Nova head coil, 3D T_2^*-weighted multi-echo FLASH data of the human brain were acquired with an isotropic resolution of $0.7\,\text{mm}^3$ and Field-of-View (FOV) $224 \times 224 \times 126\,\text{mm}^3$, matrix size 320×180, 320 slices with transverse slice encoding direction, 6 echoes with echo times (TEs) ranging from 3 ms to 21 ms, repetition time (TR) 23.4 ms, flip angle 12°, and second order imagebased B_0-shimming. The data were fully sampled with an elliptical shutter, such that the total scanning time was 22.5 min.

On both scanners, raw data were exported and stored for offline reconstruction experiments. 12 healthy subjects were included, from whom written informed consent (under an institutionally approved protocol) was obtained beforehand. Two scans were made of each subject, once on both scanners. The subjects were then divided the same way for both scanners into training, validation, and test sets, such that no subject in the validation or test set of one scanner appeared in the training set of the other. The training sets consisted of 10 subjects, whereas one subject was used for model selection, and another for final evaluation on the test sets.

Finally, we use the dataset described in Sawyer et al. (2013), which contains knee scans of 20 subjects.[1] On a 3T GE scanner, equipped with an 8-channel knee coil, T_2-weighted data were acquired using a fast spin-echo protocol at a FOV of $160 \times 160\,\text{mm}^2$, with a matrix size 320×320, and 256 slices with slice thickness 0.6 mm, creating a dataset with anisotropic image resolution of 0.5–0.6 mm. The scans took around 15.3 min per subject. The 10th

[1] The fully sampled knee dataset can be found here: http://mridata.org/fullysampled/knees.

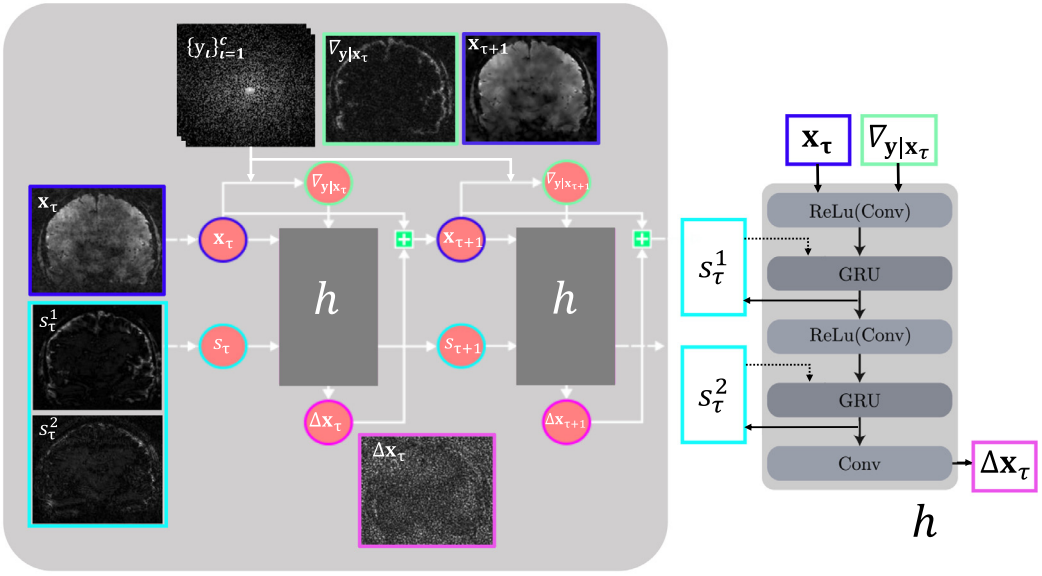


Fig. 2. The Recurrent Inference Machine (RIM) update function h as implemented in this work. All images show magnitudes scaled individually. Magnitudes of intermediate internal states $\mathbf{s}^1$ and $\mathbf{s}^2$ were averaged over features. Bold lines depict connections within a single time-step, whereas dotted lines represent recurrent connections that pass information to the next time-step.

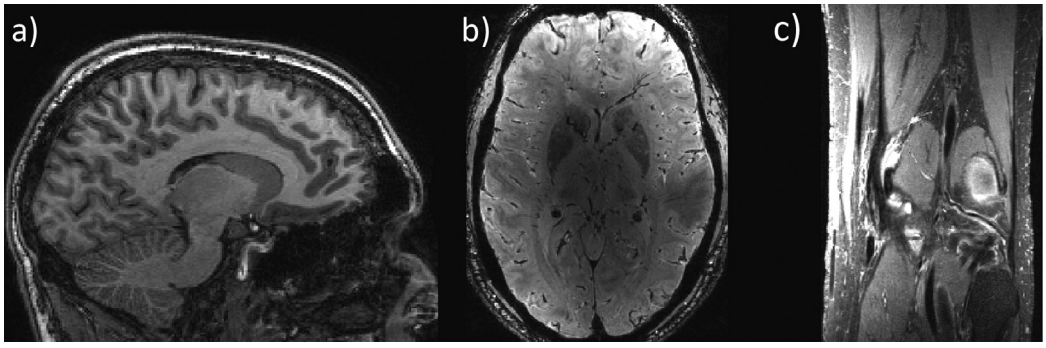


Fig. 3. Sample images from the test sets of the three different types of data used in this work, reconstructed from fully sampled raw k-space data. (a) T_1-weighted brain images acquired from a 3T scanner at 1.0 mm resolution. (b) T_2^*-weighted brain images acquired from a 7T scanner at 0.7 mm resolution. (c) T_2-weighted knee images acquired from a 3T scanner at an anisotropic resolution of 0.5-0.6 mm. (a) and (b) are of the same subject, but were made in separate MR-scanners under different acquisition protocols.

subject was discarded due to motion artifacts, and subjects 19 and 18 were used for model selection and evaluation, respectively. The remaining 17 subjects were used for training.

Coil sensitivities were estimated from the data using auto-calibration (Uecker et al., 2014), and for each subject, the full image volume was normalized with respect to the maximum magnitude after the partial coil images had been combined. For data augmentation purposes, models were trained on randomly cropped patches, which were randomly rotated, flipped and mirrored. At this stage, under-sampled data were acquired retrospectively by applying the forward model in (1) for randomly generated Ps, to be further detailed in Section 4.2.

This concludes the description of the data used for training, validation and testing. We made additional scans, of a single separate subject on the 3T Philips Ingenia scanner described above, in order to verify that our algorithm also works on prospectively under-sampled data. For these scans, we acquired brain scans using two sequences: a T_1-weighted MPRAGE protocol, and T_2-weighted TSE protocol, both with cartesian 3D-acquisitions and a resolution of 1.0 mm. For the MPRAGE protocol, six scans were made sequentially, one fully sampled scan and 5 prospectively under-sampled scans, with acceleration factors 3.6x, 4.6x, 7.0x, 9.2x, and 11.4x. For the TSE protocol, 2 scans were made in sequence, one fully sampled and one 12.1x prospectively under-sampled scan. The sampling schemes used came pre-installed on the scanner for CS acquisitions and are pseudorandom patterns with increasing sampling density toward the lower frequencies.

4.2. Acceleration method used for training and testing

Through-out our experiments, low frequencies near the origin in k-space were always fully sampled within an ellipse with half-axes set to 2% of the image axes. Outside this central region, data points were sampled from a Gaussian distribution with a full width at half maximum of 0.7, favoring low frequencies that contain more information about the general shape of the image content, while also creating incoherent noise due to randomness. We thereby adhere to the requirement of processing incoherent aliasing artifacts, as established in CS literature (Lustig et al., 2007).

To reduce the problem complexity, our model was initially developed for use on synthetic data, where coil images had been combined to a single image before discarding measurements. This method was described in Lønning et al. (2018), and amounts to setting $c = 1$ and $S_1 = I$. Integrating the use of PI with our model,

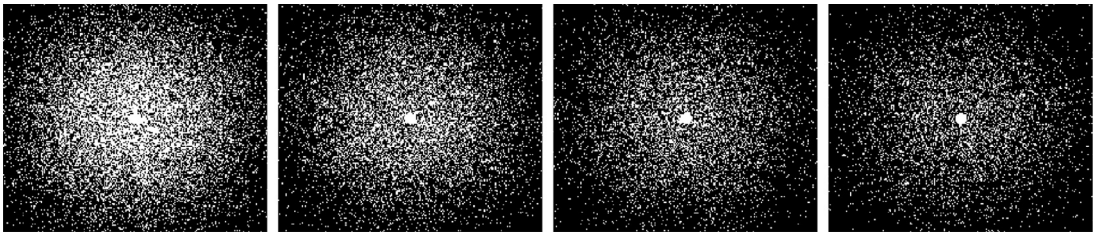


Fig. 4. Examples of 4x, 6x, 8x, and 10x accelerated cartesian sub-sampling masks, with sampled k-lines denoted in white.

in the way outlined through-out Section 3, enables the reconstruction of more highly accelerated images. As such, much of our results were obtained on models without the use of PI, but these are distinguishable for the reader by the acceleration factors used: We test on acceleration factors 2x, 3x, 4x, and 5x for the models not using PI, and on acceleration factors 4x, 6x, 8x, and 10x for the models with PI integrated. Furthermore, in all plots we assign green colors to RIMs without PI and blue colors to RIMs with PI.

In previous papers on accelerated MRI reconstruction with deep learning, models are trained on the same acceleration factor used for testing (Hammernik et al., 2018; Yang et al., 2017; Hyun et al., 2018; Schlemper et al., 2018). Whether this is an empirically known requirement or a working assumption is unclear. Either way, our hypothesis is that RIMs are capable of dealing with a range of acceleration factors during training. We verify this by comparing the performance of models trained on a single acceleration factor with models trained on a range of randomly sampled acceleration factors. In all other experiments, the models were trained on acceleration factors that were randomly sampled from a uniform distribution covering the acceleration factors used for testing. All models were trained using the ADAM optimizer (Kingma and Ba, 2014).

After selecting proper hyper-parameters for the training procedure, the model with the best average MSE on three sub-sampling patterns (P kept constant within each acceleration category) was selected for final evaluation. As mentioned in Section 4.1, the subjects used for model selection and in final test comparisons were kept separate, not only within the same type of data but also between the two brain scan protocols used. Final results acquired on the test sets were made on 3–6 sub-sampling patterns (these were also kept constant within each acceleration factor). Examples of acceleration patterns used for evaluation can be seen in Fig. 4.

4.3. Research focus

Since CS is widely used as part of scanner software today, we will use this as a benchmark for final performance evaluation. For this, we use the BART toolbox in Lustig et al. (2007), with the ℓ_1-norm of the wavelet transform as a regularizer. A regularization factor of 0.005 is used for the T_1-weighted brain data, whereas on the T_2^*-weighted brain and T_2-weighted knee data we set $\lambda = 0.008$. For all data types we use 80 iterations, and all results for CS was generated in combination with PI (PICS).

We report the Structural Similarity (SSIM) (Wang et al., 2004), Normalized Root Mean Square Error (NRMSE) and Peak Signal-to-Noise Ratio (PSNR) reconstruction scores on the magnitude images as performance metrics. Since magnitudes are never negative, and because each subject in our data was normalized by the maximum magnitude value, we used a dynamic range of 1 for the SSIM. This lowers the score somewhat when comparing against the use of 2, which is the default dynamic range value in many SSIM implementations. Similarly, we use a peak signal of 1 for the PSNR.

These metrics are good performance indicators, but it is difficult to capture the features corresponding to high quality reconstructions as perceived by human researchers or clinicians through mathematical estimates. For this reason, we supplement our results by including a randomized test given to an experienced neuroradiologist. Considering that knee images are outside this person's area of expertise, we only include the two brain datasets in this experiment. For each of the two contrasts we included 15 samples, of which 5 were randomly selected from one of the three anatomical orientations, for each of the 5 following categories: 4x and 8x accelerated RIM reconstructions, 4x and 8x accelerated CS reconstructions, and ground truth images. The samples were randomized within their respective contrast category, and the test was set up to be double blinded. The radiologist was asked to assign each image with a score from a five level Likert-scale, in which the categories were 'Excellent', 'Very Good', 'Good', 'Fair', or 'Poor'. The former three and latter two categories were to be considered acceptable and unacceptable for clinical use, respectively.

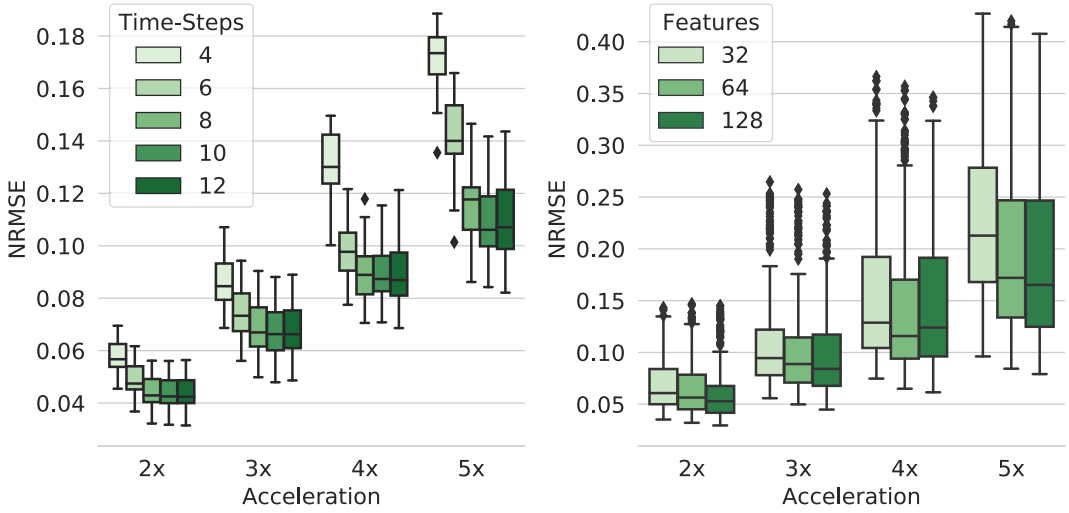
Beyond evaluating the best achievable quality obtained with our method, we also wish to illustrate the RIM's ability to generalize to unseen types of data, and its robustness against overfitting. We believe that the RIM's iterative structure and its internal and external states, all leveraged to include information about the image statistics and the forward model, help the RIM to generalize well. As such, we report scores for models evaluated on a different data type than that used for training. To illustrate this, we compare against the U-net. The U-net is a well-known architecture for the somewhat related task of image segmentation, but has also been suggested for reconstruction. We implemented nearly the same architecture as Hyun et al. (2018), however, a major difference is that we use the acceleration method described in Section 4.2, rather than a one-dimensional periodic subsampling scheme that remains constant throughout training. We also achieved better results using max-pooling instead of average-pooling when downscaling the image, and the post-processing step described in Hyun et al. (2018) was not used.

Note that, whereas RIMs and CS reconstruct the complex-valued signal, thereby yielding both phase and magnitude images, the U-net reconstructs the magnitude image only. It was also not implemented with PI. Therefore, we compare this method against the RIM without PI.

A common concern regarding data-driven approaches like artificial neural networks, is the question of how much data is needed to train a well-performing model. For this reason, we evaluate the performance impact of using a different number of subjects when training networks on the T_1-weighted brain dataset.

Before getting to the aforementioned experiments, we begin by optimizing for a few hyper-parameters. We try varying the number of time-steps t and the number of features f. We also compare two ways of weighting the loss function in Eq. (7); using equal weights, $w_\tau = 1$, and emphasizing the latter time-steps with $w_\tau = 10^{-\frac{t-\tau}{t-1}}$. Our hypothesis is that, since the final reconstruction is what matters, priority should be given to the latter time-steps. Another hyper-parameter is the size of the cropped patches used for training, and we compare patches of 30×30 pixels against 200×200 pixels. Patches were cropped in image space and corruptions were generated by applying the forward model in Eq. (1).

(a) Number of time-steps. (b) Number of features.



(c) Loss weights per time-step (d) Training patch size.

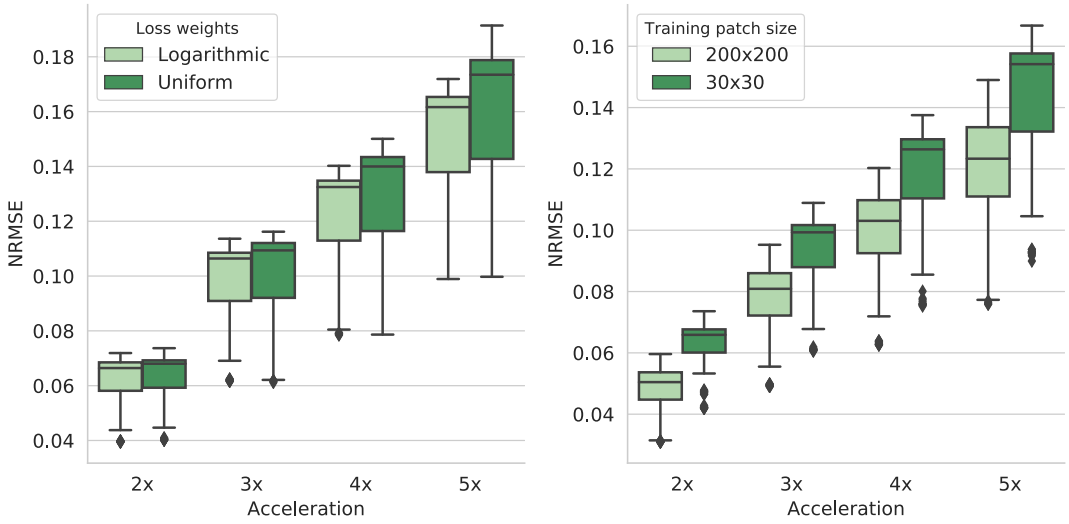


Fig. 5. Boxplots of NRMSE-values for the final time-step reconstructions on the datasets used for model selection and hyper-parameter tuning. The RIMs on the first and second row were trained and evaluated on 0.7 mm T_2^*-weighted brain images and 1.0 mm T_1-weighted brain images, respectively. Results are shown for each acceleration factor category. (a) The effect of varying the number of time-steps trained on. (b) The effect of varying the number of features trained on. (c) The effect of weighting the loss equally per time-step, or setting $w_\tau = 10^{-\frac{\tau}{T}}$, thereby favoring the final reconstructions. (d) The impact of training the network on smaller patches of 30×30 pixels, or larger patches of 200×200 pixels.

Hyper-parameters were selected based on the RIM's performance on the validation sets used for model selection, whereas all other experiments were conducted on the test sets.

As for the variance in measurement error σ in (5), we assume it to be somewhat similar between subjects. This allows us to set $\sigma = 1$ through-out all experiments, letting the RIM learn the scaling factor implicitly, without need for further fine-tuning. There is one exception to this setting, which we point out later.

5. Results

5.1. The number of time-steps

Fig. 5a shows the NRMSE-scores of the final reconstructions for RIMs trained on 0.7 mm T_2^*-weighted brain images, using a different number of time-steps: $t = 4, 6, 8, 10,$ and 12. These models were all trained with $f = 64$ features, on patches of size 30×30 and with weights in (7) set to $w_\tau = 1$.

As can be seen, the higher the acceleration factor, the more there is to be gained by increasing the number of time-steps. It should also be mentioned that training is somewhat unstable for 4 and 6 time-steps. At 8 time-steps, this is no longer and issue, and the improvement from adding more iterations becomes relatively insignificant, but is more expensive in terms of memory and computation time, hence we proceed with models trained on 8 time-steps going forward.

5.2. The number of features

Next, we ask what the effect is of varying the number of features. NRMSE-scores for models trained on $f = 32, 64,$ and 128 are shown in Fig. 5b. The same settings and dataset was used for these results as in Section 5.1. There is a substantial improvement when going from 32 to 64 features, and using 128 features also seems somewhat better. However, once more for the sake of saving mem-

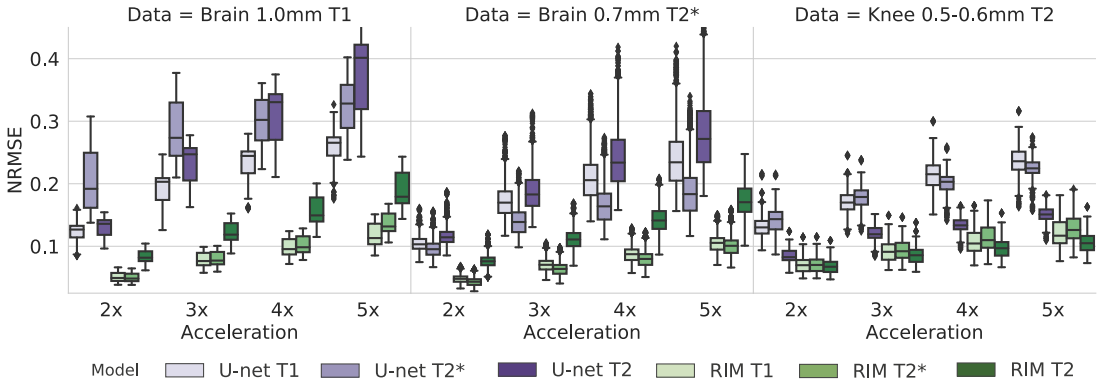


Fig. 6. Boxplots of NRMSE-values for the final time-step reconstructions of RIMs and U-nets (both without PI) trained on all three types of data: 1.0 mm T_1-weighted brains, 0.7 mm T_2^*-weighted brains, and 0.5-0.6 mm T_2-weighted knees. Hues indicate the model and the type of data trained on, and columns indicate the type of data evaluated on.

Table 1
Deep learning models used in experiments.

Model	Training data			
	Res. (mm)	Weighting	Anatomy	# Subjects
RIM T1	1.0	T_1	Brain	10
RIM T2*	0.7	T_2^*	Brain	10
RIM T2	0.5–0.6	T_2	Knee	17
U-net T1	1.0	T_1	Brain	10
U-net T2*	0.7	T_2^*	Brain	10
U-net T2	0.5–0.6	T_2	Knee	17

ory and computation time, we deem the performance at 64 features as acceptable and proceed with this setting.

5.3. Loss-function weights

Fig. 5c shows NRMSE-scores for training with equally weighted loss terms, versus emphasizing good performance on the latter time-steps by setting $w_\tau = 10^{-\frac{t-\tau}{t-1}}$. This model was trained and evaluated on 1.0 mm T_1-weighted brains. There is a slight improvement when the latter time-steps are emphasized, and so we continue with this setting.

5.4. Training patch size

Until now, we have used image patches of size 30×30 in the training set. This leads to larger mini-batches, but with a lower range of frequencies in k-space and less global information contained in each data point. We now try using larger patches of 200×200 pixels with smaller mini-batch sizes, resulting in approximately 180 instead of 6 samples per mini-batch. Our results, shown for 1.0 mm T_1-weighted brains in Fig. 5d, indicate that RIMs benefit from having access to more global information about the object to be reconstructed during training, and so we proceed training on patch sizes of 200×200 pixels.

Having selected hyper-parameters, we will henceforth report results on the test sets using models referred to and trained on data as listed in Table 1.

5.5. Comparing RIMs and U-nets

Fig. 6 shows NRMSE-scores for RIMs and U-nets. Results are plotted across acceleration factors for models trained and cross-evaluated on all three types of data. The third column of Fig. 6 is illustrated qualitatively in the second and third rows of Fig. 7, showing reconstructions of a 5x accelerated sample from the 0.5-0.6 mm T_2-weighted knee dataset.

A further comparison between RIMs and U-nets is given in Fig. 8a, showing the NRMSE-scores when trained and evaluated on 1.0 mm T_1-weighted brains, using two different acceleration schemes during training: For each model, one network is trained using only a single acceleration factor of 4x, and another network is trained on acceleration factors sampled from a uniform distribution $U(2, 5)$ covering the factors tested on.

5.6. The number of subjects required to train a RIM

Fig. 8b shows NRMSE-scores for RIMs trained on a different number of subjects. Training and testing was done on 1.0 mm T_1-weighted brain images.

5.7. PI-RIM

Until now, results were shown for models trained and evaluated on synthetic data using only a single coil. We now present results showing the benefit of including PI in our model. For the RIM T2*-model trained with PI, it was necessary to increase the noise parameter σ in (5) in order for reconstruction to work for all T_2-weighted knee images on acceleration factors 4x and 6x. We therefore set it to $\sigma = \sqrt{5}/2$ when using RIM T2* to reconstruct knees. For all other cases, it was kept at $\sigma = 1$. Fig. 9 shows median NRMSE-scores for the final time-steps on all three RIM models from Table 1, trained with and without integrated PI. To illustrate the improved convergence rate when using PI, median NRMSE-scores are plotted for the reconstruction of each recurrent time-step of the RIM T1-model in Fig. 10. Since the overall performance is quite different for the same acceleration factor, the model using PI is reconstructing 10x accelerated images, whereas an acceleration factor of 5x was used for the model without PI.

5.8. Comparing RIMs and CS

Fig. 11 shows SSIM- and PSNR-scores for the three RIM models from Table 1, along with CS under the same settings detailed in Section 4.2. The first and second columns of Fig. 11 are illustrated qualitatively in Figs. 12 and 13. The first shows reconstructions of a 8x accelerated sample from the 1.0 mm T_1-weighted brain dataset, whereas the second shows both magnitude and phase reconstructions of a 10x accelerated sample from the 0.7 mm T_2^*-weighted brain dataset. We report results from our double blinded Likert-scale reconstruction quality assessment test in Fig. 14. Additionally, we also included a reconstruction sample for CS when applied without PI to a 5x accelerated knee image in Fig. 7.

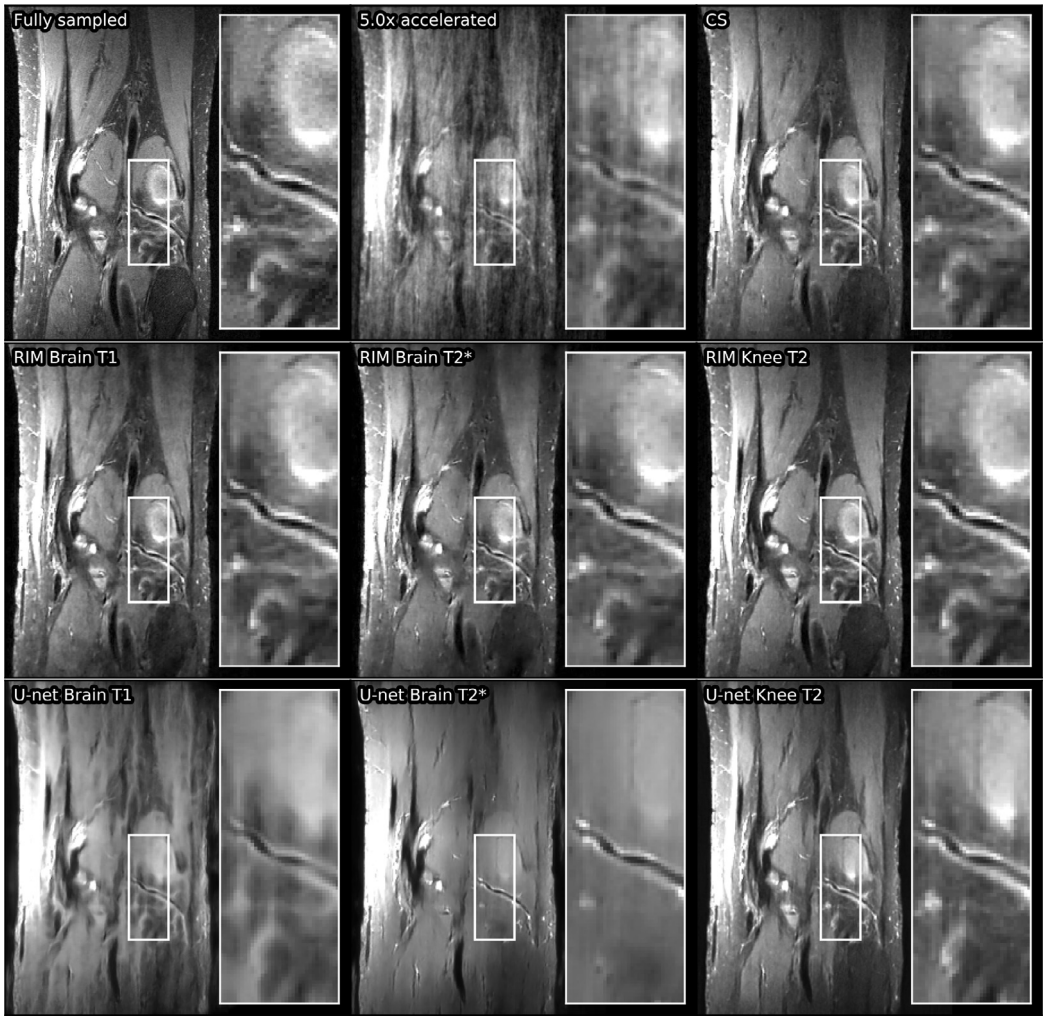
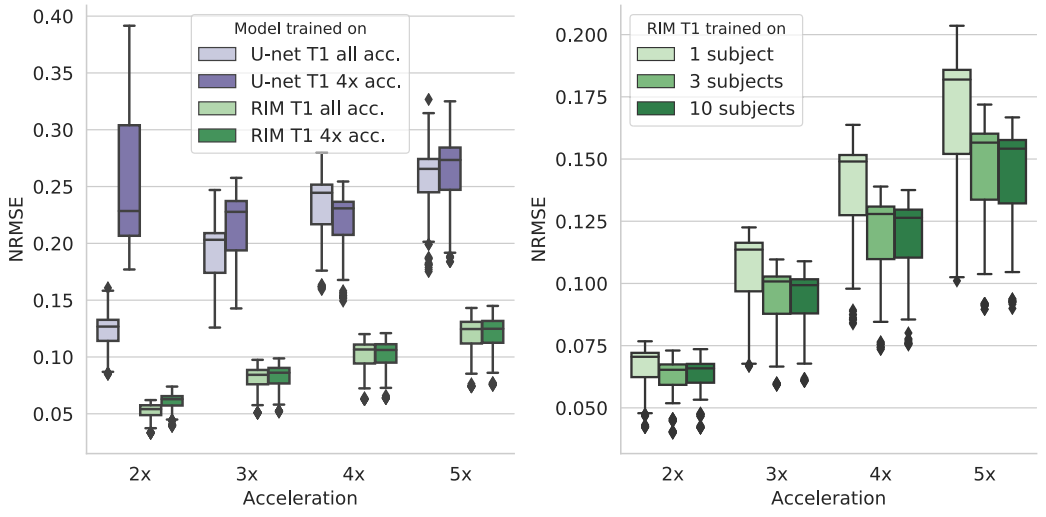


Fig. 7. Reconstructions of a 5x accelerated sample image from the test set of the T_2-weighted knee data. Top row shows, from left to right, the fully sampled ground truth, its 5x accelerated linear reconstruction, and a CS (without PI) reconstruction. Middle and bottom rows are RIM and U-net reconstructions (both without PI), respectively, where the models used across columns were trained on three different types of data, from left to right: 1.0 mm T_1-weighted brains, 0.7 mm T_2^*-weighted brains, and 0.5–0.6 mm T_2-weighted knees.



(a) Acceleration factor trained on. (b) Amount of data needed.

Fig. 8. Boxplots of NRMSE-scores on 1.0 mm T_1-weighted brains, showing (a) The effect of using a single acceleration factor during training, or sampling the acceleration factor from a uniform distribution covering the range 2x–5x, and (b) The impact of using a different number of subjects in the training set.

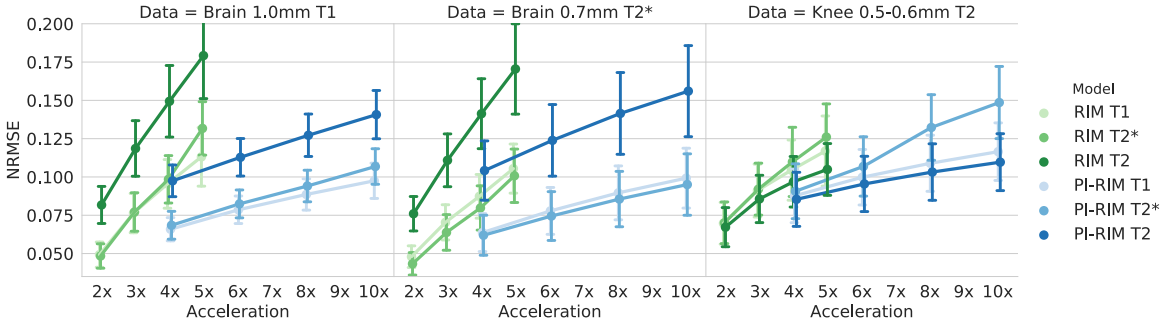


Fig. 9. Median point plot, with whiskers denoting the standard deviation, showing the NRMSE-scores of the final time-step reconstruction for RIMs with and without integrated PI.

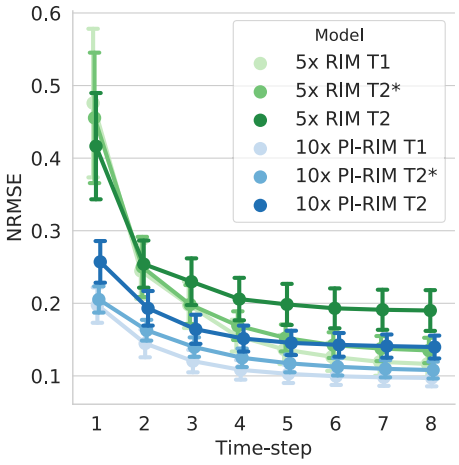


Fig. 10. Median point plot, with whiskers denoting the standard deviation, showing the NRMSE-scores of the individual time-steps for 10x accelerated RIM with PI reconstructions, and 5x accelerated RIM without PI reconstructions. The plot shows scores evaluated on T_1-weighted brains.

5.9. Prospective under-sampling

Shown in Fig. 15 are the RIM reconstructions generated from prospectively under-sampled raw data for the MPRAGE sequence described in Section 4.1. For this task, we use the RIM T1 model in Table 1. Reconstructions are shown for acceleration factors 3.6x, 4.6x, 7.0x, 9.2x, and 11.4x. In Fig. 16, 12.1x prospectively under-sampled TSE data is reconstructed using CS and all three RIM models from Table 1.

6. Discussion

We presented Recurrent Inference Machines (RIMs) for accelerated MRI reconstruction, with results demonstrating an ability to reconstruct high-quality images under varying measurement conditions. RIMs are robust against perturbations in the sub-sampling pattern used for data-acquisition, and can be exposed to a large range of acceleration factors in the same training session without loss of reconstruction quality. Further, we have shown that RIMs generalize well, not only in terms of the small amount of training data needed, but also with respect to unseen types of data,

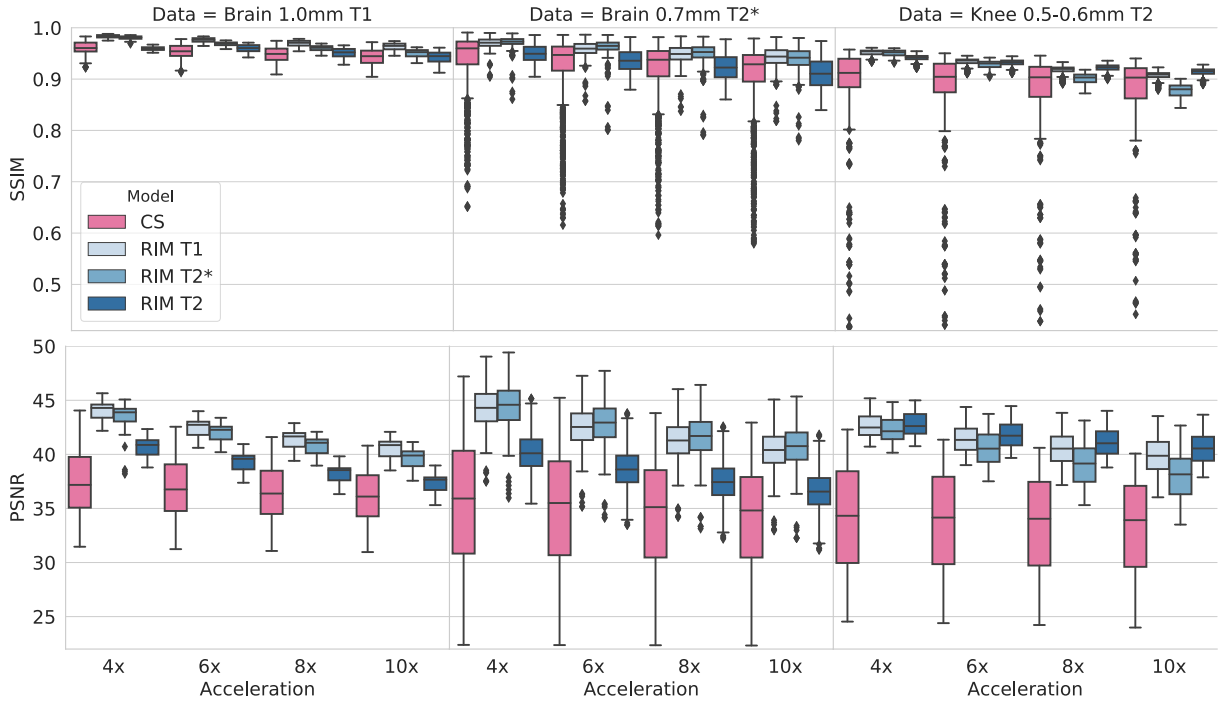


Fig. 11. Boxplots of SSIM- and PSNR-values for CS and the final time-step reconstructions of RIMs (both with PI) trained on all three types of data: 1.0 mm T_1-weighted brains, 0.7 mm T_2^*-weighted brains, and 0.5-0.6 mm T_2-weighted knees. Hues indicate the model and, in the case of the RIM, the type of data trained on, whereas columns indicate the type of data evaluated on.

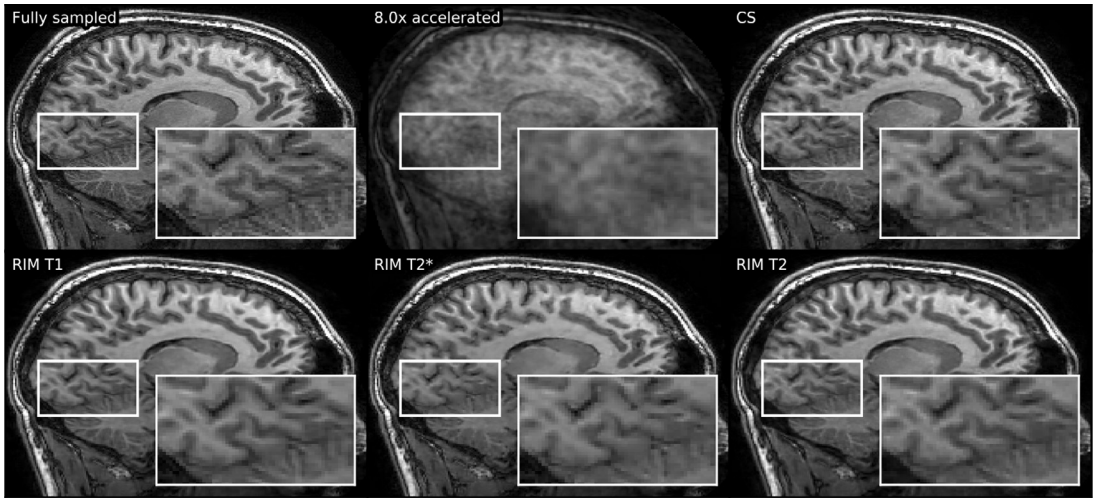


Fig. 12. Image shows reconstructions of a 8x accelerated 1.0 mm T_1-weighted brain seen in the sagittal plane. From left to right, the bottom row reconstructions were produced by RIMs trained on the 1.0 mm T_1-weighted brain dataset, the 0.7 mm T_2^*-weighted brain dataset, and the 0.5–0.6 mm T_2-weighted knee dataset. The top row shows the fully sampled reconstruction, the zero-filled reconstruction, used as the RIM's starting point, and finally a CS reconstruction on the right.

acquired at different field strengths and varying resolution levels, and even to other types of anatomical scans. This demonstrates the RIM's potential to perform well across measurement conditions that are present in clinical practice. A neuroradiologist rated the RIM reconstructions higher than those reconstructed with CS, the current standard for acceleration in the clinic. Finally, RIMs were shown to successfully reconstruct prospectively accelerated data, as shown for an MPRAGE sequence in Fig. 15, and even for a segmentation scheme, as shown for the TSE sequence in Fig. 16.

From the results, we conclude that RIMs have a low tendency to overfit, and a high capacity to generalize. We think this is owed to knowledge about the forward model in Eq. (1) being used to assess the state of the system during the reconstruction process, enabling the RIM to learn the iterative procedure necessary to reconstruct "any" given signal. As a result, this makes the RIM less dependent on learning the features that are specific to the signal itself. The use of internal states in a recurrent architecture also assists in this aim. Using the external state of the reconstruction, the RIM is implicitly learning from the data to evaluate the gradient of the log-prior distribution in Eq. (2), once again aiming to separate between the reconstruction procedure and the underlying data statistics. To a large extent this holds true, since reconstruction quality of images dissimilar to the training data approach those achieved on data types from the same distribution, as evidenced by Figs. 6 and 11.

6.1. Dataset size

The RIM's robustness against overfitting has the benefit of requiring very little data to train a good model. As Fig. 8b shows, the performance has nearly saturated when using three subjects instead of one in the training set. Related work on variational networks trained on data of 20 subjects (Hammernik et al., 2018) also showed that for image reconstruction purposes a limited training dataset should suffice.

6.2. Inference time

An advantage of deep learning methods lies in the short inference times. RIMs take 208 ms to process a 230×230-image with 32 coils. Earlier work on a Variational Network reported a comparable inference time of 193ms for a 320×256-image with 15 coils (Hammernik et al., 2018). The time could be further reduced by lowering the number of features or time-steps, which our results indicate could come at an acceptable cost in quality. Judging from the convergence rates shown in Fig. 10, reducing the number of time-steps seems like a good option for the PI-RIM model in particular, where the inference time can be reduced to 130 ms using 5 time-steps. Alternatively, using 32 features and 6 time-steps would lead to an inference time of 87 ms on a single GPU per slice, while still maintaining an acceptable quality depending on the application. As for CS, we have not measured the inference time on the GPU. However, with a CPU time of 35.6 s, it is substantially slower than the RIM, which spends 3.48 s processing an image on the CPU. Use cases requiring fast access to images may greatly benefit from the fast inference. These include an optimized clinical workflow, and real-time imaging applications such as radiotherapy planning.

6.3. Comparison to the U-net

Considering the U-net (Hyun et al., 2018), which is ubiquitous in the domain of image segmentation (Ronneberger et al., 2015; Long et al., 2014), we see in Fig. 7 that it cannot accurately reconstruct the data, even within the same data category used for training. Particularly, it suffers from issues related to overfitting. The model trained on T_1-weighted brain images generates structural artifacts resembling gyri and sulci, whereas the finer boundaries found in the high resolution T_2^*-weighted brain images causes the model to average out all but the sharpest of boundaries. We see from Fig. 8a that the U-net also overfits with respect to the acceleration factor. The model produces the same or worse quality if it was never exposed to acceleration factors close to 2x during training, even though this should be the easiest acceleration category to reconstruct. In contrast, RIMs trained on other data do lead to good reconstruction results and the RIM is also largely unaffected by the acceleration factor used during training. These comparisons illustrate the previously mentioned benefits of the RIM's architecture. What the RIM can extract from the log-likelihood gradient and its external and internal states, the U-net must make up for using its parameters. Indeed, at 1,328,833 and 94,336 parameters, the U-net has many more parameters than the RIM as well. This may be another contributing cause for the discrepancy in overfitting between the two models.

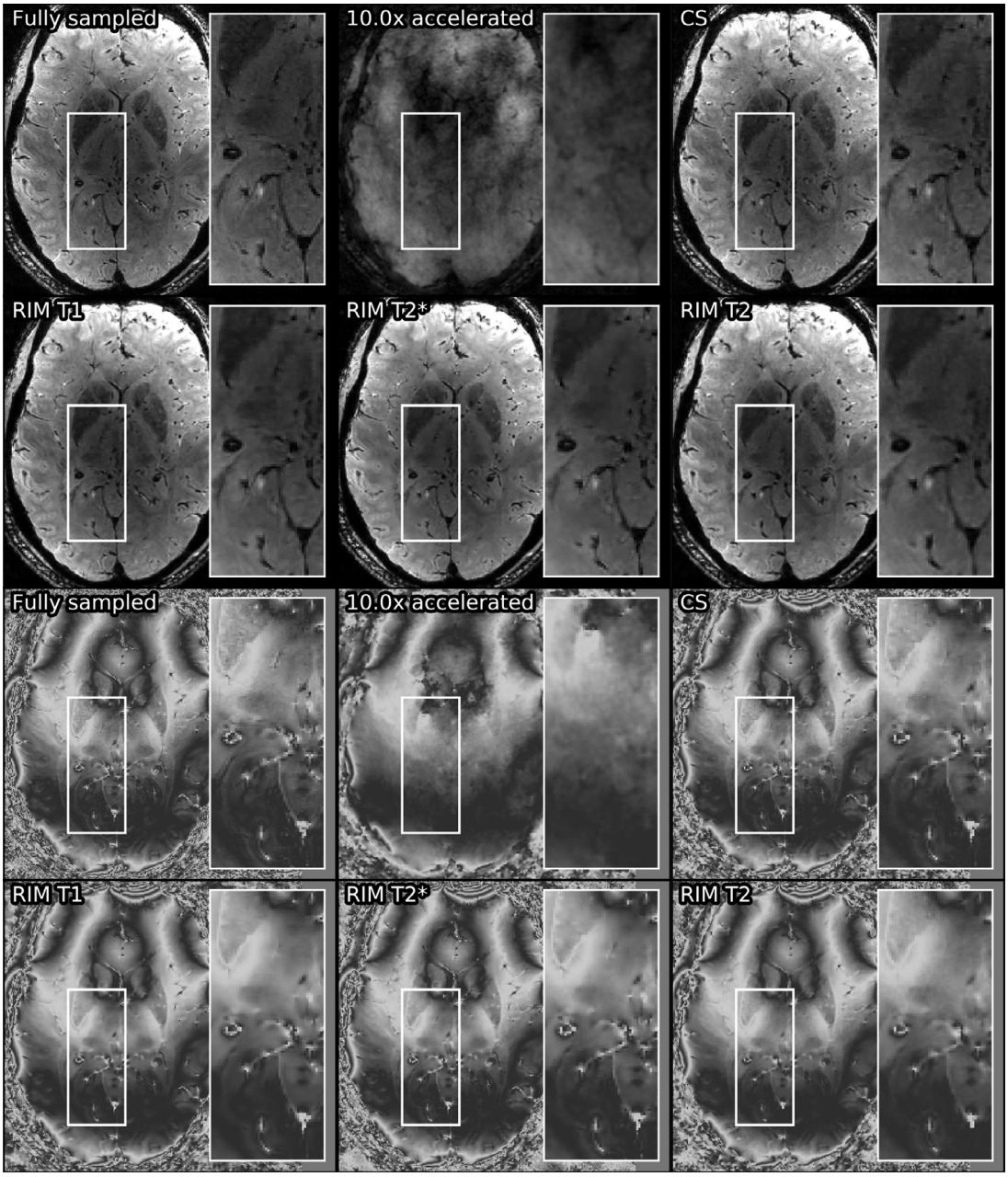


Fig. 13. Image shows reconstructions of a 10x accelerated 0.7 mm T_2^*-weighted brain seen in the transverse plane. Top two rows show the reconstructed magnitude, whereas the bottom two rows show the phase of the reconstructions (we use a cyclic color map from Kovesi, 2015). From left to right, the second and fourth row reconstructions were produced by RIMs trained on the 1.0 mm T_1-weighted brain dataset, the 0.7 mm T_2^*-weighted brain dataset, and the 0.5–0.6 mm T_2-weighted knee dataset. The first and second row shows the fully sampled reconstruction, the zero-filled reconstruction, used as the RIM's starting point, and finally a CS reconstruction on the right.

6.4. Comparison to compressed sensing

Compared to Compressed Sensing (CS), the reconstruction quality is more stable for the RIM over multiple realizations of random undersampling masks. Images produced by CS vary more with respect to the input image, as evidenced by multiple adverse outliers in Fig. 11. This plot also indicates a worse performance of CS on average. Indeed, the RIM was the preferred model over CS in a double blinded quality assessment test, as seen in Fig. 14. The comparison to CS also highlights some limitations of the RIM and the metrics used in our study.

6.5. Limitations

When training on lower SNR data, the RIM is challenged to avoid blurring during inference time. This was observed for the model trained on T_2^*-data included in this study, as can be seen in Figs. 12, 13 and 16, where the qualitative results appear to be perceptually in favor of CS. This modality is characterized by limited magnitude contrast and a strong phase evolution, because of the gradient echo readout with long echo times. The high spatial resolution of 0.7 mm and low flip angle of 12° made this the model having the lowest SNR included in this study. In one occasion this model even experienced convergence difficulties; when inferring 4x, and to a lesser extent 6x, accelerated knee data. For this case, it

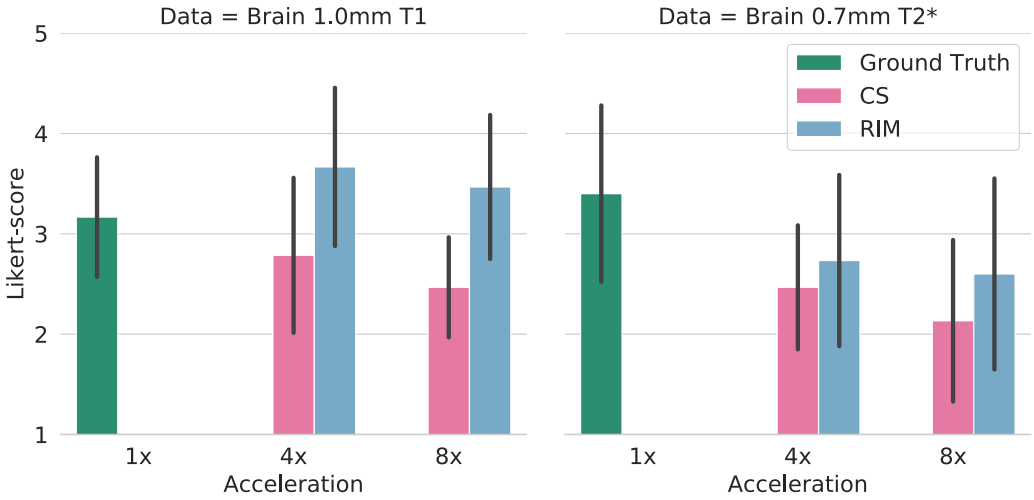


Fig. 14. The bars show the average Likert-score, as assigned by a neuro-radiologist in our quality assessment test. Black lines indicate the standard deviation.

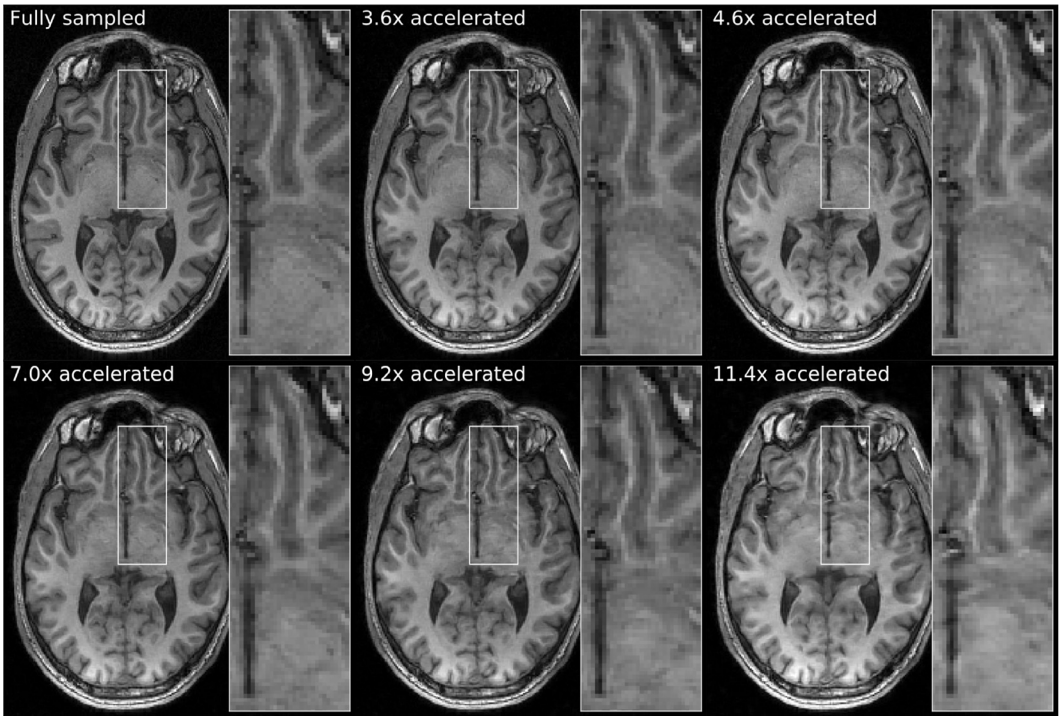


Fig. 15. RIM reconstructions of prospectively under-sampled raw T_1-weighted data, for differing acceleration factors, using the T_1-model. The highest acceleration factor seen during training was 10x.

was necessary to increase the noise value σ in Eq. (5), which had otherwise been set to 1 through-out all experiments. The extent to which σ was increased made very little difference to the final reconstruction (tested in a range of 1.1–3.0, data not shown). Thus, even though there is a threshold where the model breaks down, there seems to be a large range of values for this hyper-parameter in which performance is robust.

6.6. Future work

Future work should study the ability to reconstruct pathologies possibly unseen during training. We consider the substantial performance invariance across the three datasets to be indicative of a model that can reliably reconstruct statistical outliers, from naturally occurring structural differences in anatomy between individuals, to patient-specific pathologies. This must be properly verified in future clinical studies. Initial work shows the ability of neural networks to reconstruct white matter lesions (Tezcan et al., 2017).

An important extension of our work would be to adapt the forward model in Eq. (1) to include the correlation of noise between the receiver coils. The data acquired from separate noise reference scans usually made as part of MR acquisition pipelines would then be properly exploited in the RIM reconstruction. This should lead to higher quality images, and perhaps even resolve such issues as having to fine-tune σ in exceptional cases, as described in the Section 6.5.

Future work should further look into learning different readout strategies on heterogeneous data. Our current results are restricted to Euclidean readout strategies that sample each line in k-space individually. The AUTOMAP approach (Zhu et al., 2018) shows that good performance on various sampling patterns can be obtained.

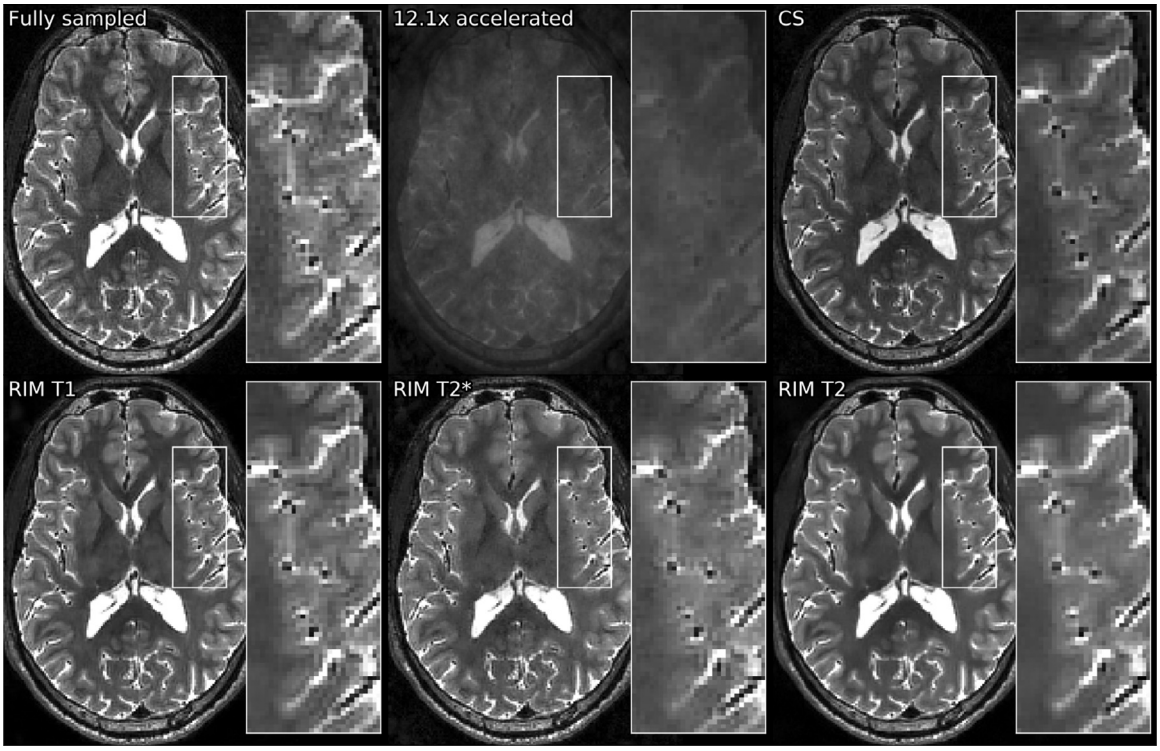


Fig. 16. CS and RIM reconstructions of 12.1x prospectively under-sampled raw T_2-weighted brain data without a fully sampled center. The highest acceleration factor seen during training was 10x.

Extending this work to become applicable in a wide variety of settings is an important next milestone to reach.

Finally, certain caution is warranted in interpreting the metrics used in this study. Especially in the case of noisy data, NRMSE and PSNR metrics favor blurred images over their noisy counterparts. The SSIM was designed aiming for more robustness against these effects. Still, it can be questioned how different types of artifacts should be ranked relative to one another, and what kind of traits constitute a good quality image.

In conclusion, our work shows that through RIMs, deep learning based reconstruction of heterogeneous MRI data is feasible, bringing such an approach a step closer toward use in clinical practice.

Conflict of Interest

None.

Acknowledgments

We kindly thank dr. F.M. Vos and dr. G.A.M. Arkesteijn for their support in data acquisition. Kai Lønning and Max Welling are supported by the Canadian Institute for Advanced Research (CIFAR). Patrick Putzky is supported by the Netherlands Organisation for Scienetific Research (NWO) and the Netherlands Institute for Radio Astronomy (ASTRON) through the big bang, big data grant.

References

Andrychowicz, M., Denil, M., Colmenarejo, S. G., Hoffman, M. W., Pfau, D., Schaul, T., de Freitas, N., 2016. Learning to learn by gradient descent by gradient descent. arXiv:1606.04474.

Chen, Y., Yu, W., Pock, T., 2015. On learning optimized reaction diffusion processes for effective image restoration. In: Computer Vision and Pattern Recognition (CVPR) IEEE Conference, pp. 5261–5269. doi:10.1109/CVPR.2015.7299163.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN EncoderDecoder for statistical machine translation. In: Empirical Methods in Natural Language Processing (EMNLP) conference. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 1724–1734. doi:10.3115/v1/D14-1179. 1406.1078.

Griswold, M.A., Jakob, P.M., Heidemann, R.M., Nittka, M., Jellus, V., Wang, J., Kiefer, B., Haase, A., 2002. Generalized autocalibrating partially parallel acquisitions (GRAPPA). Magn. Reson. Med. 47 (6), 1202–1210. doi:10.1002/mrm.10171.

Haacke, E., Brown, R., Thompson, M., Venkatesan, R., 1999. Magnetic Resonance Imaging: Physical Principles and Sequence Design. Wiley Online Library.

Hammernik, K., Klatzer, T., Kobler, E., Recht, M.P., Sodickson, D.K., Pock, T., Knoll, F., 2018. Learning a variational network for reconstruction of accelerated MRI data. Magn. Reson. Med. 79 (6), 3055–3071. doi:10.1002/mrm.26977.

Hyun, C.M., Kim, H.P., Lee, S.M., Lee, S., Seo, J.K., 2018. Deep learning for undersampled MRI reconstruction. Phys. Med. Biol. 63. doi:10.1088/1361-6560/aac71a.

Jin, K.H., McCann, M.T., Froustey, E., Unser, M., 2017. Deep convolutional neural network for inverse problems in imaging. IEEE Trans. Image Process. 26 (9), 4509–4522. doi:10.1109/TIP.2017.2713099, 1611.03679.

Kingma, D. P., Ba, J., 2014. Adam: a method for stochastic optimization. arXiv:1412.6980.

Kovesi, P., 2015. Good colour maps: how to design them. arXiv:1509.03700.

Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., van Ginneken, B., Snchez, C.I., 2017. A survey on deep learning in medical image analysis. Med Image Anal 42, 60–88. doi:10.1016/j.media.2017.07.005.

Long, J., Shelhamer, E., Darrell, T., 2014. Fully convolutional networks for semantic segmentation. arXiv:1411.4038.

Lønning, K., Putzky, P., Caan, M.W.A., Welling, M., 2018. Recurrent inference machines for accelerated MRI reconstruction. In: Medical Imaging with Deep Learning (MIDL 2018) Conference, Amsterdam, The Netherlands.

Lustig, M., Donoho, D., Pauly, J.M., 2007. Sparse MRI: the application of compressed sensing for rapid MR imaging. Magn. Reson. Med. 58 (6), 1182–1195. doi:10.1002/mrm.21391.

Pruessmann, K.P., Weiger, M., Scheidegger, M.B., Boesiger, P., 1999. SENSE: Sensitivity encoding for fast MRI. Magn. Reson. Med. 42 (5), 952–962. doi:10.1002/(SICI)1522-2594(199911)42:5<952::AID-MRM16>3.0.CO;2-S.

Putzky, P., Welling, M., 2017. Recurrent inference machines for solving inverse problems. arXiv:1706.04008.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: convolutional networks for biomedical image segmentation. arXiv:1505.04597.

Sawyer, A.M., Lustig, M., Alley, M., Uecker, P., Virtue, P., Lai, P., Vasanawala, A., Healthcare, G., 2013. Creation of fully sampled MR data repository for compressed sensing of the knee. In: Proceedings of the 22nd Annual Meeting for Section for Magnetic Resonance Technologists. Salt Lake City, Utah, USA.

Schlemper, J., Caballero, J., Hajnal, J.V., Price, A.N., Rueckert, D., 2018. A deep cascade of convolutional neural networks for dynamic MR image reconstruction. IEEE Trans Med Imaging 37 (2), 491–503. doi:10.1109/TMI.2017.2760978.

Tezcan, K. C., Baumgartner, C. F., Konukoglu, E., 2017. MR image reconstruction using deep density priors. arXiv:1711.11386.

Uecker, M., Lai, P., Murphy, M.J., Virtue, P., Elad, M., Pauly, J.M., Vasanawala, S.S., Lustig, M., 2014. ESPIRiT an eigenvalue approach to autocalibrating parallel MRI: where SENSE meets GRAPPA. Magn. Reson. Med. 1001, 990–1001. doi:10.1002/mrm.24751.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13 (4), 600–612. doi:10.1109/TIP.2003.819861.

Yang, Y., Sun, J., Li, H., Xu, Z., 2017. Admm-net: a deep learning approach for compressive sensing mri. arXiv:1705.06869.

Zhu, B., Liu, J.Z., Cauley, S.F., Rosen, B.R., Rosen, M.S., 2018. Image reconstruction by domain-transform manifold learning. Nature 555 (7697), 487–492. doi:10.1038/nature25988.