Convolutional Recurrent Neural Networks for Dynamic MR Image Reconstruction

Chen Qin*†, Jo Schlemper*, Jose Caballero, Anthony N. Price, Joseph V. Hajnal and Daniel Rueckert *Fellow, IEEE*

***Abstract*—Accelerating the data acquisition of dynamic magnetic resonance imaging (MRI) leads to a challenging ill-posed inverse problem, which has received great interest from both the signal processing and machine learning communities over the last decades. The key ingredient to the problem is how to exploit the temporal correlations of the MR sequence to resolve aliasing artefacts. Traditionally, such observation led to a formulation of a non-convex optimisation problem, which was solved using iterative algorithms. Recently, however, deep learning based-approaches have gained significant popularity due to their ability to solve general inverse problems. In this work, we propose a unique, novel convolutional recurrent neural network (CRNN) architecture which reconstructs high quality cardiac MR images from highly undersampled k-space data by jointly exploiting the dependencies of the temporal sequences as well as the iterative nature of the traditional optimisation algorithms. In particular, the proposed architecture embeds the structure of the traditional iterative algorithms, efficiently modelling the recurrence of the iterative reconstruction stages by using recurrent hidden connections over such iterations. In addition, spatio-temporal dependencies are simultaneously learnt by exploiting bidirectional recurrent hidden connections across time sequences. The proposed method is able to learn both the temporal dependency and the iterative reconstruction process effectively with only a very small number of parameters, while outperforming current MR reconstruction methods in terms of computational complexity, reconstruction accuracy and speed.**

***Index Terms*—Recurrent neural network, convolutional neural network, dynamic magnetic resonance imaging, cardiac image reconstruction**

I. Introduction

MAGNETIC Resonance Imaging (MRI) is a non-invasive imaging technique which offers excellent spatial resolution and soft tissue contrast and is widely used for clinical diagnosis and research. Dynamic MRI attempts to reveal both spatial and temporal profiles of the underlying anatomy, which has a variety of applications such as cardiovascular imaging and perfusion imaging. However, the acquisition speed is fundamentally limited due to both hardware and physiological constraints as well as the requirement to satisfy the Nyquist sampling rate. Long acquisition times are not only a burden for patients but also make MRI susceptible to motion artefacts.

†Corresponding author: Chen Qin. (Email address: c.qin15@imperial.ac.uk)
*These authors contributed equally to this work.
C. Qin, J. Schlemper, J. Caballero and D. Rueckert are with the Biomedical Image Analysis Group, Department of Computing, Imperial College London, SW7 2AZ London, UK.
J. V. Hajnal, and A. N. Price are with with the Division of Imaging Sciences and Biomedical Engineering Department, King's College London, St. Thomas' Hospital, SE1 7EH London, U.K.

In order to accelerate MRI acquisition, most approaches consider undersampling the data in k-space (frequency domain). Due to the violation of the Nyquist sampling theorem, undersampling introduces aliasing artefacts in the image domain. Images can be subsequently reconstructed by solving an optimisation problem that regularises the solution with assumptions on the underlying data, such as smoothness, sparsity or, for the case of dynamic imaging, spatio-temporal redundancy. Past literature has shown that exploiting spatio-temporal redundancy can greatly improve image reconstruction quality compared to compressed sensing (CS) based single frame reconstruction methods [1], [2]. However, the limitations of these optimisation based approaches are the following: firstly, it is non-trivial to propose an optimisation function without introducing significant bias on the considered data. In addition, the manual adjustments of hyperparameters are nontrivial. Secondly, these optimisation problems often involve highly nonlinear, non-convex terms. As a consequence, the majority of approaches resort on iterative algorithms to reconstruct the images, in which often neither attaining the global minimum nor convergence to a solution is guaranteed. Furthermore, the reconstruction speeds of these methods are often slow. Proposing a robust iterative algorithm is still an active area of research.

In comparison, deep learning methods are gaining popularity for their accuracy and efficiency. Unlike traditional approaches, the prior information and regularisation are learnt implicitly from data, allowing the reconstruction to look more natural. However, the limitation is that so far only a handful of approaches exist [3], [4] for dynamic reconstruction. Hence, the applicability of deep learning models to this problem is yet to be fully explored. In particular, a core question is how to optimally exploit spatio-temporal redundancy. In addition, most deep learning methods do not exploit domain-specific knowledge, which potentially enables the networks to efficiently learn data representation by regulating the mechanics of network layers, hence boosting their performances.

In this work, we propose a novel convolutional recurrent neural network (CRNN) method to reconstruct high quality dynamic MR image sequences from undersampled data, termed *CRNN-MRI*, which aims to tackle the aforementioned problems of both traditional and deep learning methods. Firstly, we formulate a general optimisation problem for solving accelerated dynamic MRI based on variable splitting and alternate minimisation. We then show how this algorithm can be seen as a network architecture. In particular, the proposed method consists of a CRNN block which acts as the

proximal operator and a data consistency layer corresponding to the classical data fidelity term. In addition, the CRNN block employs recurrent connections across each iteration step, allowing reconstruction information to be shared across the multiple iterations of the process. Secondly, we incorporate bidirectional convolutional recurrent units evolving over time to exploit the temporal dependency of the dynamic sequences and effectively propagate the contextual information across time frames of the input. As a consequence, the unique CRNN architecture jointly learns representations in a recurrent fashion evolving over both *time sequences* as well as *iterations* of the reconstruction process, effectively combining the benefits of traditional iterative methods and deep learning.

To the best of our knowledge, this is the first work applying RNNs for dynamic MRI reconstruction. The contributions of this work are the following: Firstly, we view the optimisation problem of dynamic data as a recurrent network and describe a novel CRNN architecture which simultaneously incorporates the recurrent relationship of data over time and iterations. Secondly, we demonstrate that the proposed method shows promising results and improves upon the current state-of-the-art dynamic MR reconstruction methods both in reconstruction accuracy and speed. Finally, we compare our architecture to 3D CNN which does not impose the recurrent structure. We show that the proposed method outperforms the CNN at different undersampling rates and speed, while requiring significantly fewer parameters.

II. Related Work

One of the main challenges associated with recovering an uncorrupted image is that both the undersampling strategy and a-priori knowledge of appropriate properties of the image need to be taken into account. Methods like k-t BLAST and k-t SENSE [5] take advantage of a-priori information about the x-f support obtained from the training data set in order to prune a reconstruction to optimally reduce aliasing. An alternative popular approach is to exploit temporal redundancy to unravel from the aliasing by using CS approaches [1], [6] or CS combined with low-rank approaches [2], [7]. The class of methods which employ CS to the MRI reconstruction is termed as CS-MRI [8]. They assume that the image to be reconstructed has a sparse representation in a certain transform domain, and they need to balance sparsity in the transform domain against consistency with the acquired undersampled k-space data. For instance, an example of successful methods enforcing sparsity in x-f domain is k-t FOCUSS [1]. A low rank and sparse reconstruction scheme (k-t SLR) [2] introduces non-convex spectral norms and uses a spatio-temporal total variation norm in recovering the dynamic signal matrix. Dictionary learning approaches were also proposed to train an over-complete basis of atoms to optimally sparsify spatio-temporal data [6]. These methods offer great potential for accelerated imaging, however, they often impose strong assumptions on the underlying data, requiring nontrivial manual adjustments of hyperparameters depending on the application. In addition, it has been observed that these methods tend to result in blocky [9] and unnatural reconstructions, and their reconstruction speed is often slow. Furthermore, these methods are not able to exploit the prior knowledge that can be learnt from the vast number of MRI exams routinely performed, which should be helpful to further guide the reconstruction process.

Recently, deep learning-based MR reconstruction has gained popularity due to its promising results for solving inverse and compressed sensing problems. In particular, two paradigms have emerged: the first class of approaches proposes to use convolutional neural networks (CNNs) to learn an end-to-end mapping, where architectures such as SRCNN [10] or U-net [11] are often chosen for MR image reconstruction [12], [13], [14], [15]. The second class of approaches attempts to make each stage of iterative optimisation learnable by unrolling the end-to-end pipeline into a deep network [9], [16], [17], [18], [19]. For instance, Hammernik et al. [9] introduced a trainable formulation for accelerated parallel imaging (PI) based MRI reconstruction termed variational network, which embedded a CS concept within a deep learning approach. ADMM-Net [17] was proposed by reformulating an alternating direction method of multipliers (ADMM) algorithm to a deep network, where each stage of the architecture corresponds to an iteration in the ADMM algorithm. More recently, Schlemper et al. [18] proposed a cascade network which simulated the iterative reconstruction of dictionary learning-based methods and were later extended for dynamic MR reconstructions [3]. Most approaches so far have focused on 2D images, whereas only a few approaches exist for dynamic MR reconstruction [3], [4]. While they show promising results, the optimal architecture, training scheme and configuration spaces are yet to be fully explored.

More recently, several ideas were proposed, which integrate deep neural networks with model-based optimisation methods to solve inverse problems [20], [21]. In contrast to these papers, which proposed to deal with a fidelity term and a regularisation term separately, we integrate the two terms in a single deep network, so that the network can be trained end-to-end. In addition, as we will show, we integrate a hidden connection over the optimisation iteration to enable the information used for the reconstruction at each iteration to be shared across all stages of the reconstruction process, aiming for an iterative algorithm that can fully benefit from information extracted at all processing stages. As to the nature of the proposed RNN units, previous work involving RNNs only updated the hidden state of the recurrent connection with a fixed input [22], [23], [24], while the proposed architecture progressively updates the input as the optimisation iteration increases. In addition, previous work only modelled the recurrence of iteration *or* time [25] exclusively, whereas the proposed method jointly exploits both dimensions, yielding a unique architecture suitable for the dynamic reconstruction problem.

III. Convolutional Recurrent Neural Network for MRI reconstruction

A. Problem Formulation

Let $\mathbf{x} \in \mathbb{C}^D$ denote a sequence of complex-valued MR images to be reconstructed, represented as a vector with $D = D_x D_y T$, and let $\mathbf{y} \in \mathbb{C}^M$ $(M << D)$ represent the

undersampled k-space measurements, where D_x and D_y are width and height of the frame respectively and T stands for the number of frames. Our problem is to reconstruct $\mathbf{x}$ from $\mathbf{y}$, which is commonly formulated as an unconstrained optimisation problem of the form:

$$\underset{\mathbf{x}}{\text{argmin}} \quad \mathcal{R}(\mathbf{x}) + \lambda \|\mathbf{y} - \mathbf{F}_u \mathbf{x}\|_2^2 \tag{1}$$

Here $\mathbf{F}_u$ is an undersampling Fourier encoding matrix, $\mathcal{R}$ expresses regularisation terms on $\mathbf{x}$ and λ allows the adjustment of data fidelity based on the noise level of the acquired measurements $\mathbf{y}$. For CS and low-rank based approaches, the regularisation terms $\mathcal{R}$ often employed are ℓ_0 or ℓ_1 norms in the sparsifying domain of $\mathbf{x}$ as well as the rank or nuclear norm of $\mathbf{x}$ respectively. In general, Eq. 1 is a non-convex function and hence, the variable splitting technique is usually adopted to decouple the fidelity term and the regularisation term. By introducing an auxiliary variable $\mathbf{z}$ that is constrained to be equal to $\mathbf{x}$, Eq. 1 can be reformulated to minimize the following cost function via the penalty method:

$$\underset{\mathbf{x},\mathbf{z}}{\text{argmin}}\ \mathcal{R}(\mathbf{z}) + \lambda \|\mathbf{y} - \mathbf{F}_u \mathbf{x}\|_2^2 + \mu \|\mathbf{x} - \mathbf{z}\|_2^2 \tag{2}$$

where μ is a penalty parameter. By applying alternate minimisation over $\mathbf{x}$ and $\mathbf{z}$, Eq. 2 can be solved via the following iterative procedures:

$$\mathbf{z}^{(i+1)} = \underset{\mathbf{z}}{\text{argmin}}\ \mathcal{R}(\mathbf{z}) + \mu \|\mathbf{x}^{(i)} - \mathbf{z}\|_2^2 \tag{3a}$$

$$\mathbf{x}^{(i+1)} = \underset{\mathbf{x}}{\text{argmin}}\ \lambda \|\mathbf{y} - \mathbf{F}_u \mathbf{x}\|_2^2 + \mu \|\mathbf{x} - \mathbf{z}^{(i+1)}\|_2^2 \tag{3b}$$

where $\mathbf{x}^{(0)} = \mathbf{x}_u = \mathbf{F}_u^H y$ is the zero-filled reconstruction taken as an initialisation and $\mathbf{z}$ can be seen as an intermediate state of the optimisation process. For MRI reconstruction, Eq. 3b is often regarded as a *data consistency* (DC) step where we can obtain the following closed-form solution [18]:

$$\begin{aligned} &\mathbf{x}^{(i+1)} = \text{DC}(\mathbf{z}^{(i)}; \mathbf{y}, \lambda_0, \Omega) = \mathbf{F}^H \mathbf{\Lambda} \mathbf{F} \mathbf{z}^{(i)} + \tfrac{\lambda_0}{1+\lambda_0} \mathbf{F}_u^H \mathbf{y}, \\ &\mathbf{\Lambda}_{kk} = \begin{cases} 1 & \text{if } k \notin \Omega \\ \frac{1}{1+\lambda_0} & \text{if } k \in \Omega \end{cases} \end{aligned} \tag{4}$$

in which $\mathbf{F}$ is the full Fourier encoding matrix (a discrete Fourier transform in this case), $\lambda_0 = \lambda/\mu$ is a ratio of regularization parameters from Eq. 4, Ω is an index set of the acquired k-space samples and Λ is a diagonal matrix. Please refer to [18] for more details of formulating Eq. 4 as a data consistency layer in a neural network. Eq. 3a is the proximal operator of the prior $\mathcal{R}$, and instead of explicitly determining the form of the regularisation term, we propose to directly learn the proximal operator by using a convolutional recurrent neural network (CRNN).

Previous deep learning approaches such as Deep-ADMM net [17] and method proposed by Schlemper et al. [18] unroll the traditional optimisation algorithm. Hence, their models learn a sequence of transition $\mathbf{x}^{(0)} \rightarrow \mathbf{z}^{(1)} \rightarrow \mathbf{x}^{(1)} \rightarrow \cdots \rightarrow \mathbf{z}^{(N)} \rightarrow \mathbf{x}^{(N)}$ to reconstruct the image, where each state transition at stage (i) is an operation such as convolutions independently parameterised by $\boldsymbol{\theta}$, nonlinearities or a data consistency step. However, since the network implicitly learns some form of proximal operator at each iteration, it may be redundant to individually parameterise each step. In our formulation, we model each optimisation stage (i) as a learnt, *recurrent*, forward encoding step $f_i(\mathbf{x}^{(i-1)}, \mathbf{z}^{(i-1)}; \boldsymbol{\theta}, \mathbf{y}, \lambda, \Omega)$. The difference is that now we use one model which performs proximal operator, however, it also allows itself to propagate information across iteration, making it adaptable for the changes across the optimisation steps. The detail will be discussed in the following section. The different strategies are illustrated in Fig 1.

B. CRNN for MRI reconstruction

RNN is a class of neural networks that makes use of sequential information to process sequences of inputs. They maintain an internal state of the network acting as a "memory", which allows RNNs to naturally lend themselves to the processing of sequential data. Inspired by iterative optimisation schemes of Eq. 3, we propose a novel convolutional RNN (CRNN) network. In the most general scope, our neural encoding model is defined as follows,

$$\mathbf{x}_{rec} = f_N(f_{N-1}(\cdots(f_1(\mathbf{x}_u)))), \tag{5}$$

in which $\mathbf{x}_{rec}$ denotes the prediction of the network, $\mathbf{x}_u$ is the sequence of undersampled images with length T and also the input of the network, $f_i(\mathbf{x}_u; \boldsymbol{\theta}, \lambda, \Omega)$ is the network function for each iteration of optimisation step, and N is the number of iterations. We can compactly represent a single iteration f_i of our network as follows:

$$\mathbf{x}_{rnn}^{(i)} = \mathbf{x}_{rec}^{(i-1)} + \text{CRNN}(\mathbf{x}_{rec}^{(i-1)}), \tag{6a}$$

$$\mathbf{x}_{rec}^{(i)} = \text{DC}(\mathbf{x}_{rnn}^{(i)}; \mathbf{y}, \lambda_0, \Omega), \tag{6b}$$

where CRNN is a learnable block explained hereafter, DC is the data consistency step treated as a network layer, $\mathbf{x}_{rec}^{(i)}$ is the progressive reconstruction of the undersampled image $\mathbf{x}_u$ at iteration i with $\mathbf{x}_{rec}^{(0)} = \mathbf{x}_u$, $\mathbf{x}_{rnn}^{(i)}$ is the intermediate reconstruction image before the DC layer, and $\mathbf{y}$ is the acquired k-space samples. Note that the variables $\mathbf{x}_{rec}, \mathbf{x}_{rnn}$ are analogous to $\mathbf{x}, \mathbf{z}$ in Eq. 3 respectively. Here, we use CRNN to encode the update step, which can be seen as one step of a gradient descent in the sense of objective minimisation, or a more general approximation function regressing the difference $\mathbf{z}^{(i+1)} - \mathbf{x}^{(i)}$, i.e. the distance required to move to the next state. Moreover, note that in every iteration, CRNN updates its internal state $\mathcal{H}$ given an input which is discussed shortly. As such, CRNN also allows information to be propagated efficiently across iterations, in contrast to the sequential models using CNNs which collapse the intermediate feature representation to $\mathbf{z}^{(i)}$.

In order to exploit the dynamic nature and the temporal redundancy of our data, we further propose to jointly model the recurrence evolving over time for dynamic MRI reconstruction. The proposed CRNN-MRI network and CRNN block are shown in Fig. 2(a), in which CRNN block comprised of 5 components:

1) bidirectional convolutional recurrent units evolving over time and iterations (BCRNN-t-i),
2) convolutional recurrent units evolving over iterations only (CRNN-i),

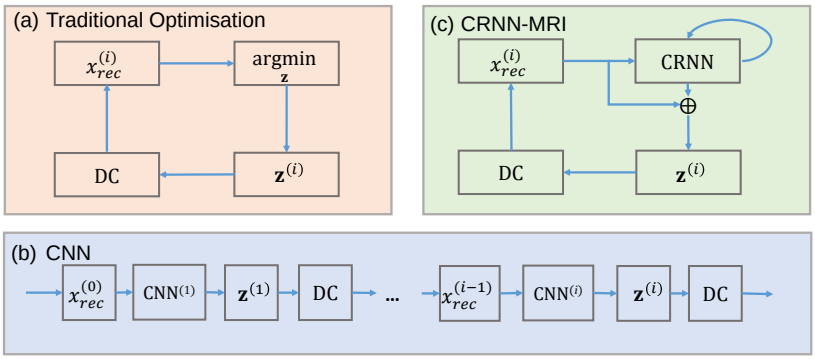

Fig. 1: (a) Traditional optimisation algorithm using variable splitting and alternate minimisation approach, (b) the optimisation unrolled into a deep convolutional network incorporating the data consistency step, and (c) the proposed architecture which models optimisation recurrence.

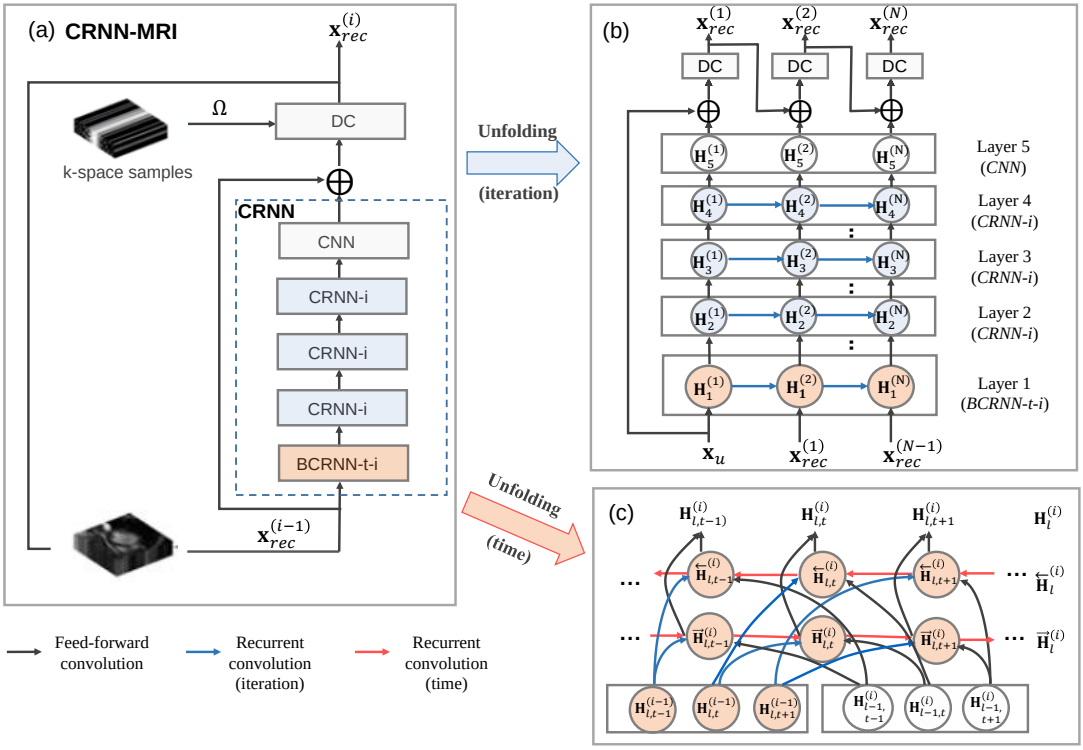

Fig. 2: (a) The overall architecture of proposed CRNN-MRI network for MRI reconstruction. (b) The structure of the proposed network when unfolded over iterations, in which $\mathbf{x}_{rec}^{(0)} = \mathbf{x}_u$. (c) The structure of BCRNN-t-i layer when unfolded over the time sequence. The black arrows indicate feed-forward convolutions. The blue arrows and red arrows indicate recurrent convolutions over iterations and the time sequence respectively.

3) 2D convolutional neural network (CNN),
4) residual connection and
5) DC layers.

We introduce details of the components of our network in the following subsections.

1) CRNN-i: As aforementioned, we encapsulate the iterative optimisation procedures explicitly with RNNs. In the CRNN-i unit, the iteration step is viewed as the sequential step in the vanilla RNN. If the network is unfolded over the iteration dimension, the network can be illustrated as in Fig. 2(b), where information is propagated between iterations. Here we use $\mathbf{H}$ to denote the feature representation of our sequence of frames throughout the network. $\mathbf{H}_l^{(i)}$ denotes the representation at layer l (subscript) and iteration step i (superscript). Therefore, at iteration (i), given the input $\mathbf{H}_{l-1}^{(i-1)}$ and the previous iteration's hidden state $\mathbf{H}_l^{(i-1)}$, the hidden state $\mathbf{H}_l^{(i)}$ at layer l of a CRNN-i unit can be formulated as:

$$\mathbf{H}_l^{(i)} = \sigma(\mathbf{W}_l * \mathbf{H}_{l-1}^{(i)} + \mathbf{W}_i * \mathbf{H}_l^{(i-1)} + \mathbf{B}_l). \quad (7)$$

Here $*$ represents convolution operation, $\mathbf{W}_l$ and $\mathbf{W}_i$ represent the filters of input-to-hidden convolutions and hidden-to-hidden recurrent convolutions evolving over iterations

respectively, and $\mathbf{B}_l$ represents a bias term. Here $\mathbf{H}_l^{(i)}$ is the representation of the whole T sequence with shape $(batchsize, T, n_c, D_x, D_y)$, where n_c is the number of channels which is 2 at the input and output but is greater while processing inside the network, and the convolutions are computed on the last two dimensions. The latent features are activated by the rectifier linear unit (ReLU) as a choice of nonlinearity, i.e. $\sigma(x) = max(0, x)$.

The CRNN-i unit offers several advantages compared to independently unrolling convolutional filters at each stage. Firstly, compared to CNNs where the latent representation from the previous state is not propagated, the hidden-to-hidden iteration connections in CRNN-i units allow contextual spatial information gathered at previous iterations to be passed to the future iterations. This enables the reconstruction step at each iteration to be optimised not only based on the output image but also based on the hidden features from previous iterations, where the hidden connection convolutions can "memorise" the useful features to avoid redundant computation. Secondly, as the iteration number increases, the effective receptive field of a CRNN-i unit in the spatial domain also expands whereas CNN resets it at each iteration. This property allows the network to further improve the reconstruction by allowing it to have better contextual support. In addition, since the weight parameters are shared across iterations, it greatly reduces the number of parameters compared to CNNs, potentially offering better generalization properties.

In this work, we use a vanilla RNN to model the recurrence due to its simplicity. Note this can be naturally generalised to other RNN units, such as long short-term memory (LSTM) and gated recurrent unit (GRU), which are considered to have better memory properties, although using these units would significantly increase computational complexity.

2) BCRNN-t-i: Dynamic MR images exhibit high temporal redundancy, which is often exploited as a-priori knowledge to regularise the reconstruction. Hence, it is also beneficial for the network to learn the dynamics of sequences. To this extent, we propose a bidirectional convolutional recurrent unit (BCRNN-t-i) to exploit both temporal *and* iteration dependencies jointly. BCRNN-t-i includes three convolution layers: one on the input which comes into the unit from the previous layer, one on the hidden state from the past and future time frames and the one on the hidden state from the previous iteration of the unit (Fig. 2(c)). Note that we simultaneously consider temporal dependencies from past and future time frames, and the encoding weights are shared for both directions. The output for the BCRNN-t-i layer is obtained by summing the feature maps learned from both directions. The illustration figure of the unit when it is unfolded over time sequence is shown in Fig. 2(c).

As we need to propagate information along temporal dimensions in this unit, here we introduce an additional index t in the notation to represent the variables related with time frame t. Here $\mathbf{H}_{l,t}^{(i)}$ represents feature representations at l-th layer, time frame t, and at iteration i, $\overrightarrow{\mathbf{H}}_{l,t}^{(i)}$ denotes the representations calculated when information is propagated forward inside the BCRNN-t-i unit, and similarly, $\overleftarrow{\mathbf{H}}_{l,t}^{(i)}$ denotes the one in the backward direction. Therefore, for the formulation of BCRNN-t-i unit, given (1) the current input representation of the l-th layer at time frame t and iteration step i, which is the output representation from $(l-1)$-th layer $\mathbf{H}_{l-1,t}^{(i)}$, (2) the previous iteration's hidden representation within the same layer $\mathbf{H}_{l,t}^{(i-1)}$, (3) the hidden representation of the past time frame $\overrightarrow{\mathbf{H}}_{l,t-1}^{(i)}$, and the hidden representation of the future time frame $\overleftarrow{\mathbf{H}}_{l,t+1}^{(i)}$, then the hidden state representation of the current l-th layer of time frame t at iteration i, $\mathbf{H}_{l,t}^{(i)}$ with shape $(batchsize, n_c, D_x, D_y)$, can be formulated as:

$$\begin{aligned}
\overrightarrow{\mathbf{H}}_{l,t}^{(i)} &= \sigma(\mathbf{W}_l * \mathbf{H}_{l-1,t}^{(i)} + \mathbf{W}_t * \overrightarrow{\mathbf{H}}_{l,t-1}^{i} + \mathbf{W}_i * \mathbf{H}_{l,t}^{(i-1)} + \overrightarrow{\mathbf{B}}_l), \\
\overleftarrow{\mathbf{H}}_{l,t}^{(i)} &= \sigma(\mathbf{W}_l * \mathbf{H}_{l-1,t}^{(i)} + \mathbf{W}_t * \overleftarrow{\mathbf{H}}_{l,t+1}^{(i)} + \mathbf{W}_i * \mathbf{H}_{l,t}^{(i-1)} + \overleftarrow{\mathbf{B}}_l), \\
\mathbf{H}_{l,t}^{(i)} &= \overrightarrow{\mathbf{H}}_{l,t}^{(i)} + \overleftarrow{\mathbf{H}}_{l,t}^{(i)},
\end{aligned} \tag{8}$$

Similar to the notation in Section III-B1, $\mathbf{W}_t$ represents the filters of recurrent convolutions evolving over time. When $l = 1$ and $i = 1$, $\mathbf{H}_{0,t}^{(1)} = \mathbf{x}_{u_t}$, that is the t-th frame of undersampled input data, and when $l = 1$ and $i = 2, ...T$, $\mathbf{H}_{0,t}^{(i)} = \mathbf{x}_{rec_t}^{(i-1)}$, which stands for the t-th frame of the intermediate reconstruction result from iteration $i-1$. For $\mathbf{H}_{l,t}^{(0)}$, $\overrightarrow{\mathbf{H}}_{l,0}^{(i)}$ and $\overleftarrow{\mathbf{H}}_{l,T+1}^{(i)}$, they are set to be zero initial hidden states.

The temporal connections of BCRNN-t-i allow information to be propagated across the whole T time frames, enabling it to learn the differences and correlations of successive frames. The filter responses of recurrent convolutions evolving over time express dynamic changing biases, which focus on modelling the temporal changes across frames, while the filter responses of recurrent convolutions over iterations focus on learning the spatial refinement across consecutive iteration steps. In addition, we note that learning recurrent layers along the temporal direction is different to using 3D convolution along the space and temporal direction. 3D convolution seeks invariant features across space-time, hence several layers of 3D convolutions are required before the information from the whole sequence can be propagated to a particular time frame. On the other hand, learning recurrent 2D convolutions enables the model to easily and efficiently propagate the information through time, which also yields fewer parameters and a lower computational cost.

In summary, the set of hidden states for a CRNN block to update at iteration i is $\mathcal{H} = \{\mathbf{H}_l^{(i)}, \mathbf{H}_{l,t}^{(i)}, \overleftarrow{\mathbf{H}}_{l,t}^{(i)}, \overrightarrow{\mathbf{H}}_{l,t}^{(i)}\}$, for $l = 1, \ldots, L$ and $t = 1, \ldots, T$, where L is the total number of layers in the CRNN block and T is the total number of time frames.

C. Network Learning

Given the training data S of input-target pairs $(\mathbf{x}_u, \mathbf{x}_t)$, the network learning proceeds by minimizing the pixel-wise mean squared error (MSE) between the predicted reconstructed MR image and the fully sampled ground truth data:

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{n_S} \sum_{(\mathbf{x}_u, \mathbf{x}_t) \in S} \|\mathbf{x}_t - \mathbf{x}_{rec}\|_2^2 \tag{9}$$

where $\boldsymbol{\theta} = \{\mathbf{W}_l, \mathbf{W}_i, \mathbf{W}_t, \mathbf{B}_l\}$, $l = 1 \ldots L$, and n_S stands for the number of samples in the training set S. Note that

the total number of time sequences T and iteration steps N assumed by the network before performing the reconstruction is a free parameter that must be specified in advance. The network weights were initialised using He initialization [26] and it was trained using the Adam optimiser [27]. During training, gradients were hard-clipped to the range of $[-5, 5]$ to mitigate the gradient explosion problem. The network was implemented in Python using Theano and Lasagne libraries.

IV. Experiments

A. Dataset and Implementation Details

The proposed method was evaluated using a complex-valued MR dataset consisting of 10 fully sampled short-axis cardiac cine MR scans. Each scan contains a single slice SSFP acquisition with 30 temporal frames, which have a 320×320 mm field of view and 10 mm thickness. The raw data consists of 32-channel data with sampling matrix size 192×190, which was then zero-filled to the matrix size 256×256. The raw multi-coil data was reconstructed using SENSE [28] with no undersampling and retrospective gating. Coil sensitivity maps were normalized to a body coil image and used to produce a single complex-valued reconstructed image. In experiments, the complex valued images were back-transformed to regenerate k-space samples, simulating a fully sampled single-coil acquisition. The input undersampled image sequences were generated by randomly undersampling the k-space samples using Cartesian undersampling masks, with undersampling patterns adopted from [1]: for each frame the eight lowest spatial frequencies were acquired, and the sampling probability of k-space lines along the phase-encoding direction was determined by a zero-mean Gaussian distribution. Note that the undersampling rates are stated with respect to the matrix size of raw data, which is 192×190.

The architecture of the proposed network used in the experiment is shown in Fig. 2: each iteration of the CRNN block contains five units: one layer of BCRNN-t-i, followed by three layers of CRNN-i units, and followed by a CNN unit. For all CRNN-i and BCRNN-t-i units, we used a kernel size $k = 3$ and the number of filters was set to $n_f = 64$ for Proposed-A and $n_f = 128$ for Proposed-B in Table I. The CNN after the CRNN-i units contains one convolution layer with $k = 3$ and $n_f = 2$, which projects the extracted representation back to the image domain which contains complex-valued images expressed using two channels. The output of the CRNN block is connected to the residual connection, which sums the output of the block with its input. Finally, we used DC layers on top of the CRNN output layers. During training, the iteration step is set to be $N = 10$, and the time sequence for training is $T = 30$. Note that this architecture is by no means optimal and more layers can be added to increase the ability of our network to better capture the data structures. While the original sequence has size $256 \times 256 \times T$, we extract patches of size $256 \times D_{patch} \times T$, where D_{patch} is the patch size and the direction of patch extraction corresponds to the frequency-encoding direction. Note that since we only consider Cartesian undersampling, the aliasing occurs only along the phase encoding direction, so patch extraction does not alter the aliasing artefact. The evaluation was done via a 3-fold cross validation. The minibatch size during the training was set to 1, and we observed that the performance can reach a plateau within 6×10^4 backpropagations.

B. Evaluation Method

We compared the proposed method with the representative algorithms of the CS-based dynamic MRI reconstruction, such as k-t FOCUSS [1] and k-t SLR [2], and two variants of 3D CNN networks named 3D CNN-S and 3D CNN in our experiments. The built baseline 3D CNN networks share the same architecture with the proposed CRNN-MRI network but all the recurrent units and 2D CNN units were replaced with 3D convolutional units, that is, in each iteration, the 3D CNN block contain 5 layers of 3D convolutions, one DC layer and a residual connection. Here 3D CNN-S refers to network sharing weights across iterations, however, this does not employ the hidden-to-hidden connection as in the CRNN-i unit. The 3D CNN-S architecture was chosen so as to make a fair comparison with the proposed model using a comparable number of network parameters. In contrast, 3D CNN refers to the network without weight sharing, in which the network capacity is $N = 10$ times of that of 3D CNN-S, and approximately 12 times more than that of our first proposed method (Proposed-A).

Reconstruction results were evaluated based on the following quantitative metrics: MSE, peak-to-noise-ratio (PSNR), structural similarity index (SSIM) [29] and high frequency error norm (HFEN) [30]. The choice of the these metrics was made to evaluate the reconstruction results with complimentary emphasis. MSE and PSNR were chosen to evaluate the overall accuracy of the reconstruction quality. SSIM put emphasis on image quality perception. HFEN was used to quantify the quality of the fine features and edges in the reconstructions, and here we employed the same filter specification as in [30], [31] with the filter kernel size 15×15 pixels and a standard deviation of 1.5 pixels. For PSNR and SSIM, it is the higher the better, while for MSE and HFEN, it is the lower the better.

C. Results

The comparison results of all methods are reported in Table I, where we evaluated the quantitative metrics, network capacity and reconstruction time. Numbers shown in Table I are mean values of corresponding metrics with standard deviation of different subjects in parenthesis. Bold numbers in Table I indicate the better performance of the proposed methods than the competing ones. Compared with the baseline method (k-t FOCUSS and k-t SLR), the proposed methods outperform them by a considerable margin at different acceleration rates. When compared with deep learning methods, note that the network capacity of Proposed-A is comparable with that of 3D CNN-S and the capacity of Propose-B is around one third of that of 3D CNN. Though their capacities are much smaller, both Proposed-A and Proposed-B outperform 3D CNN-S and 3D CNN for all acceleration rates by a large margin, which shows the competitiveness and effectiveness of our method. In addition, we can see a substantial improvement of the reconstruction results on all acceleration rates and in all metrics when the

TABLE I: Performance comparisons (MSE, PSNR:dB, SSIM, and HFEN) on dynamic cardiac data with different acceleration rates. MSE is scaled to 10^{-3}. The bold numbers are better results of the proposed methods than that of the other methods.

Method		k-t FOCUSS	k-t SLR	3D CNN-S	3D CNN	Proposed-A	Proposed-B
Capacity		-	-	338,946	3,389,460	**262,020**	**1,040,132**
6×	MSE	0.592 (0.199)	0.371(0.155)	0.385 (0.124)	0.275 (0.096)	**0.261 (0.097)**	**0.201 (0.074)**
	PSNR	32.506 (1.516)	34.632 (1.761)	34.370 (1.526)	35.841 (1.470)	**36.096 (1.539)**	**37.230 (1.559)**
	SSIM	0.953 (0.040)	0.970 (0.033)	0.976 (0.008)	0.983 (0.005)	**0.985 (0.004)**	**0.988 (0.003)**
	HFEN	0.211 (0.021)	0.161 (0.016)	0.170 (0.009)	0.138 (0.013)	**0.131 (0.013)**	**0.112 (0.010)**
9×	MSE	1.234 (0.801)	0.846 (0.572)	0.929 (0.474)	0.605 (0.324)	**0.516 (0.255)**	**0.405 (0.206)**
	PSNR	29.721 (2.339)	31.409 (2.404)	30.838 (2.246)	32.694 (2.179)	**33.281 (1.912)**	**33.379 (2.017)**
	SSIM	0.922 (0.043)	0.951 (0.025)	0.950 (0.016)	0.968 (0.010)	**0.972 (0.009)**	**0.979 (0.007)**
	HFEN	0.310(0.041)	0.260 (0.034)	0.280 (0.034)	0.215 (0.021)	**0.201 (0.025)**	**0.173 (0.021)**
11×	MSE	1.909 (0.828)	1.237 (0.620)	1.472 (0.733)	0.742 (0.325)	**0.688 (0.290)**	**0.610 (0.300)**
	PSNR	27.593 (2.038)	29.577 (2.211)	28.803 (2.151)	31.695 (1.985)	**31.986 (1.885)**	**32.575 (1.987)**
	SSIM	0.880 (0.060)	0.924 (0.034)	0.925 (0.022)	0.960 (0.010)	**0.964 (0.009)**	**0.968 (0.011)**
	HFEN	0.390 (0.023)	0.327 (0.028)	0.363 (0.041)	0.257 (0.029)	**0.248 (0.033)**	**0.227 (0.030)**
Time		15s	451s	8s	8s	**3s**	**6s**

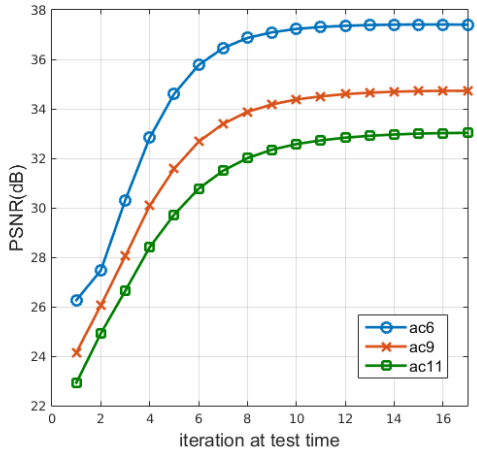

Fig. 3: Mean PSNR values (Proposed-B) vary with the number of iterations at test time on data with different acceleration rates. Here ac stands for acceleration rate.

number of network parameters is increased for the proposed method (Proposed-B), and therefore we will only show the results from Proposed-B in the following. The number of iterations used by the network at test time is set to be the same as the training stage, which is $N = 10$, however, if the iteration number is increased up to $N = 17$, it shows an improvement of 0.324dB on average. Fig. 3 shows the model's performance varying with the number of iterations at test time. In fact, for accelerated imaging, higher undersampling factors significantly add aliasing to the initial zero-filled reconstruction, making the reconstruction more challenging. This suggests that while the 3D CNN possesses higher modelling capacity owing to its large number of parameters, it may not necessarily be an ideal architecture to perform dynamic MR reconstruction, presumably because the simple CNN is not as efficient as propagating the information across the whole sequence.

A comparison of the visualization results of a reconstruction from $9\times$ acceleration is shown in Fig. 4 with the reconstructed images and their corresponding error maps from different reconstruction methods. As one can see, our proposed model (Proposed-B) can produce more faithful reconstructions for those parts of the image around the myocardium where there are large temporal changes. This is reflected by the fact that RNNs effectively use a larger receptive field to capture the characteristics of aliasing seen within the anatomy. For the 3D CNN approaches, it is also observed that it is not able to denoise the background region. This could be explained by the fact that 3D CNN only exploits local information due to the small filter sizes it used, while in contrast, the proposed CRNN improves the denoising of the background region because of its larger receptive field sizes. Their temporal profiles at $x = 120$ are shown in Fig. 5. Similarly, one can see that the proposed model has overall much smaller error, faithfully modelling the dynamic data. This suggests its capability to learn motion compensation implicitly between frames although the network is trained for the dynamic image reconstruction. It could be due to the fact that spatial and temporal features are learned separately in the proposed model while 3D CNN seeks invariant feature learning across space and time.

In terms of speed, the proposed RNN-based reconstruction is faster than the 3D CNN approaches because it only performs convolution along time once per iteration, removing the redundant 3D convolutions which are computationally expensive. Reconstruction time of 3D CNN and the proposed methods reported in Table I were calculated on a GPU GeForce GTX 1080, and the time for k-t FOCUSS and k-t SLR were calculated on CPU.

V. Discussion and Conclusion

In this work, we have demonstrated that the presented network is capable of producing faithful image reconstructions from highly undersampled data, both in terms of various quantitative metrics as well as inspection of error maps. In contrast to unrolled deep network architectures proposed previously, we modelled the recurrent nature of the optimisation iteration using hidden representations with the ability to

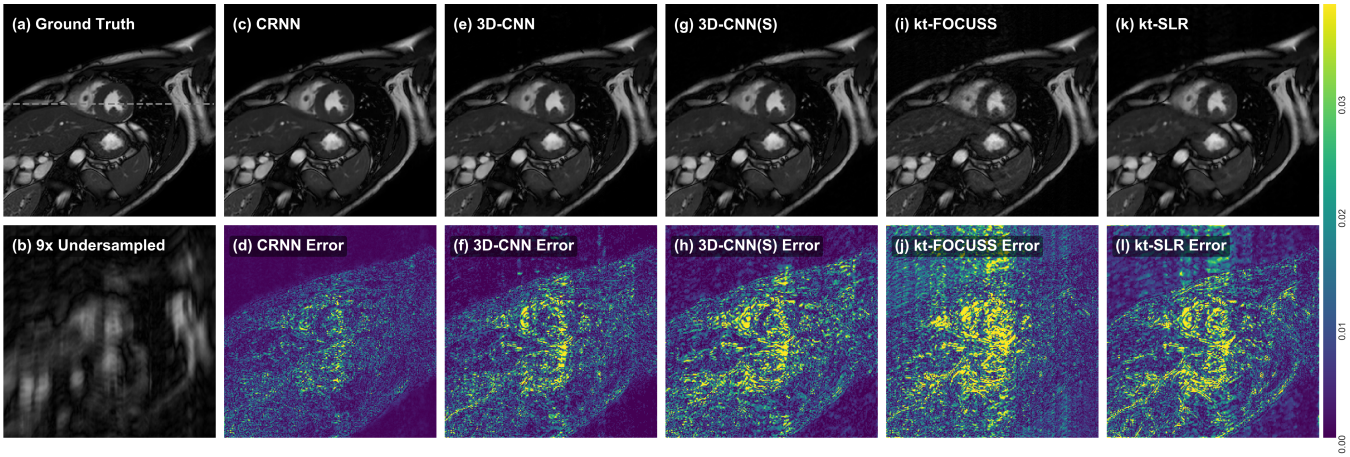

Fig. 4: The comparison of reconstructions on spatial dimension with their error maps. (a) Ground Truth (b) Undersampled image by acceleration factor 9 (c,d) Proposed-B (e,f) 3D CNN (g,h) 3D CNN-S (i,j) k-t FOCUSS (k,l) k-t SLR

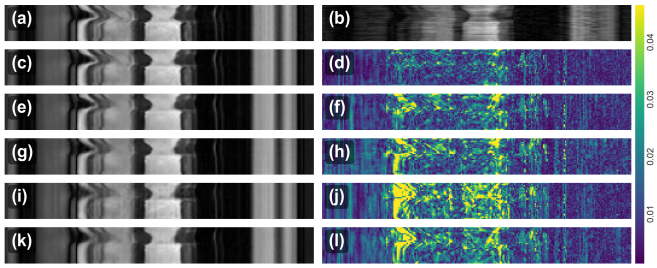

Fig. 5: The comparison of reconstructions along temporal dimension with their error maps. (a) Ground Truth (b) Undersampled image by acceleration factor 9 (c,d) Proposed-B (e,f) 3D CNN (g,h) 3D CNN-S (i,j) k-t FOCUSS (k,l) k-t SLR

retain and propagate information across the optimisation steps. Compared with 3D CNN models, the proposed methods have a much lower network capacity but still have a higher accuracy, reflecting the effectiveness of our architecture. This is due to the ability of the proposed RNN units to increase the receptive field size while iteration steps increase, as well as to efficiently propagate information across the temporal direction. In addition, our network also offers very fast reconstruction on a GPU GeForce GTX 1080 compared with other competing methods.

In this work, we modeled the recurrence using the relatively simple (vanilla) RNN architecture. For the future work, we will explore other recurrent units such as LSTM or GRU. As they are trained to explicitly select what to remember, they may allow the units to better control the flow of information and could reduce the number of iterations required for the network to generate high-quality output. In addition, we have found that the majority of errors between the reconstructed image and the fully sampled image lie at the part where motion exists, indicating that motion exhibits a challenge for such dynamic sequence reconstruction, and CNNs or RNNs trained for reconstruction loss cannot fully learn such motion compensation implicitly during training. Thus it will be interesting to explore the benefits of doing simultaneous motion compensation and image reconstruction for the improvement in the dynamic region. Additionally, current analysis only considers a single coil setup. In the future, we will also aim at investigating such methods in a scenario where multiple coil data from parallel MR imaging can be used jointly for higher acceleration acquisition.

To conclude, inspired by variable splitting and alternate minimisation strategies, we have presented an end-to-end deep learning solution, CRNN-MRI, for accelerated dynamic MRI reconstruction, with a forward, CRNN block implicitly learning iterative denoising interleaved by data consistency layers to enforce data fidelity. In particular, the CRNN architecture is composed of the proposed novel variants of convolutional recurrent unit which evolves over two dimensions: time and iterations. The proposed network is able to learn both the temporal dependency and the iterative reconstruction process effectively, and outperformed the other competing methods in terms of computational complexity, reconstruction accuracy and speed for different undersampling rates.

REFERENCES

[1] H. Jung, J. C. Ye, and E. Y. Kim, "Improved k–t BLAST and k–t SENSE using FOCUSS," *Physics in medicine and biology*, vol. 52, no. 11, p. 3201, 2007.

[2] S. G. Lingala, Y. Hu, E. Dibella, and M. Jacob, "Accelerated dynamic MRI exploiting sparsity and low-rank structure: K-t SLR," *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1042–1054, 2011.

[3] J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for dynamic MR image reconstruction," *arXiv preprint arXiv:1704.02422*, 2017.

[4] D. Batenkov, Y. Romano, and M. Elad, "On the global-local dichotomy in sparsity modeling," *arXiv preprint arXiv:1702.03446*, 2017.

[5] J. Tsao, P. Boesiger, and K. P. Pruessmann, "k-t BLAST and k-t SENSE: Dynamic MRI with high frame rate exploiting spatiotemporal correlations," *Magnetic Resonance in Medicine*, vol. 50, no. 5, pp. 1031–1042, 2003.

[6] J. Caballero, A. N. Price, D. Rueckert, and J. V. Hajnal, "Dictionary learning and time sparsity for dynamic MR data reconstruction," *IEEE Transactions on Medical Imaging*, vol. 33, no. 4, pp. 979–994, 2014.

[7] R. Otazo, E. Candès, and D. K. Sodickson, "Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components," *Magnetic Resonance in Medicine*, vol. 73, no. 3, pp. 1125–1136, 2015.

[8] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing mri," *IEEE signal processing magazine*, vol. 25, no. 2, pp. 72–82, 2008.

[9] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll, "Learning a variational network for reconstruction of accelerated MRI data," *arXiv preprint arXiv:1704.00447*, 2017.

[10] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*. Springer, 2014, pp. 184–199.

[11] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

[12] D. Lee, J. Yoo, and J. C. Ye, "Deep artifact learning for compressed sensing and parallel MRI," *arXiv preprint arXiv:1703.01120*, 2017.

[13] Y. S. Han, J. Yoo, and J. C. Ye, "Deep learning with domain adaptation for accelerated projection reconstruction MR," *arXiv preprint arXiv:1703.01135*, 2017.

[14] S. Wang, Z. Su, L. Ying, X. Peng, and D. Liang, "Exploiting deep convolutional neural network for fast magnetic resonance imaging," in *ISMRM 24th Annual Meeting and Exhibition*, 2016.

[15] S. Wang, N. Huang, T. Zhao, Y. Yang, L. Ying, and D. Liang, "1D Partial Fourier Parallel MR imaging with deep convolutional neural network," in *ISMRM 25th Annual Meeting and Exhibition*, vol. 47, no. 6, 2017, pp. 2016–2017.

[16] J. Adler and O. Öktem, "Learned primal-dual reconstruction," *arXiv preprint arXiv:1707.06474*, 2017.

[17] J. Sun, H. Li, Z. Xu *et al.*, "Deep ADMM-Net for compressive sensing mri," in *Advances in Neural Information Processing Systems*, 2016, pp. 10–18.

[18] J. Schlemper, J. Caballero, J. V. Hajnal, A. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for MR image reconstruction," *arXiv preprint arXiv:1703.00555*, 2017.

[19] J. Adler and O. Öktem, "Solving ill-posed inverse problems using iterative deep neural networks," *arXiv preprint arXiv:1704.04058*, 2017.

[20] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," *arXiv preprint arXiv:1704.03264*, 2017.

[21] J. Chang, C.-L. Li, B. Póczos, B. Kumar, and A. C. Sankaranarayanan, "One network to solve them all—solving linear inverse problems using deep projection models," *arXiv preprint arXiv:1703.09912*, 2017.

[22] K. Gregor, I. Danihelka, A. Graves, D. Rezende, and D. Wierstra, "Draw: A recurrent neural network for image generation," in *Proceedings of The 32nd International Conference on Machine Learning*, 2015, pp. 1462–1471.

[23] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3367–3375.

[24] J. Kuen, Z. Wang, and G. Wang, "Recurrent attentional networks for saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3668–3677.

[25] Y. Huang, W. Wang, and L. Wang, "Bidirectional recurrent convolutional networks for multi-frame super-resolution," in *Advances in Neural Information Processing Systems*, 2015, pp. 235–243.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

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

[28] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, P. Boesiger *et al.*, "SENSE: sensitivity encoding for fast MRI," *Magnetic resonance in medicine*, vol. 42, no. 5, pp. 952–962, 1999.

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

[30] S. Ravishankar and Y. Bresler, "Mr image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE transactions on medical imaging*, vol. 30, no. 5, pp. 1028–1041, 2011.

[31] X. Miao, S. G. Lingala, Y. Guo, T. Jao, M. Usman, C. Prieto, and K. S. Nayak, "Accelerated cardiac cine mri using locally low rank and finite difference constraints," *Magnetic resonance imaging*, vol. 34, no. 6, pp. 707–714, 2016.