MoDL-MUSSELS: Model-Based Deep Learning for Multishot Sensitivity-Encoded Diffusion MRI

Hemant K. Aggarwal, IEEE [Member],
department of electrical and computer engineering, University of Iowa, Iowa, USA.

Merry P. Mani,
Division of neuroradiology, University of Iowa, Iowa, USA.

Mathews Jacob, IEEE [Senior Member]
department of electrical and computer engineering, University of Iowa, Iowa, USA.

Abstract

We introduce a model-based deep learning architecture termed MoDL-MUSSELS for the correction of phase errors in multishot diffusion-weighted echo-planar MR images. The proposed algorithm is a generalization of the existing MUSSELS algorithm with similar performance but significantly reduced computational complexity. In this work, we show that an iterative re-weighted least-squares implementation of MUSSELS alternates between a multichannel filter bank and the enforcement of data consistency. The multichannel filter bank projects the data to the signal subspace, thus exploiting the annihilation relations between shots. Due to the high computational complexity of the self-learned filter bank, we propose replacing it with a convolutional neural network (CNN) whose parameters are learned from exemplary data. The proposed CNN is a hybrid model involving a multichannel CNN in the k-space and another CNN in the image space. The k-space CNN exploits the annihilation relations between the shot images, while the image domain network is used to project the data to an image manifold. The experiments show that the proposed scheme can yield reconstructions that are comparable to state-of-the-art methods while offering several orders of magnitude reduction in run-time.

I. INTRODUCTION

Diffusion MRI (DMRI), which is sensitive to anisotropic diffusion processes in the brain tissue, has the potential to provide rich information on white matter anatomy [1]. It has several applications, including the studies of neurological disorders [2], the aging process [3], and acute stroke [4]. Diffusion MRI relies on large bipolar directional gradients to

encode water diffusion, which attenuates the signals from diffusing molecules in the direction of the gradient. The diffusion-sensitized signal is often spatially encoded using single-shot echo-planar imaging (ssEPI), which allows the acquisition of the entire k-space in a single excitation and readout. While such acquisitions can offer high sampling efficiency, the longer readout makes the acquisition vulnerable to distortions induced by B0 inhomogeneity. Specifically, the recovered images often exhibit geometric distortions [5]. These artefacts, resulting from the long readouts, essentially limit the extent of k-space coverage and thereby the spatial resolution that ssEPI sequences can achieve.

Multishot echo-planar imaging (msEPI) methods were introduced to minimize the distortions related to the long readouts in ssEPI. This scheme splits the k-space sampling over multiple excitations and readouts, resulting in shorter readout lengths for each shot, as shown in Fig. 1. While multishot imaging can offer high resolution, a challenge associated with this scheme is its vulnerability to inter-shot motion in the diffusion setting. Specifically, subtle physiological motion during the large bipolar gradients manifests as phase differences between different shots. The direct combination of the k-space data from these shots results in artefacts in the diffusion-weighted images (DWI) arising from phase inconsistencies.

We recently introduced a multishot sensitivity-encoded diffusion data recovery algorithm using structured low-rank matrix completion (MUSSELS) [6], which allows the reconstruction of DWI that are immune to the motion-induced phase artefacts. The method exploits the redundancy between the Fourier samples of the shots to jointly recover the missing k-space samples in each of the shots [7]. The k-space data recovery is then posed as a matrix completion problem that utilizes a structured low-rank algorithm and parallel imaging to recover the missing k-space data in each shot. While this scheme can offer state-of-the-art results, the challenge is the high computational complexity. The large data size and the need for matrix lifting make it challenging to reconstruct the high-resolution data from different directions and slices despite the existence of fast structured low-rank algorithms [7], [8].

In this paper, we introduce a novel deep learning framework to minimize the computational complexity of MUSSELS [6]. This work is inspired by the convolutional network structure of MUSSELS and is formulated in k-space to exploit the convolutional relations between the Fourier samples of the shots. The proposed scheme is also motivated by our recent work on model-based deep learning (MoDL) [9] and similar algorithms that rely on the unrolling of iterative algorithms [10]–[12]. The main benefit of MoDL is the ability to exploit the physics of the acquisition scheme and the ability to incorporate multiple regularization priors [13], in a deep learning setting, to achieve improved performance. The use of the conjugate-gradient algorithm within the network to enforce data consistency in MoDL provides improved performance for a specified number of iterations. The sharing of network parameters across iterations enables MoDL to keep the number of learned parameters decoupled from the number of iterations, thus providing good convergence without increasing the number of trainable parameters. A smaller number of trainable parameters translates to significantly reduced training data demands, which is particularly attractive for data-scarce medical-imaging applications.

We first introduce an approach based on the iterative reweighted least-squares algorithm (IRLS) [14] to solve the MUSSELS cost function [6]. The MUSSELS algorithm [6], which is based on iterative singular value shrinkage, alternates between a data-consistency block and a low-rank matrix recovery block. By contrast, the IRLS-MUSSELS algorithm [7] alternates between a data-consistency block and a residual multichannel[1] convolution block. The multichannel convolution block can be viewed as the projection of the data to the nullspace of the multichannel signals; the subtraction of the result from the original ones, induced by the residual structure, projects the data to the signal subspace, thus removing the artefacts in the signal. The IRLS-MUSSELS algorithm learns the parameters of the denoising filter from the data itself, which requires several iterations. Motivated by our earlier work [9], we propose replacing the multichannel linear convolution block in IRLS-MUSSELS with a convolutional neural network (CNN). Unlike the self-learning strategy in IRLS-MUSSELS, where the filter parameters are learned from the measured data itself, we propose learning the parameters of the non-linear CNN from exemplar data. We hypothesize that the non-linear structure of the CNN will enable us to learn and generalize from examples. The learned CNN will facilitate the projection of each test dataset to the associated signal subspace. While the architecture is conceptually similar to MoDL, the main difference is the extension to multishot settings and the learning in the Fourier domain (k-space) enabled by the IRLS-MUSSELS reformulation.

The proposed framework has similarities to recent k-space deep learning strategies [16]–[19], which also exploit the convolution relations in the Fourier domain. The main distinction of the proposed scheme with these methods is the model-based framework, along with the training of the unrolled network. Many of the current schemes [18] are not designed for the parallel imaging setting. The use of the conjugate gradient steps in our network allows us to account for parallel imaging efficiently, requiring few iterations. We also note the relation of the proposed work with that of Akcakaya et al. [20], which uses a self-learned network to recover parallel MRI data. The weights of the network are estimated from the measured data itself. Since we estimate the weights from exemplar data, the proposed scheme is significantly faster.

II. BACKGROUND

A. Problem formulation

The high-resolution DMRI requires long-duration EPI readouts that are vulnerable to field-inhomogeneity-induced spatial distortions. Also, the large rewinder gradients make the achievable echo-time rather long, resulting in lower signal-to-noise ratio (SNR). To minimize these distortions, it is common practice to acquire the data using msEPI schemes for high-resolution applications. These schemes acquire a highly undersampled subset of the k-space at each shot. Since the subsets are complementary, the data from all these shots can be combined to obtain the final image. The image acquisition of the ith shot and the jth coil can be expressed as

---

[1] The term multichannel is used in a traditional signal processing sense to refer to multichannel convolution using a multichannel filter. The different channels corresponds to the images from different shots of the multishot data. Since we rely on the SENSE forward model [15], the channels do not refer to multi-coil data in parallel MRI.

$$y_{i,j}[\mathbf{k}] = \int_{\mathbb{R}^2} \rho(\mathbf{r})s_j(\mathbf{r})\exp\left(\hat{\imath}\,\mathbf{k}^T\mathbf{r}\right)d\mathbf{r} + n_{i,j}[\mathbf{k}]; \quad \forall \mathbf{k} \in \Theta_i. \tag{1}$$

Here, $\mathbf{s}_j(\mathbf{r})$ denotes the coil sensitivity of the jth coil and Θ_i, $i = 1,..,N$, denotes the subset of the k-space that is acquired at the ith-shot. Note that the sampling indices of the different shots are complementary, implying that the combination of the data from the different shots will result in a fully sampled image. Specifically, we have $\cup_{i=1}^N \Theta_i = \Theta$, where Θ is the Fourier grid corresponding to the fully sampled image. The above relation, to acquire the desired image $\rho(\mathbf{r})$ from N shots, can be compactly represented as

$$\mathbf{y}_i = \mathcal{A}_i(\rho(\mathbf{r})) + \mathbf{n}, \; i = 1, \ldots, N \tag{2}$$

in the absence of phase errors. Here, $\mathbf{y}_i$ represents the under-sampled multichannel measurements of the ith shot acquired using the acquisition operator A_i, and $\mathbf{n}$ represents the additive Gaussian noise that may corrupt the samples during acquisition.

Diffusion MRI uses large bipolar diffusion gradients to encode the diffusion motion of water molecules. Unfortunately, subtle physiological motion between the bipolar gradients often manifests as phase errors in the acquisition. With the addition of the unknown phase function $\phi_i(\mathbf{r})$, $|\phi_i(\mathbf{r})| = 1$ introduced by physiological motion, the forward model is modified as

$$\mathbf{y}_i = \mathcal{A}_i(\underbrace{\rho(\mathbf{r})\phi_i(\mathbf{r})}_{\rho_i(\mathbf{r})}) + \mathbf{n}, \quad i = 1, .., N. \tag{3}$$

If the phase errors $\phi_i(\mathbf{r})$, $i = 1, .., N$, are uncompensated, the image obtained by the combination of $\mathbf{y}_i$, $i = 1, .., N$ will show artefacts arising from the inconsistent phase. The widely used multishot method, termed MUSE [21], [22], relies on the independent estimation of $\phi_i(\mathbf{r})$ from low-resolution reconstructions of the phase-corrupted images $\rho_i(\mathbf{r})$. The forward model can be compactly written as $\mathbf{y} = A(\rho)$, where $\rho = \left[\rho_1^T, \ldots \rho_N^T\right]^T$ is the vector of multishot images. Once the phases are estimated, the reconstruction is posed as a phase-aware reconstruction [21], [22].

B. Brief Review of MUSSELS

The MUSSELS algorithm [6] relies on a structured low-rank formulation to jointly recover the phase-corrupted images ρ_i from their under-sampled multi-coil measurements. The MUSSELS algorithm capitalizes on the multi-coil nature of the measurements as well as annihilation relations between the phase-corrupted images. The key observation is that these phase-corrupted images satisfy an image domain annihilation relation [23]

$$\rho_i(\text{r})\phi_j(\text{r}) - \rho_j(\text{r})\phi_i(\text{r}) = 0, \forall \text{r}. \tag{4}$$

This multiplicative annihilation relation, resulting from phase inconsistencies, translates to convolution relations in the Fourier domain:

$$\widehat{\rho_i}(\mathbf{k}) * \widehat{\phi_j}(\mathbf{k}) - \widehat{\rho_j}(\mathbf{k}) * \widehat{\phi_i}(\mathbf{k}) = 0 \quad \forall \mathbf{k}, \tag{5}$$

where $\hat{x}$ denotes the Fourier transform of x. Since the phase images $\phi_j(\mathbf{r})$ are smooth, their Fourier coefficients $\widehat{\phi_j}(\mathrm{k})$ can be assumed to be support-limited to a region Λ in the Fourier domain. This allows us to rewrite the convolution relations in (5) in a matrix form using block-Hankel convolution matrices $\mathbf{H}_\Lambda^\Gamma(\rho)$. The matrix product $\mathbf{H}_\Lambda^\Gamma(\rho)\,\mathbf{s}$ corresponds to the 2D convolution between a signal ρ supported on a grid Γ and the filter $\mathbf{s}$ of size Λ. Thus, the Fourier domain convolution relations can be compactly expressed using matrices [6] as

$$\left[\mathbf{H}_\Lambda^\Gamma(\hat{\rho}_i) \mid \mathbf{H}_\Lambda^\Gamma(\hat{\rho}_j) \right] \underbrace{\left[\frac{\hat{\phi}_j}{-\hat{\phi}_i} \right]}_{\mathbf{s}} = 0. \tag{6}$$

We note that there exists a similar annihilation relation between each pair of shots, which implies that the structured matrix

$$\mathbf{T}(\hat{\rho}) = \left[\mathbf{H}_\Lambda^\Gamma(\hat{\rho}_1) \mid \cdots \mid \mathbf{H}_\Lambda^\Gamma(\hat{\rho}_N) \right] \tag{7}$$

is low-rank. The MUSSELS algorithm [6] recovers the multishot images from their undersampled k-space measurements by solving

$$\tilde{\rho} = \underset{\rho}{\mathrm{argmin}} \parallel \mathscr{A}(\rho) - \mathbf{y} \parallel_2^2 + \lambda \parallel \mathbf{T}(\hat{\rho}) \parallel_*, \tag{8}$$

where $\parallel \cdot \parallel_*$ denotes the nuclear norm. The above problem is solved in earlier work [6] using an iterative shrinkage algorithm.

III. DEEP LEARNED MUSSELS

A. IRLS reformulation of MUSSELS

To bring the MUSSELS framework to the MoDL setting, we first introduce an IRLS reformulation [14] of the MUSSELS. Using an auxiliary variable $\mathbf{z}$, we rewrite (8) as

$$\underset{\rho, \mathbf{z}}{\mathrm{argmin}} \parallel \mathscr{A}(\rho) - \mathbf{y} \parallel_2^2 + \beta \parallel \hat{\rho} - \mathbf{z} \parallel_F^2 + \lambda \parallel \mathbf{T}(\mathbf{z}) \parallel_*. \tag{9}$$

We observe that (9) is equivalent to (8) as $\beta \to \infty$. An alternating minimization algorithm to solve the above problem yields the following steps:

$$\rho_{n+1} = \underset{\rho}{\text{argmin}} \parallel \mathscr{A}(\rho) - \text{y} \parallel_2^2 + \beta \parallel \hat{\rho} - z_n \parallel_F^2 \qquad (10)$$

$$z_{n+1} = \underset{\mathbf{z}}{\text{argmin}} \parallel \hat{\rho}_{n+1} - \mathbf{z} \parallel_F^2 + \frac{\lambda}{\beta} \parallel \mathbf{T(z)} \parallel_* . \qquad (11)$$

We now borrow from the literature [24], [25] and majorize the nuclear norm term in (11) as

$$\parallel \mathbf{T(z)} \parallel_* \leq \parallel \mathbf{T(z)Q} \parallel_F^2 , \qquad (12)$$

where the weight matrix is specified by

$$\mathbf{Q} = \left[\mathbf{T}^H(\mathbf{z})\mathbf{T}(\mathbf{z}) + \epsilon \mathbf{I} \right]^{-1/4} \qquad (13)$$

Here, $\mathbf{I}$ is the identity matrix. Similar majorization strategies were used in the work [8]. With the majorization in (12), the $\mathbf{z}$-subproblem in (11) would involve the alternation between

$$\mathbf{z}_{n+1} = \underset{\mathbf{z}}{\text{argmin}} \parallel \hat{\rho}_{n+1} - \mathbf{z} \parallel_F^2 + \frac{\lambda}{\beta} \parallel \mathbf{T(z)Q} \parallel_F^2 \qquad (14)$$

and the update of the $\mathbf{Q}$ using (13). Thus the IRLS reformulation of the MUSSELS scheme would alternate between (10), (14), and (13) as summarized in Algorithm 1. The matrix $\mathbf{Q}$ may be viewed as a surrogate for the nullspace of $\mathbf{T(z)}$ as shown in the work [8]. The $\mathbf{Q}$ matrix at each iteration is estimated based on the previous iterate of $\mathbf{z}$. The update step (14) can be interpreted as finding an approximation of $\hat{\rho}_{n+1}$ from the signal subspace.

Algorithm 1

Summary of the IRLS-MUSSELS algorithm

Require: ρ_0, z_0

Ensure: ρ_{n+1}

1: **for** $n = 1$ to max_Iterations **do**

2: ρ_{n+1} = solve (10) using conjugate gradient

3: z_{n+1} = solve (14) using conjugate gradient.

4: $\mathbf{Q}_{n+1} = [\mathbf{T}^H(\mathbf{z}_{n+1})\mathbf{T}(\mathbf{z}_{n+1}) + \epsilon\mathbf{I}]^{-1/4}$

5: **end for**

6: **return** ρ_{n+1}

B. Interpretation of IRLS- MUSSELS as an iterative denoiser

We now focus on the term $\|\mathbf{T(z)Q}\|^2 = \sum_i \|\mathbf{T(z)q}_i\|^2$ in (14); $\mathbf{q}_i$ are the columns of $\mathbf{Q}$ representing nullspace vectors of $\mathbf{T(z)}$ similar vector $\mathbf{s}$ in Eq. (6). We note that the matrix-

vector product $\mathbf{T}(\mathbf{z})\mathbf{q}_i$ corresponds to the multichannel convolution of $\mathbf{z}$ with the columns of $\mathbf{Q}$, specified by $\mathbf{q}_i$. We split each column $\mathbf{q}_i$ into sub-filters $\mathbf{q}_{ij}$ to obtain

$$
Q = \left[\underbrace{\begin{bmatrix} \mathbf{q}_{11} \\ \vdots \\ \mathbf{q}_{1N} \end{bmatrix}}_{\mathbf{q}_1} \underbrace{\begin{bmatrix} \mathbf{q}_{21} \\ \vdots \\ \mathbf{q}_{2N} \end{bmatrix}}_{\mathbf{q}_2} \underbrace{\begin{bmatrix} \mathbf{q}_{11} \\ \vdots \\ \mathbf{q}_{1N} \end{bmatrix}}_{\mathbf{q}_N} \right]
\tag{15}
$$

where each $\mathbf{q}_{ij}$ is of length $|\Lambda|$. Note that $\mathbf{z} = [\mathbf{z}_1| \; ... \; |\mathbf{z}_N]$ is the multishot data. This allows us to rewrite the multichannel convolution

$$
\mathbf{T}(\mathbf{z})\mathbf{q}_i = \mathbf{H}_\Lambda^\Gamma(\mathbf{z}_1)\mathbf{q}_{i1} + ..\mathbf{H}_\Lambda^\Gamma(\mathbf{z}_N)\mathbf{q}_{iN}....
\tag{16}
$$

as the sum of convolutions of $\mathbf{z}_j$ with $\mathbf{q}_{i,j}$. Due to the commutativity of convolution $h * g = g * h$, each term in (16) can be re-expressed as

$$
\mathbf{H}_\Lambda^\Gamma(\mathbf{g})\mathbf{h} = S(\mathbf{h})\mathbf{g},
\tag{17}
$$

where $S(\mathbf{h})$ is an appropriately[2] sized block Hankel matrix constructed from the zero-filled entries of $\mathbf{h}$. We use this relation to rewrite

$$
\mathbf{T}(\mathbf{z})\mathbf{Q} = \underbrace{\begin{bmatrix} S(\mathbf{q}_{11}) & S(\mathbf{q}_{12}) & ... & S(\mathbf{q}_{1N}) \\ \vdots & ... & & \vdots \\ S(\mathbf{q}_{N1}) & S(\mathbf{q}_{12}) & ... & S(\mathbf{q}_{NN}) \end{bmatrix}}_{\mathbf{G(Q)}} \underbrace{\begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_N \end{bmatrix}}_{\mathbf{z}}.
$$

We note that $\mathbf{G(Q)z}$ corresponds to the multichannel convolution of $\mathbf{z}_1, ..., \mathbf{z}_N$ with the filterbank having filters $\mathbf{q}_{i,j}$. With this reformulation, (14) is simplified as

$$
\mathbf{z}_{n+1} = \underset{\mathbf{z}}{\mathrm{argmin}} \; \| \hat{\rho}_n - \mathbf{z} \|_F^2 + \frac{\lambda}{\beta} \| \mathbf{G(Q)z} \|_F^2 .
\tag{18}
$$

Differentiating the above expression and setting it equal to zero, we get

$$
z_{n+1} = \left(I + \frac{\lambda}{\beta} G(Q)^H G(Q) \right)^{-1} \rho_{n+1} .
$$

One may use a numerical solver to determine $\mathbf{z}_{n+1}$. An alternative is to solve this step approximately using the matrix inversion lemma, assuming $\lambda \ll \beta$:

---

[2] The size of the matrix is $|\Gamma| - |\Gamma \ominus \Lambda| \times |\Gamma|$ such that (17) holds. Here, Γ is the size of the image, and Λ is the size of the filter. $\ominus$ refers to the set erosion operator as defined in the work [8].

$$\mathbf{z}_{n+1} \approx \left[\mathbf{I} - \frac{\lambda}{\beta} \mathbf{G}(\mathbf{Q})^H \mathbf{G}(\mathbf{Q}) \right] \hat{\rho}_{n+1}$$

$$= \hat{\rho}_{n+1} - \frac{\lambda}{\beta} \mathbf{G}(\mathbf{Q})^H \mathbf{G}(\mathbf{Q}) \hat{\rho}_{n+1} .$$

(19)

We note that $G(\mathbf{Q})$ can be viewed as a single layer convolutional filter bank, while multiplication by $G(\mathbf{Q})^H$ can be viewed as flipped convolutions (deconvolutions in deep learning context) with matching boundary conditions. Note that neither of the above layers have any non-linearities. Thus, (19) can be thought of as a residual block, which involves the convolution of the multishot signals $\hat{\rho}_n$ with the columns of $\mathbf{Q}$, followed by deconvolution as shown in Fig. 2(a). As discussed before, the filters specified by the columns of $\mathbf{Q}$ are surrogates for the nullspace of $\mathbf{T}(\hat{\rho})$. Thus, the update (19) can be thought of as removing the components of $\hat{\rho}_n$ in the nullspace and projecting the data to the signal subspace, which may be viewed as a *sophisticated denoiser*, as shown in Fig. 2(a).

The IRLS-MUSSELS scheme [7], as summarized in Fig. 2, provides state-of-the-art results. However, note that the filters specified by the columns of $\mathbf{Q}$ are estimated for each diffusion direction by using Algorithm 1, which has high computational complexity, especially in the context of diffusion-weighted imaging, where several directions need to be estimated for each slice.

C. MoDL-MUSSELS Formulation

To minimize the computational complexity of the IRLS-MUSSELS, we propose learning a non-linear *denoiser* from exemplar data rather than learning a custom denoising block specified by $\left[I - \frac{\lambda}{\beta} G(Q)^H G(Q) \right]$ for each direction and slice. We hypothesize that the non-linearities in the network, as well as the larger number of filter layers, can facilitate the learning of a generalizable model from the exemplar data. This framework may be viewed as a multishot extension of the MoDL [9] approach. The cost function associated with the network is

$$\underset{\rho}{\operatorname{argmin}} \ \| \mathscr{A}(\rho) - \mathbf{y} \|_2^2 + \lambda_1 \| \mathscr{N}_k(\rho) \|_2^2 .$$

(20)

Here, $N_k(\rho)$ is a non-linear residual convolutional filterbank working in the Fourier domain, with

$$\mathscr{N}_k(\rho) = \rho - \mathscr{D}_k(\rho) .$$

(21)

$D_k(\rho)$ can be thought of as a multichannel CNN in the Fourier domain. The image domain input ρ is first transformed to k-space as $\hat{\rho}$, then passes through the k-space model, and, then transformed back to the image domain. Figure 3(a) shows the proposed M-layer CNN architecture. The overall k-space MoDL-MUSSELS network architecture is shown in Fig.

3(b), which solves Eq. (20). Unlike IRLS-MUSSELS in Fig. 2, the parameters of this network are not updated within the iterations and are learned from the exemplar data.

D. Hybrid MoDL-MUSSELS Regularization

A key benefit of the MoDL framework over direct inversion methods is the ability to exploit different kinds of priors, as shown in our prior work [13]. The IRLS-MUSSELS and the MoDL-MUSSELS schemes exploit the multichannel convolution relations between the k-space data. By contrast, we relied on an image domain convolutional neural network in earlier work [9] to exploit the structure of patches in the image domain. Note that this structure is completely complementary to the multichannel convolution relations. We now propose to jointly exploit both the priors as follows:

$$\underset{\rho}{\text{argmin}} \parallel \mathcal{A}(\rho) - \text{y} \parallel_2^2 + \lambda_1 \parallel \mathcal{N}_k(\rho) \parallel_2^2 + \lambda_2 \parallel \mathcal{N}_I(\rho) \parallel_2^2, \tag{22}$$

where N_k is the same prior as in (20), while N_I is an image space residual network of the form $N_I(\rho) = \rho - D_I(\rho)$. Here, D_I is an image domain CNN as in earlier work [9]. The problem (22) can be rewritten as

$$\underset{\rho}{\text{argmin}} \parallel \mathcal{A}(\rho) - \mathbf{y} \parallel_2^2 + \lambda_1 \parallel \rho - \mathcal{D}_k(\rho) \parallel_2^2 + \lambda_2 \parallel \rho - \mathcal{D}_I(\rho) \parallel_2^2.$$

By substituting $\eta = D_k(\rho)$, and $\zeta = D_I(\rho)$, an alternating minimization-based solution to the above problem iterates between the following steps:

$$\rho_{n+1} = \left(\mathcal{A}^H \mathcal{A} + \lambda_1 \mathcal{I} + \lambda_2 \mathcal{I}\right)^{-1} \left(\mathcal{A}^H \mathbf{y} + \lambda_1 \eta + \lambda_2 \zeta\right) \tag{23}$$

$$\eta_{n+1} = \mathcal{D}_k(\rho_{n+1}) \tag{24}$$

$$\zeta_{n+1} = \mathcal{D}_I(\rho_{n+1}). \tag{25}$$

The above solution results in the hybrid MoDL-MUSSELS architecture shown in Fig. 4. Note that this alternating minimization scheme is similar to the plug-and-play priors [26] widely used in inverse problems. The main exception is that we train the resulting network in an end-to-end fashion. Note that, unlike the plug-and-plug denoisers that learn the image manifold, the network D_k is designed to exploit the redundancies between the multiple shots resulting from the annihilation relations. This non-linear network is expected to project the multichannel k-space data orthogonal to the nullspaces of the multichannel Hankel matrices. The regularization parameters λ_1 and λ_2 control the contribution of the k-space network and the image-domain network, respectively. During experiments we kept the values of $\lambda_1 = 0,01$, $\lambda_2 = 0.05$ fixed. However, it can be noted that these values can be made trainable as in the MoDL [9].

IV. EXPERIMENTS

We perform several experiments to validate different aspects of the proposed model, such as, the benefits of the recursive network, the impact of regularization, robustness to outliers, comparison with existing deep learning models such as U-NET [27], and comparison with a model-based technique P-MUSE [22].

A. Dataset Description

In vivo data were collected from healthy volunteers at the University of Iowa in accordance with the Institutional Review Board recommendations. The imaging was performed on a GE MR750W 3T scanner using a 32-channel head coil. A Stejskal-Tanner spin-echo diffusion imaging sequence was used with a 4-shot EPI readout. A total of 60 diffusion gradient direction measurements were taken with a b-value of 700 s/mm^2. The relevant imaging parameters were FOV = 210×210 mm, matrix size = 256×152 with partial Fourier oversampling of 24 lines, slice thickness = 4 mm and TE = 84 ms. Data were collected from 7 subjects.

The training dataset constituted a total of 68 slices, each having 60 directions and 4 shots, from 5 subjects. The validation was performed on 6 slices of the 6th subject, whereas testing was carried out on 5 slices of the 7th subject. Thus, a total of 4080, 360, and 300 complex images each having size 256×256×4 (rows×columns×shots) were used for training, validation, and testing, respectively.

To perform quantitative comparisons, we also made use of simulated data with high SNR. For this purpose, we utilized a subset of pre-processed, relatively high-SNR diffusion dataset from the human connectome project [28]. We extracted 15 volumes and 20 slices from 100 subjects, which resulted in 30,000 magnitude images of size 145×174. We prepared a dataset of 23,000 training images, 3,000 validation images, and 3,000 test images. We simulated the sensitivity maps using Walsh algorithm [29]. To simulate the multishot data with phase errors, we multiplied each magnitude image with synthetically generated random bandlimited phase errors using the image formation model in Eq. (3). Gaussian noise of varying amounts of standard deviation σ was added to the phase-corrupted k-space data. The k-space data was under-sampled to generate the multishot data.

B. Multi-coil forward model

All of the model-based schemes used in this study (MUSE, MUSSELS, MoDL-MUSSELS) rely on a forward model that mimics the image formation. We implement this forward model as described in (1) and (3). The raw dataset consists of 32 channels. We reduce the data to four virtual channels using singular value decomposition (SVD) of the non-diffusion weighted (b0) image. The coil sensitivity maps of these four virtual channels were estimated using ESPIRIT [30]. The same channel combination weights were used to reduce the diffusion-weighted MRI data to four coils.

C. Quantitative metrics used in experiments

The reconstruction quality is measured using the structure similarity index (SSIM) [31] and peak signal-to-noise ratio (PSNR). The PSNR is defined as

$$PSNR(x, y) = 10 * \log_{10}\left(\frac{\max(x)^2}{MSE(x, y)}\right)$$

where MSE is the mean-square-error between **x** and **y**. The final PSNR/SSIM value is estimated by the average of the PSNR/SSIM of individual shots.

D. Algorithms used for comparison

We compare the proposed scheme against IRLS-MUSSELS [7],P-MUSE [22], and a solution based on U-NET [27]. The IRLS-MUSSELS is a modification of the MUSSELS algorithm [6]. Specifically, the modification involve an IRLS based implementation instead of iterative shirnkage algorithm in [6], which results in a faster implementation. Moreover, it also includes an additional conjugate symmetry constraint in addition to the annihilation relations between the shots that is exploited in the MUSSELS method [6]. We refer the readers to [7], which shows that the addition of the conjugate symmetry constraint reduces blurring and results in sharper images compared to the original MUSSELS method [6] for partial Fourier acquisitions. In the results section, the IRLS-MUSSELS is referred to as simply IRLS-M.

P-MUSE [22] is a two-step algorithm that first estimates the motion-induced phase using the SENSE [15] reconstruction and the total-variation denoising. With the knowledge of the phase errors, it recovers the images using a regularized optimization with (3) as the forward model. The P-MUSE algorithm [22] has three parameters $\lambda_1 = 0.01$, $\lambda_2 = 0.01$, and the number of iterations = 40. The parameters λ_1 and λ_2 control the total variation regularization during phase estimation and reconstruction, respectively. We searched over the parameters to yield the best possible reconstruction.

We extended the U-NET [27] model for the multishot diffusion MR image reconstruction. The number of convolution layers, the feature maps in each layer, and the filter size were kept the same as in [27]. The input to the extended U-NET model was the concatenation of the real and imaginary parts of phase-corrupted coil-combined complex 4 shots images. The IRLS-MUSSELS [7] reconstructions were used as the ground truth for the training of the deep learning models on experimental data. We trained the network in the image domain with 1,000 epochs for 13 hours using the Adam [32] optimizer.

We also performed a comparison between the k-space MoDL-MUSSELS formulation in section III-C and the hybrid MoDL-MUSSELS formulation in section III-D. We refer to the former as the k-space network and the latter as the hybrid network. To perform a fair comparison between hybrid and k-space networks, the number of parameters in the k-space network was kept the same as that of the hybrid model by increasing the number of feature maps in the convolution layers.

E. Network architecture and training

In this work, we trained an 8-layer CNN having convolution filters of size 3×3 in each layer. Each layer comprises a convolution, followed by ReLU, except the last layer, which consists of a 1×1 convolution as shown in Fig. 5. The real and imaginary components of the complex 4 shots data were considered as channels in the residual learning CNN architecure, whereas the data-consistency block worked explicitly with complex data.

The proposed network architecture, as shown in Fig. 4, was unfolded for three iterations, and the end-to-end training was performed for 100 epochs. The input to the unfolded network is the zero-filled complex data from the four shots, which corresponds to $A^H \mathbf{y}$, while the network outputs the fully sampled complex data for the four shots. The proposed MoDL-MUSSELS architecture combines the data from the four shots using the sum-of-squares approach. The network weights were randomly initialized using Xavier initialization and shared between iterations. The network was implemented using the TensorFlow library in Python 3.6 and trained using the NVIDIA P100 GPU. The conjugate-gradient optimization in the DC step was implemented as a layer operation in the TensorFlow library as described in the work [9]. We utilized the mean-square-error as the loss function during training. The total network training time of the network was around 37 hours.

The plot in Fig. 6 shows training loss decays smoothly with epochs. It can be noted that the loss on the validation dataset also has overall decaying behavior, which implies that the trained model did not over-fit the dataset. The model-based framework has considerably fewer parameters than direct inverse methods and hence requires far fewer training data to achieve good performance, as seen from the experiments in the previous work [9].

V. RESULTS

A. Comparisons using simulated data

Table I summarizes the quantitative results (PSNR and SSIM values) obtained from the simulated data in Section IV-A. Specifically, we quantitatively compare the reconstructions provided by the five algorithms, while varying the noise levels. We did not perform the training of the deep learning methods (k-space, U-NET, and hybrid) for different noise levels but instead utilized the same model trained for a single noise level ($\sigma = 0.001$). We adjusted the parameters of the P-MUSE and IRLS-MUSSELS algorithms for different noise levels to get the best average results. The average performance of the U-NET is lower than all other methods since the U-NET does not have an explicit data-consistency term like the other methods. It is evident from the graphs in Fig. 7 that proposed hybrid method performs better than other methods on all individual slices and directions of a test subject.

Figure 8 shows an example set of images reconstructed using the five methods for this simulated data. For comparison, the uncorrected image and the error maps for all the reconstructions, compared to the ground truth image are also provided. It is evident from the error maps that the proposed hybrid model has the least error among the methods compared. Figure 8(e) shows that the k-space network is able to compensate for phase errors of multishot data. The addition of image-domain regularization in the hybrid model further improves the reconstructions in Fig. 8(f). We note that the image domain network exploits

the manifold structure of patches, which serves as a strong prior that the k-space network has difficulty capturing.

B. Robustness to outliers

We further performed an experiment to determine the robustness of the proposed MoDL-MUSSELS approach against outliers. In particular, we simulated a lesion image by increasing the intensity at a few pixels as indicated by an arrow in Fig. 9(a). This image was passed through the existing trained model. It is observed from Fig. 9(b) that simulated lesion was preserved by the proposed method. Note that the training dataset did not have any lesion images and we did not simulate the lesion images during training. The robustness of the algorithm to such outliers can be attributed to the fact that the algorithm relies on k-space and q-space deep-learning networks with small receptive fields, unlike direct inversions methods that rely on large receptive fields. Hence, the proposed scheme learns only to exploit local redundancies in k-space and q-space and does not memorize whole images.

C. Comparison of reconstruction time

Table II compares the time taken to reconstruct the entire testing dataset for various methods. It is noted that the computational complexity of the MoDL-MUSSELS is around 28-fold lower than the IRLS-MUSSELS. Note that IRLS-MUSSELS estimates the optimal linear filter bank from the measurements itself, which requires significantly many iterations. By contrast, MoDL-MUSSELS pre-learns non-linear network weights. The quite significant speed increase directly follows from the significantly fewer number of iterations. Note that we rely on a conjugate-gradient algorithm to enforce data consistency specified by (23). Also note that solving (23) exactly as opposed to the use of steepest gradient steps at each iteration would require more unrolling steps, thus diminishing the gain in speedup. The greatly reduced runtime is expected to facilitate the deployment of the proposed algorithm on clinical scanners.

D. Impact of iterations on image quality

Figure 10 shows the impact of the number of iterations in the iterative algorithm described in (23)–(25). Specifically, we unrolled the iterative algorithm for the different numbers of iterations and compared the performance of the resulting networks. We used the hybrid model due to its improved performance. The parameters of both the k-space and image-space networks were shared across iterations. Specifically, MoDL-MUSSELS uses three iterations of alternating-minimization, with five iterations of CG within each alternating step. The IRLS-MUSSELS uses five iterations of both outer loop as well as CG step. The images in Fig. 10 each correspond to a specific direction and slice in the testing dataset. We note that the contrast and details in the image improved with iterations, as did the visualization of some features, as shown in the zoomed portions.

E. Comparisons on experimental data

Next we compare the performance of the proposed method to reconstruct experimental data. Figure 11 shows the reconstructions offered by the different algorithms. A separate network was trained with the experimental data utilizing the IRLS-MUSSELS as the ground truth.

While this comparison may not be fair to P-MUSE, we used this approach since the main goal is to validate the MoDL-MUSSELS and U-NET which relied on IRLS-MUSSELS results for training. As evident from the error maps, the U-NET reconstructions appear less blurred, but it seems to miss some key features highlighted by yellow boxes. The hybrid method provides good results comparable to that of IRLS-MUSSELS.

To further validate the reconstruction accuracy of all the DWIs corresponding to the test slice, we performed a tensor fitting using all the DWIs and compared the resulting fractional anisotropy (FA) maps and the fiber orientation maps. For this purpose, the DWIs reconstructed using various methods were fed to a tensor fitting routine (FDT Toolbox, FSL). The FA maps were computed from the fitted tensors, and the direction of the primary eigenvectors of the tensors was used to estimate the fiber orientation. The FA maps generated using the various reconstruction methods are shown in Fig. 12, which has been color-coded based on the fiber direction. It is noted that these fiber directions reconstructed by the IRLS-MUSSELS method and the MoDL-MUSSELS match the true anatomy known for this brain region from a diffusion tensor imaging (DTI) white matter atlas (http://www.dtiatlas.org).

VI. DISCUSSION

We observe that the IRLS-MUSSELS reconstructions on experimental data are noisy. This noisy ground truth training data causes fuzziness in the training loss, which translates to the slight blurring in MoDL-MUSSELS reconstructions in Fig. 11. Note that the MoDL-MUSSELS reconstructions in the simulated data experiments in Fig. 8 are less blurred. The dependence of the final image quality on the training data is a limitation of the current work, especially in the multishot diffusion setting where noise-free training data is difficult to acquire. We plan to experiment with denoising strategies as well as the acquisition of training data with multiple averages to mitigate these problems. Further, we note from Fig. 11 that the reconstructions provided by the MoDL-MUSSELS appear less noisy and are visually more appealing than the noisy ground truth obtained using the IRLS-MUSSELS. This behavior may be attributed to the convolutional structure of the network, which is known to offer implicit regularization [33].

In this work, we utilized an eight-layer neural network, as shown in Fig. 5. However, the proposed MoDL-MUSSELS architecture in Fig. 4 is not constrained by choice of network. Any network architecture (e.g., U-NET) may be used instead. It is possible that the results can improve by utilizing more sophisticated network architecture. Further, it can be noted that the proposed model architecture is flexible to allow different network architectures for image-space and k-space models. However, for the proof of concept, we used the same network architecture for both k-space and image space.

To avoid overfitting the model and reduce the training time, the proposed network in Fig. 4 was unfolded for three iterations before performing the joint training. The sharing of network parameters allows the network to be unfolded for any number of iterations without increasing the number of trainable parameters. In this work, we restricted our implementation to a three iteration setting. We note that the results may improve with more

outer iterations. However, increasing the outer iterations require more GPU memory during network training.

The deep learning blocks used in the proposed scheme map the noisy/artefact-prone N-shot data to the noise-free N-shot data. The size of the filters in the first and last layers of the deep learning blocks depend on the number of shots. Hence, the network needs to be retrained if the number of shots changes. Since the filters capture the annihilation relations between the shots, we do not anticipate the need to retrain the network if other parameters (e.g., image size, TR, TE, etc.) change. Finally, we note that the current method depends on the estimation of the coil sensitivities to recover the multishot data.

VII. CONCLUSIONS

We introduced a model-based deep learning framework termed MoDL-MUSSELS for the compensation of phase errors in multishot diffusion-weighted MRI data. The proposed algorithm alternates between a conjugate gradient optimization algorithm to enforce data consistency and multichannel convolutional neural networks (CNN) to project the data to appropriate subspaces. We rely on a hybrid approach involving a multichannel CNN in the k-space and another one in the image space. The k-space CNN exploits the annihilation relations between the shot images, while the image domain network is used to project the data to an image manifold. The weights of the deep network, obtained by unrolling the iterations in the iterative optimization scheme, are learned from exemplary data in an end-to-end fashion. The experiments show that the proposed scheme can yield reconstructions that are comparable to state-of-the-art methods while offering several orders of magnitude reduction in run-time.

ACKNOWLEDGMENT

REFERENCES

[1]. Le Bihan D, "Looking into the functional architecture of the brain with diffusion MRI," Nature Reviews Neuroscience, vol. 4, no. 6, p. 469, 2003. [PubMed: 12778119]

[2]. Ciccarelli O, Catani M, Johansen-Berg H, Clark C, and Thompson A, "Diffusion-based tractography in neurological disorders: concepts, applications, and future developments," The Lancet Neurology, vol. 7, no. 8, pp. 715–727, 2008. [PubMed: 18635020]

[3]. Charlton RA, Barrick T, McIntyre D, Shen Y, O'sullivan M, Howe F, Clark C, Morris R, and Markus H, "White matter damage on diffusion tensor imaging correlates with age-related cognitive decline," Neurology, vol. 66, no. 2, pp. 217–222, 2006. [PubMed: 16434657]

[4]. Moseley M, Cohen Y, Mintorovitch J, Chileuitt L, Shimizu H, Kucharczyk J, Wendland M, and Weinstein P, "Early detection of regional cerebral ischemia in cats: comparison of diffusion-and T2-weighted MRI and spectroscopy," Magnetic resonance in medicine, vol. 14, no. 2, pp. 330–346, 1990. [PubMed: 2345513]

[5]. Wu W and Miller KL, "Image formation in diffusion MRI: a review of recent technical developments," Journal of Magnetic Resonance Imaging, vol. 46, no. 3, pp. 646–662, 2017. [PubMed: 28194821]

[6]. Mani M, Jacob M, Kelley D, and Magnotta V, "Multi-shot sensitivity-encoded diffusion data recovery using structured low-rank matrix completion (MUSSELS)," Magnetic Resonance in Medicine, vol. 78, no. 2, pp. 494–507, 2017. [PubMed: 27550212]

[7]. Mani M, Aggarwal HK, Magnotta V, and Jacob M, "Improved reconstruction for high-resolution multi-shot diffusion weighted imaging," in https://arxiv.org/abs/1906.10178, 2019.

[8]. Ongie G and Jacob M, "A fast algorithm for convolutional structured low-rank matrix recovery," IEEE transactions on computational imaging, vol. 3, no. 4, pp. 535–550, 2017. [PubMed: 29911129]

[9]. Aggarwal HK, Mani MP, and Jacob M, "MoDL: Model based deep learning architecture for inverse problems," IEEE Trans. Med. Imag, vol. 38, no. 2, pp. 394–405, 2019.

[10]. Adler J and Öktem O, "Learned primal-dual reconstruction," IEEE Trans. Med. Imag, vol. 37, no. 6, pp. 1322–1332, 2018.

[11]. Yang Y, Sun J, Li H, and Xu Z, "Deep ADMM-Net for compressive sensing mri," in Advances in Neural Information Processing Systems 29, 2016, pp. 10–18.

[12]. Hammernik K, Klatzer T, Kobler E, Recht MP, Sodickson DK, Pock T, and Knoll F, "Learning a Variational Network for Reconstruction of Accelerated MRI Data," Magnetic resonance in Medicine, vol. 79, no. 6, pp. 3055–3071, 2017. [PubMed: 29115689]

[13]. Biswas S, Aggarwal HK, and Jacob M, "Dynamic MRI using model-based deep learning and storm priors: MoDL-SToRM," Magnetic Resonance in Medicine, vol. 82, no. 1, pp. 485–494, 2019. [PubMed: 30860286]

[14]. Chartrand R and Yin W, "Iteratively reweighted algorithms for compressive sensing," in IEEE International Conference on Acoustics, Speech and Signal Processing, 3 2008, pp. 3869–3872.

[15]. Pruessmann KP, Weiger M, Scheidegger MB, and Boesiger P, "SENSE: sensitivity encoding for fast MRI," Magnetic resonance in medicine, vol. 42, no. 5, pp. 952–962, 1999. [PubMed: 10542355]

[16]. Lee J, Han Y, and Ye JC, "k-space deep learning for reference-free EPI ghost correction," arXiv preprint arXiv:1806.00153, 2018.

[17]. Souza R and Frayne R, "A hybrid frequency-domain/image-domain deep network for magnetic resonance image reconstruction," arXiv preprint arXiv:1810.12473, 2018.

[18]. Wang S, Ke Z, Cheng H, Jia S, Leslie Y, Zheng H, and Liang D, "DIMENSION: Dynamic MR imaging with both k-space and spatial prior knowledge obtained via multi-supervised network training," arXiv preprint arXiv:1810.00302, 2018.

[19]. Eo T, Jun Y, Kim T, Jang J, Lee H-J, and Hwang D, "KIKI-net: cross-domain convolutional neural networks for reconstructing under-sampled magnetic resonance images," Magnetic resonance in medicine, vol. 80, no. 5, pp. 2188–2201, 2018. [PubMed: 29624729]

[20]. Akcakaya M, Moeller S, Weingartner S, and Ugurbil K, "Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: Database-free deep learning for fast imaging," Magnetic Resonance in Medicine, vol. 81, no. 4 2018, pp. 439–453, 2018. [PubMed: 30277269]

[21]. Chen N.-k., Guidon A, Chang H-C, and Song AW, "A robust multishot scan strategy for high-resolution diffusion weighted MRI enabled by multiplexed sensitivity-encoding (MUSE)," Neuroimage, vol. 72, pp. 41–47, 2013. [PubMed: 23370063]

[22]. Chu M-L, Chang H-C, Chung H-W, Truong T-K, Bashir MR, and Chen N.-k., "POCS-based reconstruction of multiplexed sensitivity encoded mri (POCSMUSE): A general algorithm for reducing motionrelated artifacts," Magnetic Resonance in Medicine, vol. 74, no. 5, pp. 1336–1348, 2015. [PubMed: 25394325]

[23]. Morrison RL, Jacob M, and Do MN, "Multichannel estimation of coil sensitivities in parallel MRI," in IEEE Int. Symp. Bio. Imag. IEEE, 2007, pp. 117–120.

[24]. Mohan K and Fazel M, "Iterative reweighted algorithms for matrix rank minimization," Journal of Machine Learning Research, vol. 13, no. Nov, pp. 3441–3473, 2012.

[25]. Fornasier M, Rauhut H, and Ward R, "Low-rank matrix recovery via iteratively reweighted least squares minimization," SIAM Journal on Optimization, vol. 21, no. 4, pp. 1614–1640, 2011.

[26]. Chan SH, Wang X, and Elgendy OA, "Plug-and-Play ADMM for Image Restoration: Fixed Point Convergence and Applications," IEEE Trans. on Comp. Imag, vol. 3, no. 1, pp. 84–98, 2017.

[27]. Ronneberger O, Fischer P, and Brox T, "U-NET: Convolutional networks for biomedical image segmentation," in International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) Springer, 2015, pp. 234–241.

[28]. Van Essen DC, Smith SM, Barch DM, Behrens TE, Yacoub E, Ugurbil K, Consortium W-MH et al., "The wu-minn human connectome project: an overview," Neuroimage, vol. 80, pp. 62–79, 2013. [PubMed: 23684880]

[29]. Walsh DO, Gmitro AF, and Marcellin MW, "Adaptive reconstruction of phased array mr imagery," Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine, vol. 43, no. 5, pp. 682–690, 2000.

[30]. Uecker M, Lai P, Murphy MJ, Virtue P, Elad M, Pauly JM, Vasanawala SS, and Lustig M, "ESPIRiT - An eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA," Magnetic Resonance in Medicine, vol. 71, no. 3, pp. 990–1001, 2014. [PubMed: 23649942]

[31]. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP et al., "Image quality assessment: from error visibility to structural similarity," IEEE Trans. Image Process, vol. 13, no. 4, pp. 600–612, 2004. [PubMed: 15376593]

[32]. Kingma DP and Ba JL, "Adam: a Method for Stochastic Optimization," International Conference on Learning Representations 2015, pp. 1–15, 2015.

[33]. Ulyanov D, Vedaldi A, and Lempitsky V, "Deep image prior," in IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9446–9454.
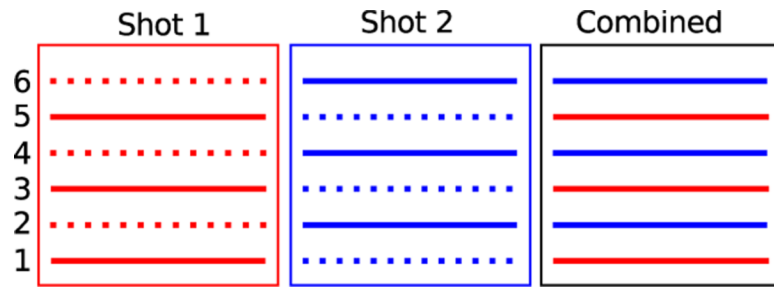
Fig. 1.
Demonstration of multishot EPI acquisition employing multiple excitations and readouts. The first radio-frequency (RF) excitation and diffusion sensitization are followed by a k-space readout by shot 1 that samples k-space lines 1, 3, and 5. The second RF excitation and diffusion sensitization are followed by a k-space readout by shot 2 capturing lines 2, 4, and 6. The combined data corresponds to the fully sampled k-space.
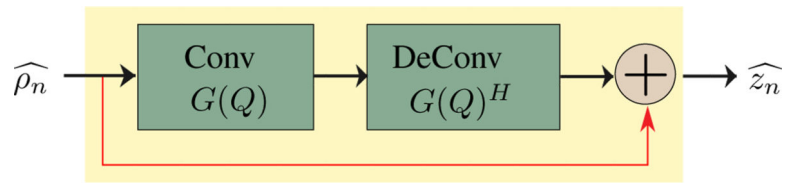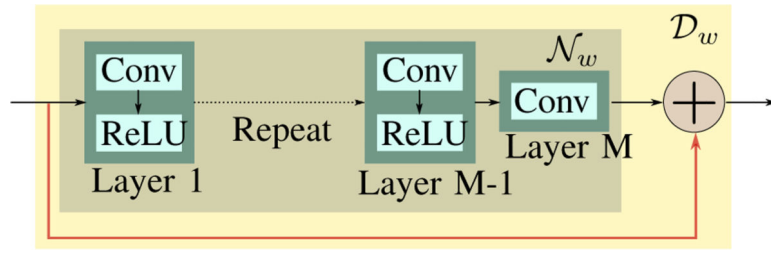
(a) Representation of Eq. (19) as the IRLS-MUSSELS denoiser $\mathcal{D}_w$.
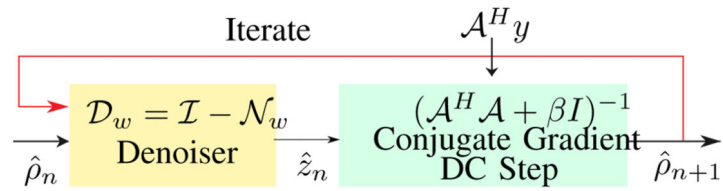
(b) The IRLS-MUSSELS algorithm

Fig. 2.
(a). The interpretation of Eq. (19) as a convolution-deconvolution network. (b) The IRLS-MUSSELS iterates between (19) and (10). The data consistency (DC) step represents the solution of Eq. (10).

(a) The M-layer CNN-based denoiser
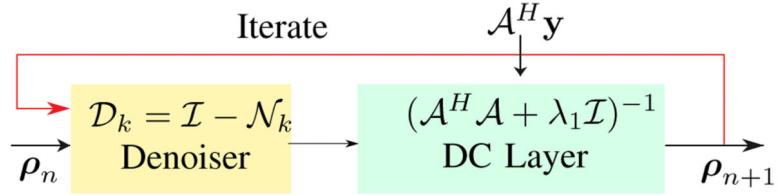
(b) Proposed k-space MoDL-MUSSELS architecture

Fig. 3.
The block diagram of the proposed k-space network architecture to solve Eq. (20). (a) The N_w block represents the deep learned noise predictor, and D_w is a residual learning block. (b) Here, the denoiser D_k is the M-layer network D_w that performs the k-space denoising.
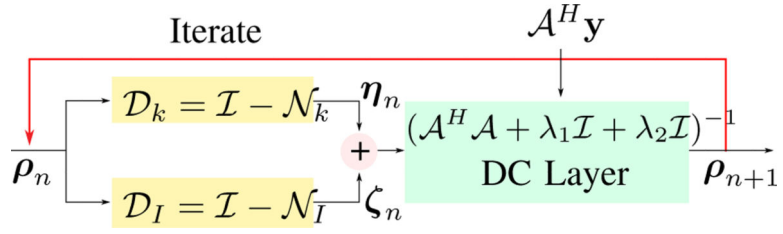
Fig. 4.
The proposed hybrid MoDL-MUSSELS architecture resulting from the alternating scheme shown in (23)–(25). Here the D_k and D_I blocks represent the k-space and the image-space denoising networks, respectively. The D_k and D_I networks have identical structures as shown in Fig 3(a). The learnable convolution weights are differnt for networks D_k and D_I but remain constant across iterations.
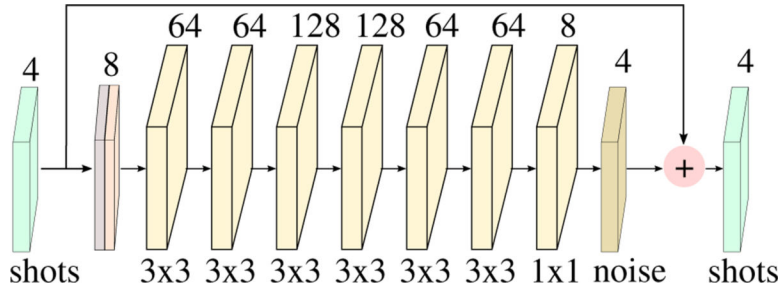
Fig. 5.
The specific M=8 layer residual learning CNN architecture used as D_k and D_I blocks in the experiments. The 4 shot complex data are the input and output of the network. The first layer concatenates the real and imaginary parts as 8 input features. The numbers on top of each layer represent the number of feature maps learned at that layer. We learn 3×3 filters at each layer except the last, where we learn 1×1 filter.
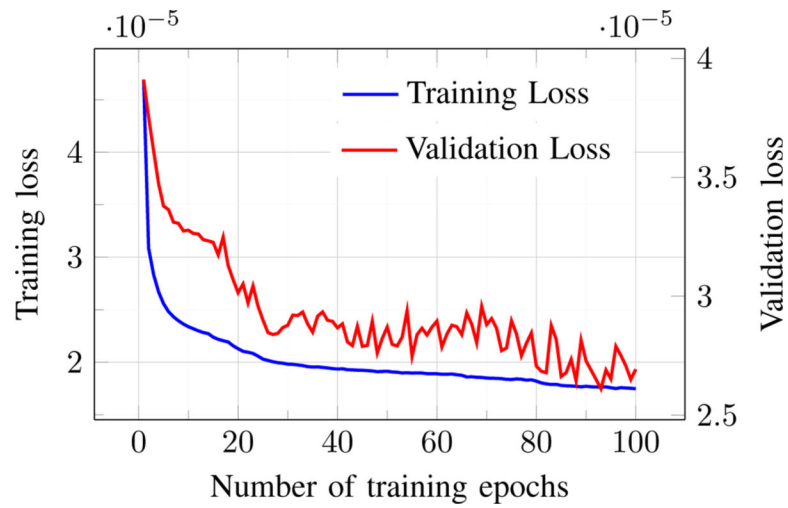
Fig. 6.
The decay of training and validation errors with epochs. Each epoch represents one sweep through the entire dataset. We note that both the losses decay with iterations. This suggests that the amount of training data is sufficient to train the parameters of the model.
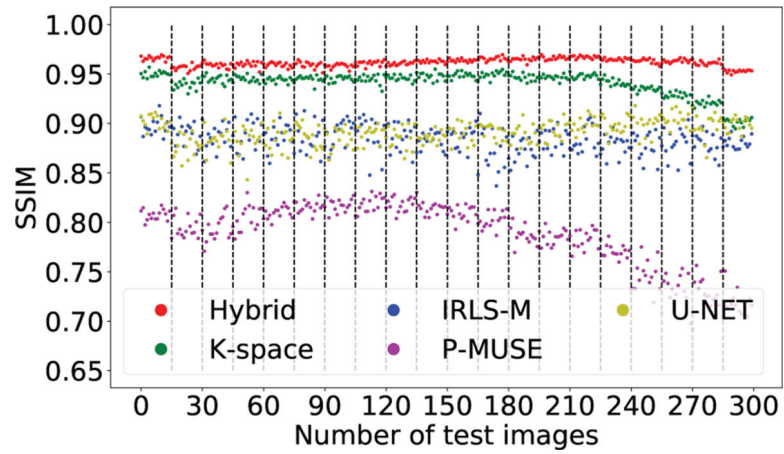
Fig. 7.

This plot compares the variation of the SSIM on all the slices and directions of one test subject from simulation dataset. The vertical lines seperate the different slices, i.e., first fifteen images are the directions corresponding to the first slice, and so on. There is a total of 20 slices, each having 15 different directions, resulting in a total of 300 images.
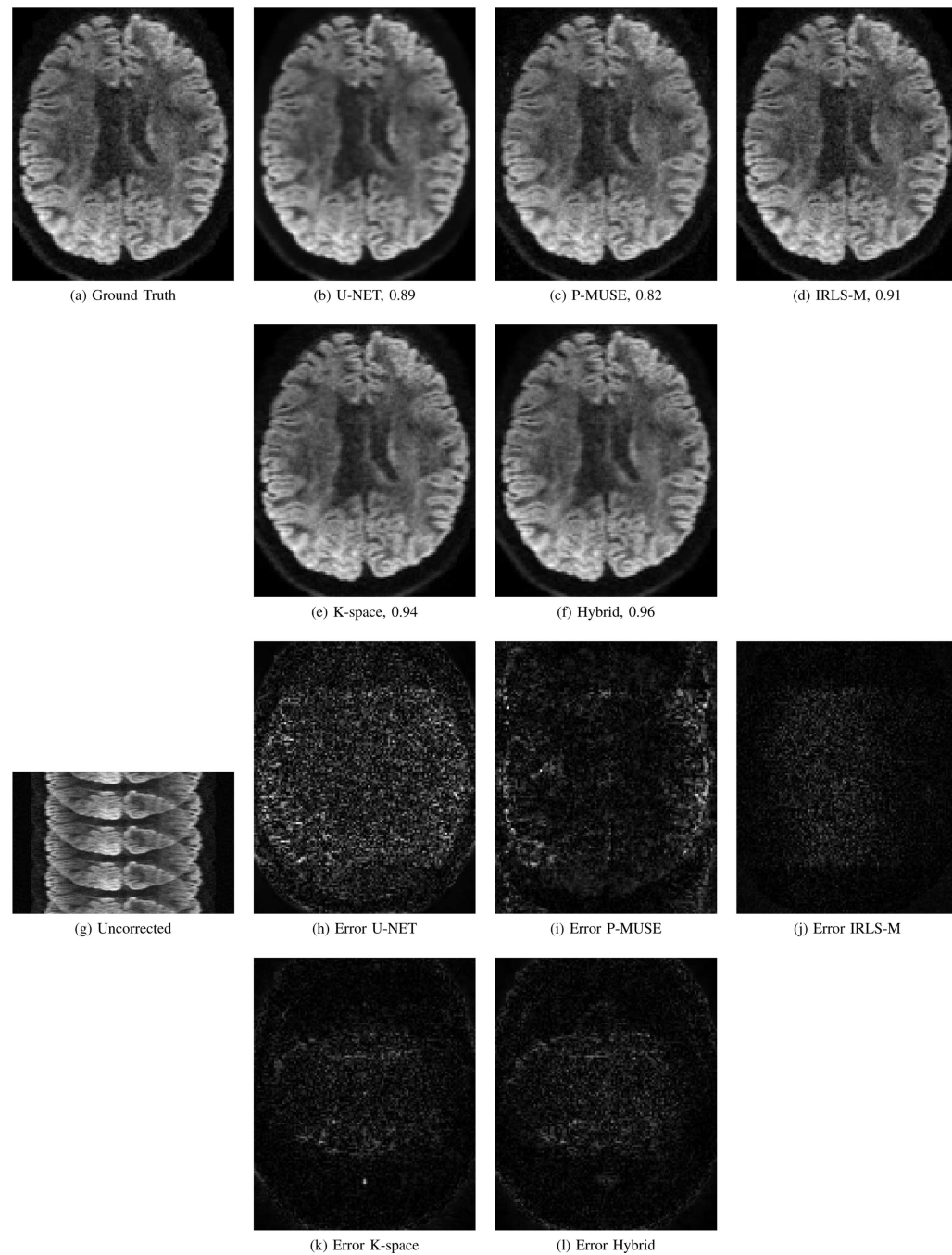
(a) Ground Truth   (b) U-NET, 0.89   (c) P-MUSE, 0.82   (d) IRLS-M, 0.91

(e) K-space, 0.94   (f) Hybrid, 0.96

(g) Uncorrected   (h) Error U-NET   (i) Error P-MUSE   (j) Error IRLS-M

(k) Error K-space   (l) Error Hybrid

Fig. 8.

Simulation results using the high-SNR data obtained from HCP. The reconstructed images and the corresponding error maps from five different algorithms are shown. Here, the ground truth is an image from the test dataset that was corrupted with phase errors of bandwidth 3×3 and noise standard deviation $\sigma = 0.001$ to simulate 4 shot acquisition. (g) shows the uncorrected image if we do not correct the phase errors during reconstruction. The numbers in the sub-captions represent the SSIM values.
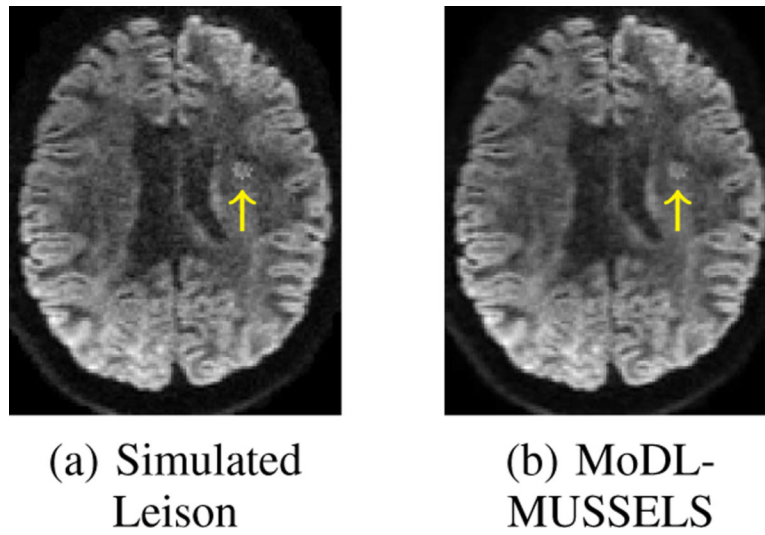
(a) Simulated Leison

(b) MoDL-MUSSELS

Fig. 9.
Lesion experiment. Arrow points to the location of Lesion. The proposed MoDL-MUSSELS method preserves the lesion.

(a)
IRLS-MUSSELS

(b) One iteration
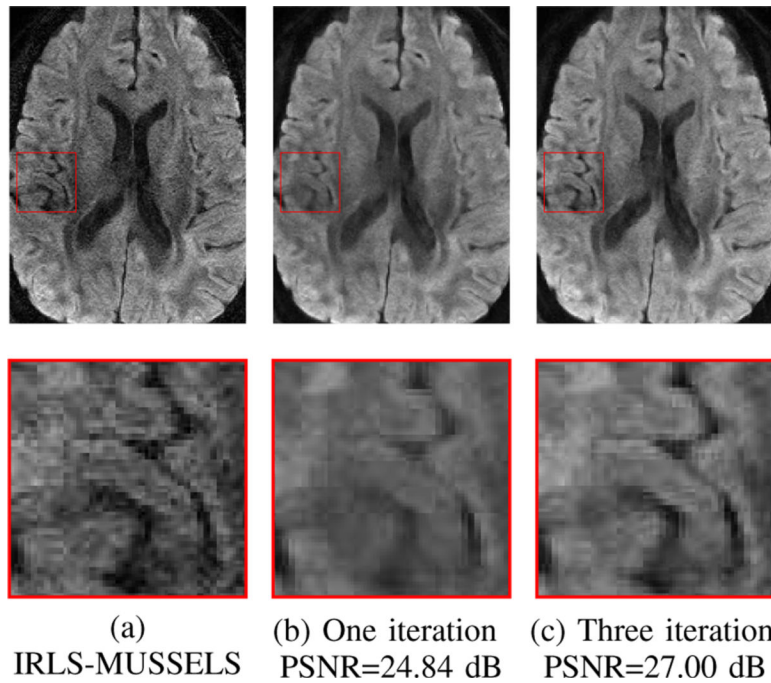PSNR=24.84 dB

(c) Three iteration
PSNR=27.00 dB

Fig. 10.
Effect of iterations on image quality. We observe that the quality of the reconstructions with the proposed MoDL-MUSSELS scheme improve with iterations. Specifically, the sharpness of the image and the contrast seem to improve with more iterations.
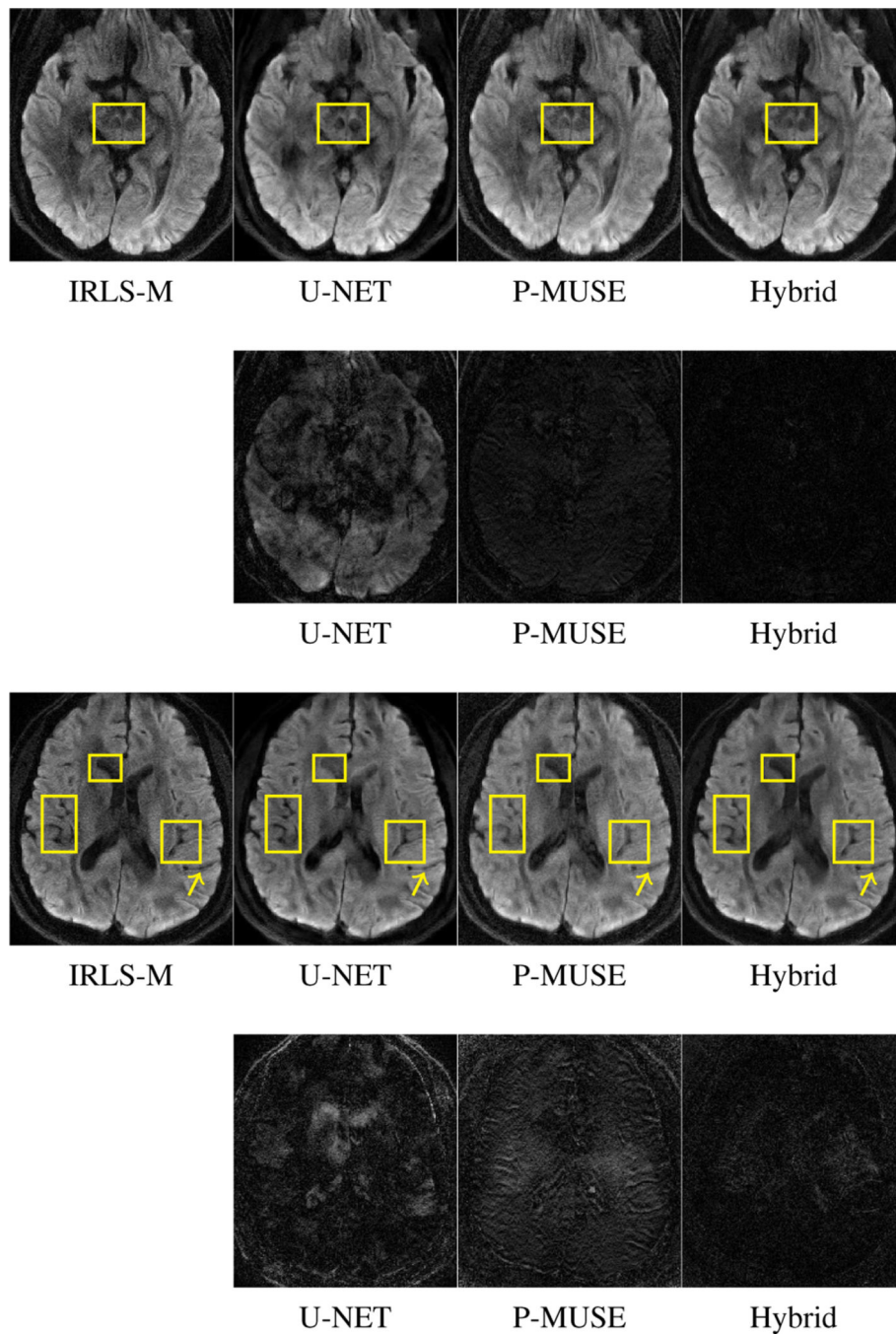
Fig. 11.
Reconstructions obtained using different algorithms on experimental partial-Fourier data. Row 1 and Row 3 shows reconstruction results from two different slices. We generated the error maps in rows 2 and 4 by considering MUSSELS reconstructions as ground truth. The yellow boxes highlight the differences.
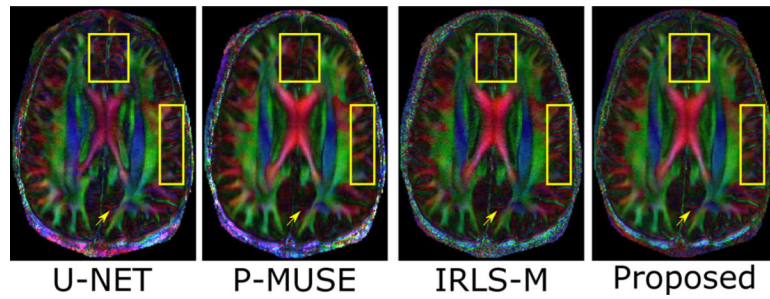
Fig. 12.
The fractional-anisotropy maps on a test dataset slice. These images are computed from the sixty directions of the slices, recovered using the respective algorithms. We note the proposed scheme provides less blurred reconstructions than P-MUSE, which are comparable with IRLS-MUSSELS.

TABLE I

THE PSNR (DB) AND SSIM VALUES OBTAINED BY FIVE METHODS ON THE TESTING DATASET WITH SIMULATED PHASES AND ADDED GAUSSIAN NOISE OF VARYING STANDARD DEVIATION σ. THE VALUES ARE REPORTED AS MEAN $\pm$ STANDARD DEVIATION.

Peak signal to noise ratio (dB)		
Noise (std)	$\sigma = 0.001$	$\sigma = 0.002$	$\sigma = 0.003$
U-NET	32.15 ± 2.12	29.98 ± 1.19	27.63 ± 0.82
P-MUSE	34.08 ± 2.31	31.68 ± 2.21	29.19 ± 1.84
IRLS-M	38.81 ± 1.98	36.21 ± 1.32	32.43 ± 1.33
K-space	40.02 ± 1.18	36.92 ± 0.96	34.69 ± 1.38
Hybrid	40.59 ± 1.87	37.37 ± 1.56	35.40 ± 1.36
Structural similarity index		
U-NET	0.89 ± 0.01	0.82 ± 0.02	0.73 ± 0.03
P-MUSE	0.79 ± 0.03	0.69 ± 0.04	0.63 ± 0.05
IRLS-M	0.88 ± 0.01	0.83 ± 0.01	0.72 ± 0.03
K-space	0.94 ± 0.01	0.89 ± 0.02	0.84 ± 0.03
Hybrid	0.96 ± 0.00	0.94 ± 0.01	0.92 ± 0.01

TABLE II

TIME TO RECONSTRUCT ALL FIVE SLICES OF THE TEST SUBJECT. EACH SLICE HAD 60 DIRECTIONS, 4 SHOTS, AND 32 COILS. IRLS-MUSSELS AND P-MUSE WERE RUNS ON CPU WITH PARALLEL PROCESSING.

Algorithm:	U-NET	P-MUSE	IRLS-M	MoDL-MUSSELS
Time (sec) :	7	632	1386	49