Dynamic MRI using model-based deep learning and SToRM priors: MoDL-SToRM

Sampurna Biswas, Hemant K. Aggarwal*, Mathews Jacob**

** Department of Electrical and Computer Engineering, The University of Iowa, Iowa*

January 8, 2019

Correspondence to:

Mathews Jacob

Department of Electrical and Computer Engineering

4016 Seamans Center

University of Iowa, IA 52242

Email: mathews-jacob@uiowa.edu

This work is supported by NIH 1R01EB019961-01A1.

Approximate word count : 2800

Number of figures & tables: 5

Abstract

Purpose: To introduce a novel framework to combine deep-learned priors along with complementary image regularization penalties to reconstruct free breathing & ungated cardiac MRI data from highly undersampled multi-channel measurements.

Methods: Image recovery is formulated as an optimization problem, where the cost function is the sum of data consistency term, convolutional neural network (CNN) denoising prior, and SmooThness regularization on manifolds (SToRM) prior that exploits the manifold structure of images in the dataset. An iterative algorithm, which alternates between denoising of the image data using CNN and SToRM, and conjugate gradients (CG) step that minimizes the data consistency cost is introduced. Unrolling the iterative algorithm yields a deep network, which is trained using exemplar data.

Results: The experimental results demonstrate that the proposed framework can offer fast recovery of free breathing and ungated cardiac MRI data from less than 8.2s of acquisition time per slice. The reconstructions are comparable in image quality to SToRM reconstructions from 42s of acquisition time, offering a five-fold reduction in scan time.

Conclusion: The results show the benefit in combining deep learned CNN priors with complementary image regularization penalties. Specifically, this work demonstrates the benefit in combining the CNN prior that exploits local and population generalizable redundancies together with SToRM, which capitalizes on patient specific information including cardiac and respiratory patterns. The synergistic combination is facilitated by the proposed framework.

Key words: model-based, learned prior, alternating minimization, subject specific prior, non-local prior, free breathing cardiac MR

INTRODUCTION

Breath-held cardiac cine MRI is a key component in cardiac MRI exams, which is used for the anatomical and functional assessment of the heart. Unfortunately, several subject groups (e.g. children and chronic obstructive pulmonary disease (COPD) patients (1)) cannot hold their breath and hence are excluded from breath-held MRI studies. In addition, breath-held protocols are also associated with long scan times. Several methods were introduced to reduce the breath-holding requirements or enable free breathing cardiac MRI protocols, including parallel MRI (2, 3), approaches that exploit the structure of x-f space (4–6), compressed sensing schemes (7, 8), low-rank methods (9, 10), blind compressed sensing (11, 12), motion compensated methods (13, 14), and kernel low-rank methods (15). Motivated by recent deep learning frameworks for static imaging applications (16–18), deep-learning based breath-held cardiac cine acceleration methods were also introduced (19, 20). Recently, several researchers have proposed to estimate cardiac and respiratory phases from the central k-space regions using band-pass filtering; the data is then binned to the respective phases, followed by reconstruction using compressed sensing (21, 22) or low-rank tensor methods (23, 24). These methods depend on the accurate estimation of phases using prior information about the cardiac and respiratory rates, which may degrade with irregular respiration or arrhythmia. We have recently introduced the SToRM (25–28) framework as an alternative to explicit motion resolved strategies (21–23). SToRM assumes that the images in the free-breathing dataset lie on a smooth and low-dimensional manifold parameterized by cardiac & respiratory phases. SToRM acquisition relies on navigator radial spokes, which are used to compute the manifold Laplacian matrix, to capture the structure of the manifold. Once the Laplacian matrix is available, the estimation of the dataset simplifies to a quadratic regularization scheme. SToRM can provide reliable free breathing reconstructions from around 40 seconds/slice of scan time, which ensures that the image manifold is well-sampled.

The main focus of this work is to further reduce the scan time of SToRM by combining the patient specific SToRM prior with deep-learned priors, which are population generalizable. While the direct deep learning approaches (17, 18) that estimate the images directly from the measured k-space data are computationally efficient, it is not straight forward for them to ensure data consistency or incorporate patient specific priors. We hence rely on our recent model-based deep learning (MoDL) framework, which formulates the image recovery as an optimization scheme (29, 30), where the cost function is the combination of a data consistency term with a deep learned prior; the unrolling of an iterative algorithm to solve the above cost

function translates to a deep network. This main difference of this scheme with other model based methods (16, 19, 31) is the use of embedded conjugate gradient (CG) blocks and the sharing of network parameters between iterations; our results in (29, 30) show that sharing of trainable parameters across iterations reduces the training data requirement significantly, while the use of CG blocks within the deep network translates to improved results for a specified number of iterations. Note that the use of CG blocks results in a slightly longer run time compared to direct learning approaches (17, 18). However, the proposed framework still provides significantly shorter run times than classical compressed sensing strategies, thanks to the reduced number of iterations and reduced number of CG steps per iterations. In this work, we use this optimization based framework for the seamless integration of the forward model (coil sensitivity information, sampling pattern) with SToRM priors and deep learning priors.

The proposed MoDL-SToRM cost consists of a data consistency term, a deep learned prior that learns population generalizable information, and the SToRM prior; the framework may also be used with other regularizers such as (32–34). The CNN based prior exploits local image redundancies of the 2D+time dataset. By contrast, the SToRM prior exploits non-local redundancies between images in the dataset, which are specific to the cardiac and respiratory patterns of the subject. The regularization parameters that weigh the individual contributions of each term are also optimized during the training phase, eliminating the need for image specific tuning of parameters. The combination of deep learning with other complementary priors in the context of free-breathing image reconstruction is not reported in the literature, to the best of our knowledge. These complementary priors enable the recovery from highly undersampled measurements, thus reducing the acquisition time by 5-10 fold over SToRM. While our focus is on free-breathing applications in this work, the algorithm may also provide good reconstructions of relatively less challenging breath-held applications.

METHODS

Acquisition scheme

Four healthy volunteers instructed to breath normally were scanned at the Siemens Aera scanner in the University of Iowa hospitals to generate prospectively undersampled free-breathing ungated radial dataset. The data was acquired using a FLASH sequence with a 32 channel cardiac array. The scan parameters were TR/TE = 4.2/2.2ms, number of slices = 5, slice thickness = 5 mm, FOV = 300mm, spatial resolution = 1.17

mm. A temporal resolution of 42 ms was obtained by binning 10 consecutive lines of k-space per frame, including 4 uniform navigator lines. Each slice comprised of 10000 radial lines of the k-space binned to 1000 frames, resulting in an acquisition time of 42s. The raw k-space data was interpolated to a Cartesian grid and 7 virtual coils were approximated out of the initial 32 using a SVD based coil-compression technique. The coil sensitivity maps were estimated from the compressed data using ESPIRiT (35). The SToRM (25) reconstructed images were used as the reference to train the deep networks. We use subsets of the above data to demonstrate the utility of the proposed scheme.

MoDL-SToRM: formulation

We generalize the model-based deep learning framework (MoDL) by adding a SToRM prior:

$$\begin{aligned} \mathcal{C}(\mathbf{X}) \quad = \quad & \underbrace{\|\mathcal{A}(\mathbf{X})-\mathbf{B}\|_2^2}_{\text{data consistency}} \; + \; \frac{\lambda_1}{2}\underbrace{\|\mathcal{N}_{\mathbf{w}}(\mathbf{X})\|^2}_{\text{CNN prior}} \\ & + \quad \frac{\lambda_2}{2}\underbrace{\mathrm{tr}\left(\mathbf{X}^T\mathbf{L}\mathbf{X}\right)}_{\text{SToRM prior}}. \end{aligned} \qquad [1]$$

Here, $\mathcal{A}$ is the multi-channel Fourier sampling operator, which includes coil sensitivity weighting. $\mathcal{N}_{\mathrm{w}}$ is a 3-D CNN based estimator that estimates the noise and alias patterns in the dataset from local neighborhoods of the 2D+time dataset; $\|\mathcal{N}_{\mathbf{w}}(\mathbf{x})\|^2$ is a measure of the alias/noise contribution in the dataset $\mathbf{X}$ (29). The *denoised signal* can thus be estimated from the data $\mathbf{X}$ as

$$\mathcal{D}_{\mathbf{w}}(\mathbf{X}) = \left(\mathcal{I}-\mathcal{N}_{\mathrm{w}}\right)(\mathbf{X}) = \mathbf{X}-\mathcal{N}_{\mathbf{w}}(\mathbf{X}). \qquad [2]$$

Note that we can also express $\mathcal{N}_{\mathrm{w}}(\mathbf{X}) = \mathbf{X} - \mathcal{D}_{\mathrm{w}}(\mathbf{X})$, where $\mathcal{D}_{\mathbf{w}}(\mathbf{X})$ is the denoised version of $\mathbf{X}$, when it is corrupted with noise and/or artifacts. However, we expect $\mathcal{D}_w(\mathbf{X}) = \mathbf{X}$ when $\mathbf{X}$ is an image free from noise and artifacts; i.e, $\mathcal{N}_w(\mathbf{X}) = 0$ in this case. The above relation allows us to rewrite the second term in [1] as the norm of the differences between the original and denoised images. The SToRM prior $\mathrm{tr}\left(\mathbf{X}^T\mathbf{L}\mathbf{X}\right)$, exploits the similarities beyond the local neighborhood. The manifold Laplacian, $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is estimated from the k-space navigators (25). The diagonal matrix $\mathbf{D}$ is specified as $\mathbf{D}_{(i,i)} = \sum_j \mathbf{W}_{(i,j)}$, where $\mathbf{W}$ is a weight matrix, such that, the weight $\mathbf{W}_{(i,j)}$ is high when $\mathbf{x}_i$ and $\mathbf{x}_j$ have similar cardiac and/or respiratory phase. tr is the trace operator.

Alternating minimization algorithm

We expand the SToRM penalty as

$$\begin{aligned} 2\text{tr}(\mathbf{X}^T\mathbf{L}\mathbf{X}) &= 2\text{tr}\left(\mathbf{X}^T[\mathbf{D}-\mathbf{W}]\mathbf{X}\right) \\ &= 2\text{tr}(\mathbf{X}^T\mathbf{D}\mathbf{X}) - 2\text{tr}\left(\mathbf{X}^T\underbrace{\mathbf{W}\mathbf{X}}_{\mathbf{Q}}\right) \end{aligned}$$

We consider temporary variables $\mathbf{Y} = \mathcal{D}_{\mathbf{w}}(\mathbf{X})$ and $\mathbf{Q} = \mathbf{W}\mathbf{X}$ and rewrite [1] as:

$$\begin{aligned} \mathcal{C}(\mathbf{X}) &= \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2 + \frac{\lambda_1}{2}\|\underbrace{\mathbf{X}-\mathbf{Y}}_{\mathcal{N}_w(\mathbf{X})}\|^2 + \\ &\qquad \lambda_2\left(\text{tr}(\mathbf{X}^T\mathbf{D}\mathbf{X}) - \text{tr}(\mathbf{X}^T\mathbf{Q})\right), \end{aligned} \quad [3]$$

We note that [3] is equivalent to [1] when $\mathbf{Y} = \mathcal{D}_{\mathbf{w}}(\mathbf{X})$ and $\mathbf{Q} = \mathbf{W}\mathbf{X}$. Minimizing the objective with respect to $\mathbf{X}$, assuming variables $\mathbf{Y}$ and $\mathbf{Q}$ to be fixed and determined from the previous iterations yields:

$$\nabla_{\mathbf{X}} C = \mathcal{A}^*(\mathcal{A}(\mathbf{X}) - \mathbf{B}) + \lambda_1(\mathbf{X}-\mathbf{Y}) + \lambda_2\left(\mathbf{D}\mathbf{X} - \mathbf{Q}\right) = 0 \quad [4]$$

where $\mathcal{A}^*$ is the adjoint of $\mathcal{A}$. This can be solved as

$$\mathbf{X} = (\mathcal{A}^*\mathcal{A} + \lambda_1\mathbf{I} + \lambda_2\mathbf{D})^{-1}\underbrace{(\mathcal{A}^*(\mathbf{B}) + \lambda_1\,\mathbf{Y} + \lambda_2\,\mathbf{Q})}_{\mathbf{R}} \quad [5]$$

As $\mathbf{D}$ is diagonal, $(\mathcal{A}^*\mathcal{A} + \lambda_1\mathbf{I} + \lambda_2\mathbf{D})^{-1}$ can be implemented on a frame-by-frame basis. We solve for [5] for each frame of $\mathbf{X}$ using conjugate gradients algorithm. This provides us with an alternating algorithm:

$$\begin{aligned} \mathbf{Y}_n &= \mathcal{D}_{\mathbf{w}}(\mathbf{X}_n) && [6] \\ \mathbf{Q}_n &= \mathbf{W}\mathbf{X}_n && [7] \\ \mathbf{R}_n &= (\mathcal{A}^*(\mathbf{B}) + \lambda_1\mathbf{Y}_n + \lambda_2\mathbf{Q}_n) && [8] \\ \mathbf{X}_{n+1} &= (\mathcal{A}^*\mathcal{A} + \lambda_1\mathbf{I} + \lambda_2\mathbf{D})^{-1}\mathbf{R}_n. && [9] \end{aligned}$$

Once the number of iterations is fixed, the network can be unrolled to yield a deep network as in Fig

1.(a). The parameters of $\mathcal{D}_{\mathbf{w}}$ and the optimization parameters λ_1 and λ_2 are trainable and shared throughout the iterations. We initialize the iterations with the SToRM solution:

$$\mathbf{X}_0 = \arg\min_{\mathbf{X}} \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2 + \frac{\eta}{2} \operatorname{tr}\left(\mathbf{X}^T \mathbf{L} \mathbf{X}\right), \qquad [10]$$

where η is fixed and chosen manually to produce the best SToRM results.

Network and training details

We note that the unrolled network described by [6]-[9] is dependent on the weight matrix $\mathbf{W}$, which captures the non-local similarities between image frames. For each training/testing dataset, we estimate $\mathbf{W}$ from navigators; the dataset and the corresponding $\mathbf{W}$ are used for training/testing. The rest of the variables including the regularization parameters λ_1, λ_2, and the weights of N_w are learned during training. Specifically, we train the unrolled network in an end-to-end fashion using different sets of $\{\mathbf{X}_0, \mathbf{X}_g, \mathbf{W}\}$. Here $\mathbf{X}_g$ is the ground-truth, while $\mathbf{W}$ is fixed during training and is different for each training and test dataset. The CNN captures the local redundancies, which is independent from the non-local information in $\mathbf{W}$. Since the end-to-end network sees different sets of training data, each with different $\mathbf{W}$ matrices, it learns the network parameters that are invariant/independent of the specific $\mathbf{W}$.

Lagged update of $\mathbf{Q}_n$: The training scheme requires the storage of $\mathbf{Y}_n$, $\mathbf{Q}_n$, and $\mathbf{X}_n$. The straightforward training of the unrolled architecture in Fig. 1.(a) requires all these intermediate variables to be available on the GPU memory, which is often not feasible. We propose a lagged approach shown in Fig.1.(e), where $\mathbf{Q}_n$ is updated less frequently during training. We update $\mathbf{Q}_n$ iby making a forward pass through the network, assuming known network parameters. The $\mathbf{Q}_n$, each corresponding to 200 frames, are then stored in the computer memory and assumed to be fixed during the inner iterations. The trainable network parameters specified by $\mathbf{w}$, λ_1 and λ_2 are optimized in the inner loop on the GPU. We form batches of seven frames and the corresponding frames of the pre-computed $\mathbf{Q}_n$ for training. Following convergence of [3] (inner-loop) for a fixed $\mathbf{Q}_n$, we update $\mathbf{Q}_n$ and re-train the network, assuming the network parameters from the previous outer iteration as the initialization. We need multiple outer iterations for the training procedure to converge.

Training data set: The data was acquired on four healthy volunteers, each but one with two different views–short axis and four chamber view– resulting in a total of seven datasets. We used the data from four datasets for training and remaining three for testing. We extracted 3 non-overlapping groups of 200 frames

each from the above datasets, which were used for training. The SToRM reconstruction of the datasets from 1000 frames are considered as reference data. Whereas, the input to the network was $\mathbf{X}_0$, the solution to [10] computed with reduced number of frames (200).

Trainable parameters of the network: The CNN block specified by $\mathcal{D}_{\mathbf{w}}$ consists of a 6 layer CNN with 64 filters of dimensions $3 \times 3 \times 3$ in the first five layers, followed by two 3×3 filters in last layer. To deal with complex data, the real and imaginary part of the frames were passed as two channels of the input tensor. The total number of trainable parameters in the network is 151666 real variables. The sharing of the parameters across iterations provides good performance, while significantly reducing training data demand as shown in (29).

Training strategy: The network was trained with the Adam optimizer on mean squared error loss, implemented on TensorFlow and trained on a NVIDIA P100 GPU. We pre-trained the $\mathcal{D}_w$ to denoise various versions of SToRM-1000 reconstructions corrupted with different levels of noise, which took 18 hours (1200 epochs). Next, we trained a MoDL-SToRM with $N = 1$. Following a single iteration training, we considered a multi-iteration model ($N > 1$), with the parameters initialized by the ones learned with $N = 1$.

We observed that a network with two iterations was sufficient to provide good reconstructions; the performance saturated beyond two repetitions. The total training time was 35 hours. The final inference for 8.4 s of data was from a single forward pass containing $N = 2$ repetitions, which takes around 28 second for all 200 frames. This is significantly faster than most compressed sensing reconstructions.

We also compare the proposed MoDL-SToRM reconstruction scheme against (a) SToRM alone and (b) Tikhonov-SToRM shown in [11].

$$\begin{aligned} \mathcal{C}(\mathbf{X}) \quad = \quad & \underbrace{\|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2}_{\text{data consistency}} \quad + \quad \frac{\lambda_{\text{Tikh}}}{2} \underbrace{\|\boldsymbol{\nabla}(\mathbf{X})\|^2}_{\text{Tikhonov}} \\ & + \quad \frac{\lambda_2}{2} \underbrace{\text{tr}\left(\mathbf{X}^T \mathbf{L} \mathbf{X}\right)}_{\text{SToRM prior}} . \end{aligned} \qquad [11]$$

We consider reconstructions from 200 frames, corresponding to 8.4sec of acquisition time. All comparisons are made with SToRM reconstructions from 1000 frames (42s) using the signal to error ratio metric (in addition to the standard PSNR and SSIM) defined as

$$\text{SER} = 20 \log_{10} \left(\frac{\|\mathbf{X}_{1000}\|}{\|\mathbf{X}_{1000} - \mathbf{X}\|} \right), \qquad [12]$$

where $\mathbf{X}_{1000}$ denotes the SToRM-reconstruction from 1000 frames and $\mathbf{X}$ is the specific reconstruction. The visual comparisons of the reconstructed images, their time profiles, and error images with SToRM reconstructions from 1000 frames as ground truth, are shown in Fig. 2, 3, and 4.

Simulated data

We generated simulated free breathing datasets by extracting three cardiac cycles of a SToRM-1000 reconstruction and deforming (using Bspline interpolation) them in space and time to generate six synthetic datasets. These simulated datasets (four training & two testing) were retrospectively undersampled using six golden angle radial lines & four uniform radial navigators. The results on a test dataset is shown in Fig 2, while quantitative comparisons are in Table 1.

RESULTS

Selection of parameters

We first discuss how the parameters of the algorithm was selected.

Number of iterations N: We observe that the performance of MoDL-SToRM saturates with N. For example, for test dataset 1, we obtained PSNR of 37.13 dB, 40.68 dB, and 41.36 dB, with $N = 0$, 1, and 2, respectively. The change in performance from $N = 2$ to $N = 3$ was negligible; we choose $N = 2$ in the rest of the experiments.

Number of outer iterations in training N_{out}: We observe that few outer iterations were sufficient for the training to converge. We obtained PSNR of 40.62 dB 41.36 dB and 41.37 dB, when the number of outer iterations is 1, 2, and 3 respectively. We choose this setting for the rest of the experiments since the performance saturates at $N_{\text{out}} = 2$.

Comparisons with other methods

The comparisons on the simulated datasets in Fig. 1 show that the proposed method provides the best reconstructions, which is also confirmed by the quantitative results shown in Table. 1. The comparisons of the reconstructions from 200 frames in Fig. 3 show that the proposed algorithm provides the most accurate reconstructions, revealed by the reduced errors and improved SER. We observe that the performance of

SToRM suffers when the number of frames are reduced, evidenced by the high amount of noise like alias artifacts. Comparison of the proposed method with MoDL (29) (explained in supporting Information Fig. S1), is provided in the supporting Information (Fig. S3-S5, Table S1). MoDL only uses local information and is hence not able to provide high quality reconstructions for such high accelerations; however, we expect the MoDL to work well in breath-held applications such as (19). This signifies the need of the additional SToRM prior, which can exploit the non-local redundancy a simple CNN model cannot capture. The comparisons on prospective data shows that the proposed reconstruction from 8.4 s scan time is most comparable with SToRM from 42s of scan time, while the SToRM alone reconstructions from 8.4 s scan time results in noise amplification.

DISCUSSION & CONCLUSION

We introduced a model-based framework, which can accommodate learnable CNN priors along with conventional SToRM regularizers, for the recovery of free breathing and ungated cardiac MRI data from radial acquisitions. The CNN exploits local population-generalizable redundancies, while the SToRM prior enables the use of patient specific non-local redundancies that depend on the cardiac and respiratory patterns. Our experiments show that very few iterations of [6]-[9] is sufficient to provide good reconstructions. The fast saturation of performance with iterations is mainly due to the use of CG algorithm within the network. The improved performance in the context of limited training data can be attributed to the trainable parameters in the network, shared across iterations. The proposed scheme also provides a fast reconstruction time of around 30 seconds on a P100 GPU for the reconstruction of 200 frames.

We observe that from the error images in Fig. 2 & 3 that the proposed approach results in reduced overall errors compared to competing methods, there are relatively higher residual errors around the image edges. Nevertheless, we note that the residual error is comparable or lower in magnitude than the ones obtained by other methods at all spatial locations, including at edges. The results also show that the MoDL-SToRM approach outperforms Tikhonov-SToRM, demonstrating the use of learnable priors.We observe from the figures in the supplementary information (S.3 & S.4) that the use of MoDL alone provides poor quality reconstructions, while its combination with SToRM provides improved results. We note that the undersampling factor needed to enable free breathing and ungated cardiac MRI is quite high ($\approx$50 fold undersampling), compared to most deep learning-based acceleration schemes. MoDL uses local redundan-

cies to recover these images, which results in poor reconstructions in this highly undersampled setting. The SToRM scheme facilitates the combination of information from different image frames. The reduced effective sampling resulting from reduced acquisition time causes increased errors, which the added MoDL regularization reduces. Although the CNN network architecture is same for MoDL and MoDL-SToRM, the two networks are trained to minimize different cost functions and hence the learned weights are expected to be different. We note that the size of the CNN in the proposed scheme is significantly smaller than those available in the literature; as shown in (29, 30), the sharing of weights between iterations allows us to significantly reduce the data demand required to avoid overfitting. The decay of validation error with training iterations as shown in the supplementary material also confirms that the model is not overfitting the data. This is also confirmed by the validation curves shown in the supporting information (Fig. S2).

We rely on the alternating minimization strategy specified by [6]-[9] to solve [3]. We have not rigorously studied the convergence of this algorithm to the objective in [3], which is beyond the scope. Similarly, we have not studied the impact of the sampling pattern (e.g numbers of spokes) on the quality of the reconstructions in this note. The data was acquired using a sequence with 4 uniform radial navigators and 6 golden angle radial lines. This will be the focus of our future work. Each image is only sampled with 10 radial lines, which translates to 50 fold undersampling. MoDL-SToRM simultaneously exploits local and global redundancies to yield improved results. The SToRM scheme facilitates the combination of information from different image frames. With low number of frames, the SToRM alone regularization results in increased errors, which the added MoDL regularization reduces significantly. MoDL and MoDL-SToRM share the network architecture (except for λ_2) but the learned weights differ significantly. In this work, we restricted our attention to L_2 loss metric to train the network. Several researchers have recently proposed alternate metrics for network training with improved results. We had experimented with $L_2 - L_1$ losses with little improvement in quality. The performance of the algorithm may be improved using sophisticated training strategies including GAN, but is beyond the scope of this work.

References

1 Gay SB, Sistrom CL, Holder CA, and Suratt PM. Breath-holding capability of adults. implications for spiral computed tomography, fast-acquisition magnetic resonance imaging, and angiography. Investigative radiology, 1994; 29:848–851.

2 Huang F, Akao J, Vijayakumar S, Duensing GR, and Limkeman M. k-t grappa: A k-space implementation for dynamic mri with high reduction factor. Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine, 2005; 54:1172–1184.

3 Pruessmann KP, Weiger M, Scheidegger MB, and Boesiger P. Sense: sensitivity encoding for fast mri. Magnetic resonance in medicine, 1999; 42:952–962.

4 Liang ZP, Jiang H, Hess CP, and Lauterbur PC. Dynamic imaging by model estimation. International journal of imaging systems and technology, 1997; 8:551–557.

5 Sharif B, Derbyshire JA, Faranesh AZ, and Bresler Y. Patient-adaptive reconstruction and acquisition in dynamic imaging with sensitivity encoding (paradise). Magnetic Resonance in Medicine, 2010; 64: 501–513.

6 Tsao J, Boesiger P, and Pruessmann KP. k-t blast and k-t sense: dynamic mri with high frame rate exploiting spatiotemporal correlations. Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine, 2003; 50:1031–1042.

7 Jung H, Sung K, Nayak KS, Kim EY, and Ye JC. k-t focuss: a general compressed sensing framework for high resolution dynamic mri. Magnetic resonance in medicine, 2009; 61:103–116.

8 Lustig M, Santos J, Donoho D, and Pauly J. k-t SPARSE: High frame rate dynamic MRI exploiting spatio-temporal sparsity. In International Society on Magnetic Resonance in Medicine 2006.

9 Zhao B, Haldar JP, Christodoulou AG, and Liang ZP. Image reconstruction from highly undersampled (k, t)-space data with joint partial separability and sparsity constraints. IEEE transactions on medical imaging, 2012; 31:1809–1820.

10 Lingala SG, Hu Y, DiBella E, and Jacob M. Accelerated dynamic MRI exploiting sparsity and low-rank structure: kt SLR. IEEE transactions on medical imaging, 2011; 30:1042–1054.

11 Lingala SG and Jacob M. Blind compressive sensing dynamic MRI. IEEE transactions on medical imaging, 2013; 32:1132–1145.

12 Lingala SG and Jacob M. A blind compressive sensing framework for accelerated dynamic mri. In IEEE International Symposium on Biomedical Imaging, 2012.

13 Asif MS, Hamilton L, Brummer M, and Romberg J. Motion-adaptive spatio-temporal regularization for accelerated dynamic mri. Magnetic Resonance in Medicine, 2013; 70:800–812.

14 Mohsin YQ, Lingala SG, DiBella E, and Jacob M. Accelerated dynamic mri using patch regularization for implicit motion compensation. Magnetic resonance in medicine, 2017; 77:1238–1248.

15 Nakarmi U, Wang Y, Lyu J, Liang D, and Ying L. A kernel-based low-rank (klr) model for low-dimensional manifold recovery in highly accelerated dynamic mri. IEEE transactions on medical imaging, 2017; 36:2297–2307.

16 Diamond S, Sitzmann V, Heide F, and Wetzstein G. Unrolled Optimization with Deep Priors. In arXiv:1705.08041, 2017; pages 1–11. URL `http://arxiv.org/abs/1705.08041`.

17 Jin KH, McCann MT, Froustey E, and Unser M. Deep Convolutional Neural Network for Inverse Problems in Imaging. IEEE Transactions on Image Processing, 2017; 29:4509–4522.

18 Lee D, Yoo J, and Ye JC. Deep Residual Learning for Compressed Sensing MRI. In IEEE International Symposium on Biomedical Imaging, 2017; pages 15–18. ISBN 9781509011728.

19 Qin C, Schlemper J, Caballero J, Price A, Hajnal JV, and Rueckert D. Convolutional Recurrent Neural Networks for Dynamic MR Image Reconstruction. ArXiv e-prints, December2017.

20 Schlemper J, Caballero J, Hajnal JV, Price AN, and Rueckert D. A deep cascade of convolutional neural networks for dynamic mr image reconstruction. IEEE transactions on Medical Imaging, 2018; 37:491–503.

21 Feng L, Grimm R, Block KT, Chandarana H, Kim S, Xu J, Axel L, Sodickson DK, and Otazo R. Golden-angle radial sparse parallel mri: combination of compressed sensing, parallel imaging, and golden-angle radial sampling for fast and flexible dynamic volumetric mri. Magnetic resonance in medicine, 2014; 72:707–717.

22 Feng L, Axel L, Chandarana H, Block KT, Sodickson DK, and Otazo R. Xd-grasp: golden-angle radial mri with reconstruction of extra motion-state dimensions using compressed sensing. Magnetic resonance in medicine, 2016; 75:775–788.

23 Christodoulou AG, Hitchens TK, Wu YL, Ho C, and Liang ZP. Improved subspace estimation for low-rank model-based accelerated cardiac imaging. IEEE Transactions on Biomedical Engineering, 2014; 61:2451–2457.

24 Christodoulou AG, Shaw JL, Nguyen C, Yang Q, Xie Y, Wang N, and Li D. Magnetic resonance multitasking for motion-resolved quantitative cardiovascular imaging. Nature Biomedical Engineering, 2018; 2:215.

25 Poddar S and Jacob M. Dynamic mri using smoothness regularization on manifolds (storm). IEEE transactions on medical imaging, 2016; 35:1106–1115.

26 Poddar S, Lingala SG, and Jacob M. Joint recovery of under sampled signals on a manifold: Application to free breathing cardiac mri. In Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on, 2014; pages 6904–6908. IEEE.

27 Poddar S and Jacob M. Low rank recovery with manifold smoothness prior: Theory and application to accelerated dynamic mri. In Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on, 2015; pages 319–322. IEEE.

28 Poddar S and Jacob M. Recovery of noisy points on band-limited surfaces: Kernel methods re-explained. CoRR, 2018; abs/1801.00890. URL `http://arxiv.org/abs/1801.00890`.

29 Aggarwal HK, Mani MP, and Jacob M. Modl: Model based deep learning architecture for inverse problems. IEEE Transactions on Medical Imaging, 2018; pages 1–1. ISSN 0278-0062. 10.1109/TMI.2018.2865356.

30 Aggarwal HK, Mani MP, and Jacob M. Model based image reconstruction using deep learned priors (modl). In Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on, 2018; pages 671–674. IEEE.

31 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, 2018; 79: 3055–3071.

32 Lingala SG and Jacob M. A blind compressive sensing framework for accelerated dynamic mri. In Proceedings/IEEE International Symposium on Biomedical Imaging: from nano to macro. IEEE International Symposium on Biomedical Imaging, 2012; page 1060. NIH Public Access.

33 Hu Y, Ongie G, Ramani S, and Jacob M. Generalized higher degree total variation (hdtv) regularization. IEEE Transactions on Image Processing, 2014; 23:2423–2435.

34 Goud S, Hu Y, and Jacob M. Real-time cardiac mri using low-rank and sparsity penalties. In Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on, 2010; pages 988–991. IEEE.

35 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, 2014; 71:990–1001.

LEGENDS

Table 1: Quantitative comparison of the methods on simulated dynamic datasets. We report the signal to error ratio (SER), peak signal to noise ratio (PSNR), and structural similarity index (SSIM). These metrics are reported for the entire field of view. By contrast, the SER (dB) metrics reported in the Figures are reported only for the myocardium area.

Fig 1: Illustration of the proposed MoDL-SToRM framework. The proposed scheme is obtained by unrolling the iterations specified by [6]-[9] as shown in (a). Each iteration consists of CNN denoiser $\mathcal{D}_w$, specified by [6], SToRM update $\mathbf{Q}_i = \mathbf{W}\mathbf{X}_n$ specified by [7], and data-consistency enforcement specified by [9], as shown in (b). The CNN denoiser $\mathcal{D}_w$ is implemented as a residual network as shown in (c), where the architecture of $\mathcal{N}_w = \mathcal{I} - \mathcal{D}_w$ is shown in (d). Here, $\mathcal{N}_w$, the noise extractor operator. The main differences between this scheme and other model-based deep-learned schemes is the sharing of the weights across iterations as shown in (a) and the use of CG blocks to enforce the data-consistency in (b), when complex forward models such as multi-channel sampling is used. Note that unlike **DC** and $\mathcal{D}_w$ that involves local operations, the update of $\mathbf{Q}_n$ is global in nature; the direct implementation of the unrolled network in (a) is associated with high memory demand and is not feasible on current GPU devices. We use the training strategy in (e), where we use the lagged update of $\mathbf{Q}_n$. Specifically, we perform a forward pass through the network to determine $\mathbf{Q}_n$ for all the frames in each training dataset. These $\mathbf{Q}_n$ parameters are stored. Batches of seven frames of $\mathbf{X}_0$ and $\mathbf{Q}_n$ are fed into the network to update the network weights, which can be performed on the GPU. We propose to pre-compute $\mathbf{Q}_n$ in an outer-loop and update it less frequently than the network parameters.

Fig 2: Comparisons on the simulated dataset: (a) Full view of a single frame from the simulated ground truth time series of 500 frames. Only (red) cropped myocardium region is shown in (b). (b) Top row: Simulated ground truth time series of 500 frames. Following six rows are three sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 100 frames, ii) Tikhonov-SToRM reconstruction with 100 frames and iii) proposed with 100 frames. First column is the time profile along a vertical cut across the myocardium shown in green in (a). Following three columns show three cardiac states at one respiratory stage. The position of the respiratory stage is marked blue on the time profile, in the first column. Three cardiac states are neighboring frames near the marked time point. The SER (dB) reported in the figure corresponds to the myocardium area.

Fig 3: Comparisons on Dataset 1: (a) Full view of a single frame from the SToRM reconstruction using 1000 frames. Only (red) cropped myocardium region is shown. (b) Top row: SToRM reconstruction using 1000 frames. Following six rows are three sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM

reconstruction with 200 frames, ii) Tikhonov-SToRM reconstruction with 200 frames and iii) proposed with 200 frames. First column is the time profile along a vertical cut across the myocardium shown in green in (a). Following three columns show three cardiac states at one respiratory stage. The positions of the respiratory stage is marked blue on the time profile, in the first column. Three cardiac states are neighboring frames near the marked time point. The SER (dB) reported in the figure corresponds to the myocardium area.

Fig 4: Dataset 2: (a) Full view of a single frame from the SToRM reconstruction using 1000 frames. Only (red) cropped myocardium region is shown. (b) Top row: SToRM reconstruction using 1000 frames. Following six rows are three sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 200 frames, ii) Tikhonov-SToRM reconstruction with 200 frames and iii) proposed with 200 frames. First column is the time profile along a vertical cut across the myocardium shown in green in (a). Following three columns show three cardiac states at one different respiratory stage. The position of the respiratory stage is marked blue on the time profile, in the first column. Three cardiac states are neighboring frames near the marked time point. The SER (dB) reported in the figure corresponds to the myocardium area.

LEGENDS OF FIGURES & TABLES IN SUPPORTING INFORMATION

Fig S1: (a) Illustration of MoDL alone implementation used for comparisons in S3-S5. The differences between this scheme and current model-based deep-learned schemes are the sharing of the weights across iterations as well as the use of CG blocks to enforce the data-consistency, when complex forward models such as multi-channel sampling is used. (b) i-th iteration of the MoDL: the iterative algorithm alternates between the denoising of the dataset using local CNN block denoted by $\mathcal{D}_w$ and the **DC** block involving conjugate gradients to enforce data-consistency at each iteration. (c) $\mathcal{I} - \mathcal{N}_w = \mathcal{D}_w$, the denoising operator (d) $\mathcal{N}_w$, the noise extractor operator.

Fig S2: Training and validation loss curves for (a) MoDL-alone training (b) proposed: MoDL-SToRM training. The monotonic decay of training and validation error with training iterations show that the models are not over fitted. The sharing of the network parameters across iterations in the MoDL and MoDL-SToRM schemes enable us to keep the number of free parameters low, thus minimizing the risk of overfitting.

Fig S3: Elaboration of Figure 2 in main manuscript. Simulated Dataset: (a) Full view of a single frame from the SToRM reconstruction using 500 frames. Only (red) cropped Myocardium region is shown. (b) Top row: SToRM reconstruction using 500 frames. Following eight rows are four sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 100 frames, ii) MoDL with 100 frames, iii) Tikhonov-SToRM reconstruction with 100 frames and iv) proposed with 100 frames. First column is the time profile along a vertical cut across the Myocardium shown in green in (a). Following six columns show three cardiac states at two different respiratory stages. The positions of those two respiratory stages are marked blue and green on the time profiles, in the first column. Three cardiac states are neighboring frames near those two marked time points. The SER (dB) reported in the figure corresponds to the myocardium area.

Fig S4: Elaboration of Figure 3 in main manuscript. Dataset 1: (a) Full view of a single frame from the SToRM reconstruction using 1000 frames. Only (red) cropped Myocardium region is shown. (b) Top row: SToRM reconstruction using 1000 frames. Following eight rows are four sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 200 frames, ii) MoDL with 200 frames, iii) Tikhonov-SToRM reconstruction with 200 frames and iv) proposed with 200 frames. First column is the time profile along a vertical cut across the Myocardium shown in green in (a). Following six columns show three cardiac states at two different respiratory stages. The positions of those two respiratory stages are marked blue and green on the time profiles, in the first column. Three cardiac states are neighboring frames near those two marked time points. The SER (dB) reported in the figure corresponds to the myocardium area.

Fig S5: Elaboration of Figure 4 in main manuscript. Dataset 2: (a) Full view of a single frame from the SToRM reconstruction using 1000 frames. Only (red) cropped Myocardium region is shown. (b) Top row: SToRM reconstruction using 1000 frames. Following eight rows are four sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 200 frames, ii) MoDL with 200 frames, iii) Tikhonov-SToRM reconstruction with 200 frames and iv) proposed with 200 frames. First column is the time profile along a vertical cut across the Myocardium shown in green in (a). Following six columns show three cardiac states at two different respiratory stages. The positions of those two respiratory stages are marked blue and green on the time profiles, in the first column. Three cardiac states are neighboring frames near those two marked time points. The SER (dB) reported in the figure corresponds to the myocardium area.

Table S1: Elaboration of Table 1 in main manuscript. We compare the reconstruction methods for all test subjects across different recovery metrics. These metrics are reported for the entire field of view. Whereas, the SER (dB) metric in the previous three figures are reported for the myocardium area.

Dataset	Method	SER (dB)	PSNR (dB)	SSIM
Subject 1	SToRM	16.31	37.31	0.8868
	Tikhonov-SToRM	18.78	39.77	0.9021
	Proposed	**20.36**	**41.36**	**0.9386**
Subject 2	SToRM	14.54	33.33	0.8161
	Tikhonov-SToRM	16.55	35.40	0.8783
	Proposed	**20.12**	**38.97**	**0.9114**
Subject 3	SToRM	17.09	34.19	0.8345
	Tikhonov-SToRM	19.02	36.17	0.8484
	Proposed	**20.73**	**37.83**	**0.9021**
Simulated dataset 1	SToRM	15.04	34.60	0.6880
	Tikhonov-SToRM	15.29	34.85	0.6968
	Proposed	**23.43**	**42.98**	**0.9602**
Simulated dataset 2	SToRM	21.63	36.03	0.8641
	Tikhonov-SToRM	21.92	36.92	0.8696
	Proposed	**26.82**	**41.23**	**0.9721**

Table 1: Quantitative comparison of the methods on simulated dynamic datasets. We report the signal to error ratio (SER), peak signal to noise ratio (PSNR), and structural similarity index (SSIM). These metrics are reported for the entire field of view. By contrast, the SER (dB) metrics reported in the Figures are reported only for the myocardium area.

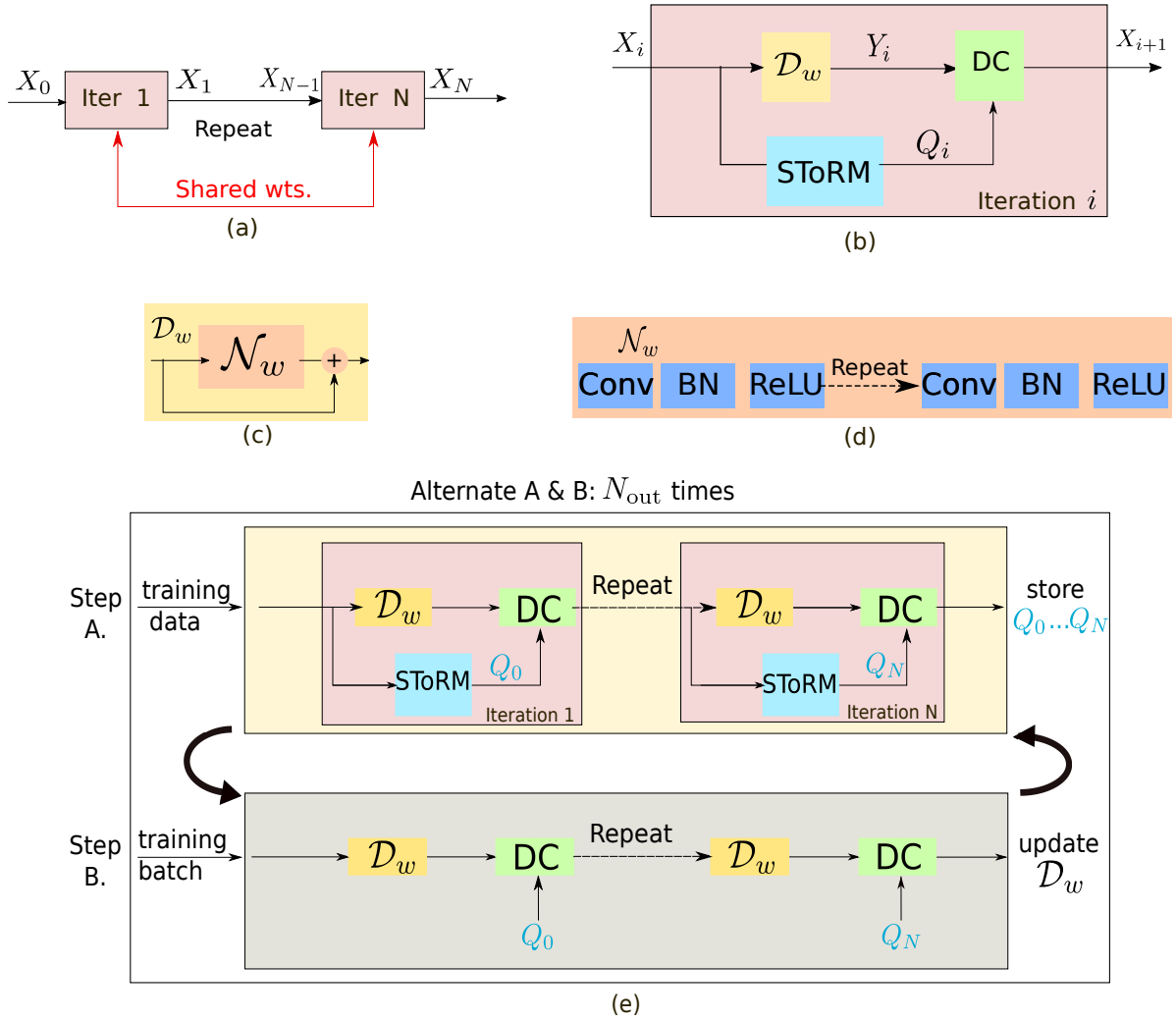
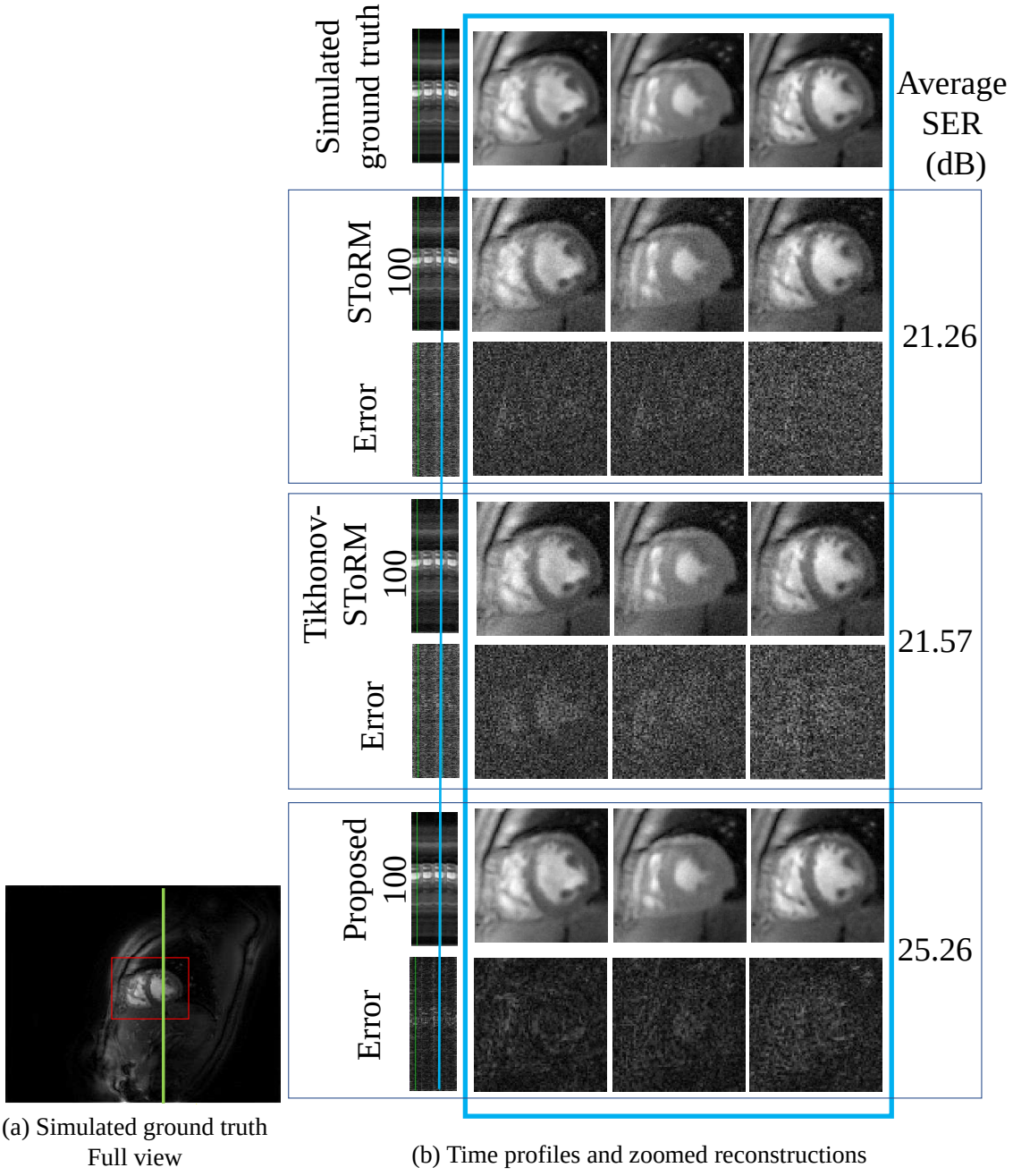


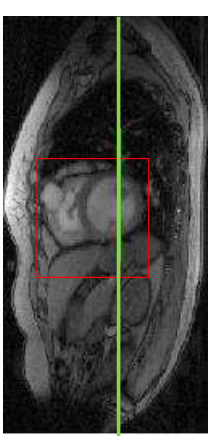
Figure 1: Illustration of the proposed MoDL-SToRM framework. The proposed scheme is obtained by unrolling the iterations specified by [6]-[9] as shown in (a). Each iteration consists of CNN denoiser $\mathcal{D}_w$, specified by [6], SToRM update $\mathbf{Q}_i = \mathbf{W}\mathbf{X}_n$ specified by [7], and data-consistency enforcement specified by [9], as shown in (b). The CNN denoiser $\mathcal{D}_w$ is implemented as a residual network as shown in (c), where the architecture of $\mathcal{N}_w = \mathcal{I} - \mathcal{D}_w$ is shown in (d). Here, $\mathcal{N}_w$, the noise extractor operator. The main differences between this scheme and other model-based deep-learned schemes is the sharing of the weights across iterations as shown in (a) and the use of CG blocks to enforce the data-consistency in (b), when complex forward models such as multi-channel sampling is used. Note that unlike **DC** and $\mathcal{D}_w$ that involves local operations, the update of $\mathbf{Q}_n$ is global in nature; the direct implementation of the unrolled network in (a) is associated with high memory demand and is not feasible on current GPU devices. We use the training strategy in (e), where we use the lagged update of $\mathbf{Q}_n$. Specifically, we perform a forward pass through the network to determine $\mathbf{Q}_n$ for all the frames in each training dataset. These $\mathbf{Q}_n$ parameters are stored. Batches of seven frames of $\mathbf{X}_0$ and $\mathbf{Q}_n$ are fed into the network to update the network weights, which can be performed on the GPU. We propose to pre-compute $\mathbf{Q}_n$ in an outer-loop and update it less frequently than the network parameters.



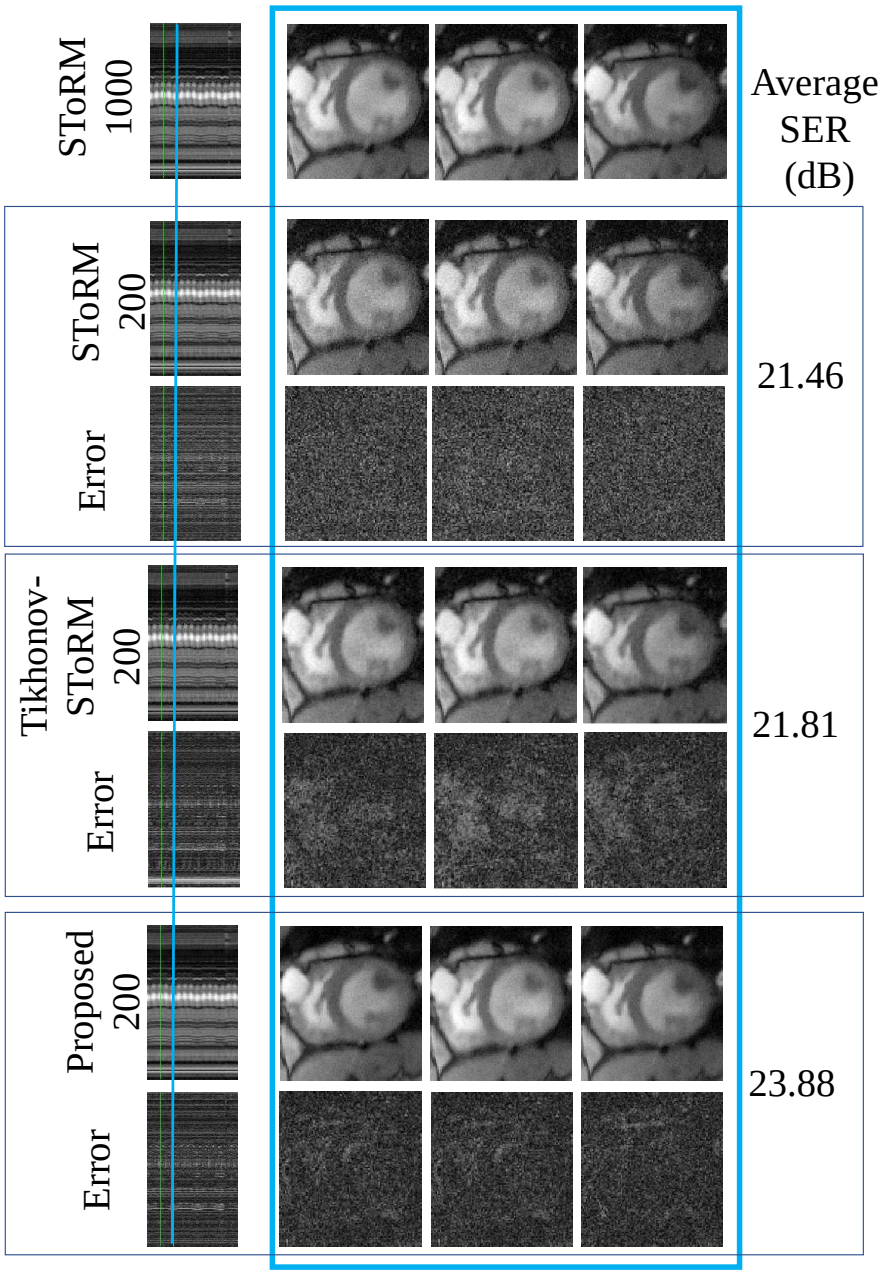
(a) Simulated ground truth Full view

(b) Time profiles and zoomed reconstructions

Figure 2: Comparisons on the simulated dataset: (a) Full view of a single frame from the simulated ground truth time series of 500 frames. Only (red) cropped myocardium region is shown in (b). (b) Top row: Simulated ground truth time series of 500 frames. Following six rows are three sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 100 frames, ii) Tikhonov-SToRM reconstruction with 100 frames and iii) proposed with 100 frames. First column is the time profile along a vertical cut across the myocardium shown in green in (a). Following three columns show three cardiac states at one respiratory stage. The position of the respiratory stage is marked blue on the time profile, in the first column. Three cardiac states are neighboring frames near the marked time point. The SER (dB) reported in the figure corresponds to the myocardium area.

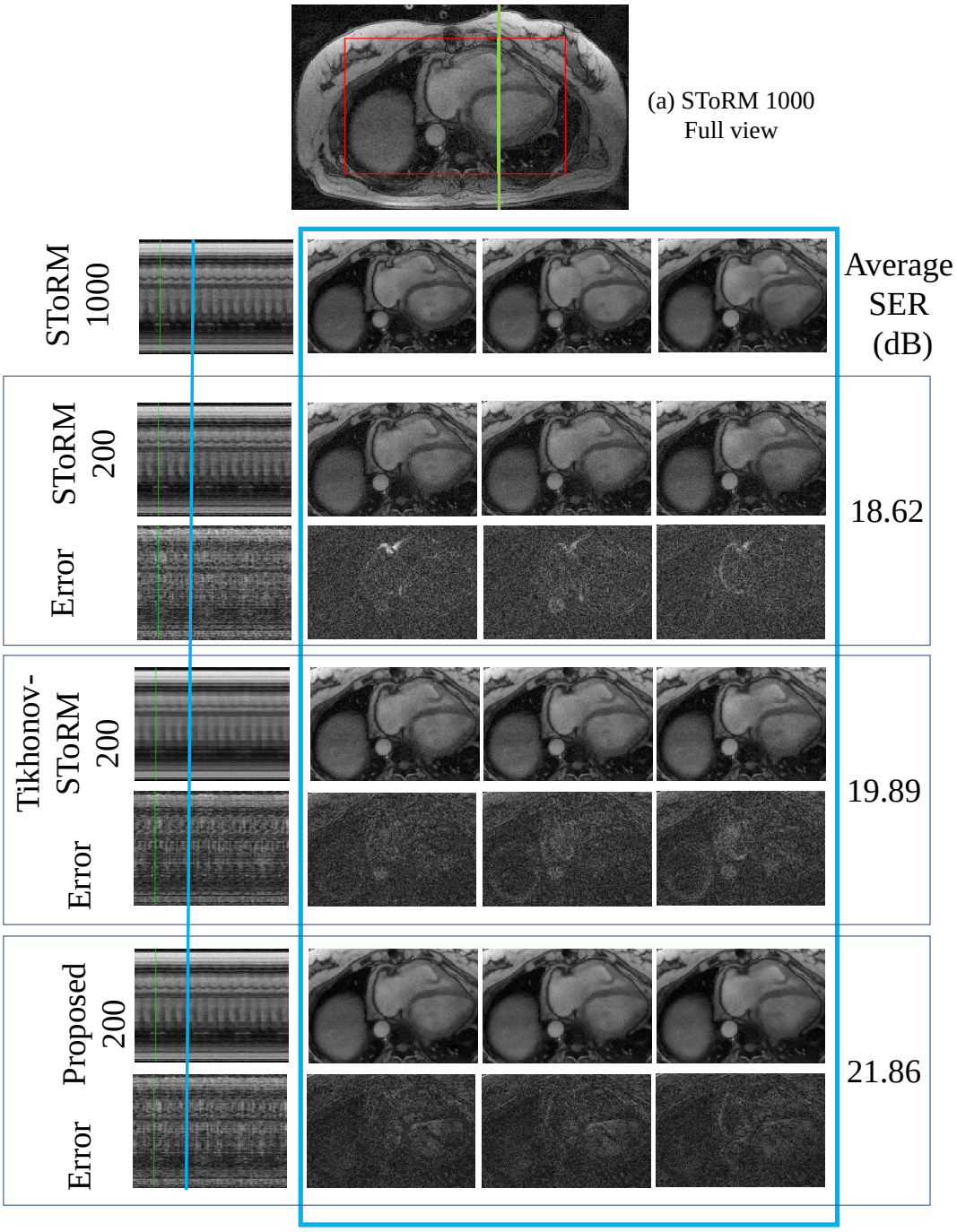


(a) SToRM 1000
Full view



(b) Time profiles, reconstructions and error images at different cardiac phases

Figure 3: Comparisons on Dataset 1: (a) Full view of a single frame from the SToRM reconstruction using 1000 frames. Only (red) cropped myocardium region is shown. (b) Top row: SToRM reconstruction using 1000 frames. Following six rows are three sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 200 frames, ii) Tikhonov-SToRM reconstruction with 200 frames and iii) proposed with 200 frames. First column is the time profile along a vertical cut across the myocardium shown in green in (a). Following three columns show three cardiac states at one respiratory stage. The positions of the respiratory stage is marked blue on the time profile, in the first column. Three cardiac states are neighboring frames near the marked time point. The SER (dB) reported in the figure corresponds to the myocardium area.



(b) Time profiles, reconstructions and error images at different cardiac phases

Figure 4: Dataset 2: (a) Full view of a single frame from the SToRM reconstruction using 1000 frames. Only (red) cropped myocardium region is shown. (b) Top row: SToRM reconstruction using 1000 frames. Following six rows are three sets of competing reconstructions and corresponding error (w.r.t to top row) images : i) SToRM reconstruction with 200 frames, ii) Tikhonov-SToRM reconstruction with 200 frames and iii) proposed with 200 frames. First column is the time profile along a vertical cut across the myocardium shown in green in (a). Following three columns show three cardiac states at one different respiratory stage. The position of the respiratory stage is marked blue on the time profile, in the first column. Three cardiac states are neighboring frames near the marked time point. The SER (dB) reported in the figure corresponds to the myocardium area.