Reconstruction of multi-shot diffusion-weighted MRI using deep learning

Yuxin Hu
yuxinh@stanford.edu

Minda Deng
mindad@stanford.edu

Yu Miao
miaoy11@stanford.edu

Abstract

Diffusion-weighted (DW) Magnetic resonance imaging (MRI) has become a very powerful tool that finds its applications in many different fields. Yet, one drawback is that the time for image reconstruction is relatively long using conventional reconstruction techniques. In this project, we combine traditional convex optimization method with deep neural networks to accelerate this process.

By replacing the presumed constraint with a U-net, we significantly reduced the required number of iterations, and the reconstruction time is reduced from 1 minute using conventional optimization algorithm to about 1 second. The difference map and averaged percentage L1 difference demonstrated that the proposed method had similar accuracies.

1. Introduction

MRI is a powerful and non-invasive imaging modality, and data are acquired in the spatial frequency domain by using spatially varying gradients. DW MRI, which measures diffusion rate of water, has been widely used in clinical applications and neuroscience researches. To achieve high-resolution diffusion-weighted images (DWIs) with reduced distortion, data are acquired by multiple segments and then combined, a.k.a. multi-shot imaging [1]. Unfortunately, significant aliasing artifacts and signal cancellation may exist due to the motion-induced phase inconsistencies between different shots. A relaxed convex model with a locally low-rank constraint has been proposed to reconstruct multi-shot DWIs, while the long reconstruction time has limited its application. In this work, we propose to use deep learning to bypass this limitation, achieving almost real-time reconstruction. In addition, we utilize the explicit relationship between acquired data and reconstructed images by using an unrolled network.

The goal of this project is to accelerate the image reconstruction process while maintaining similar accuracies and achieving relatively convincing results compared with traditional convex optimization methods.

The raw measured data in the frequency domain (k-space) will be the input for network and results from conventional optimization methods will be treated as ground truth. The loss function is set as L1-norm of the difference between the output and the ground truth. About seventeen hundreds brain DWIs are used to train the network, no data augmentation method is used [2].

We evaluated the results in two aspects. Qualitatively, we will check the existing of artifacts visually based on our knowledge of brain anatomy and MRI, and compare them to the ground truth images by eye. Quantitatively, we calculated the difference between images reconstructed by conventional optimization algorithm/ground truth images and the neural network.

2. Related Work

Deep learning has achieved a great success in many image processing areas, e.g. image classification, by using the knowledge and features extracted from huge amounts of data. There has also been some work on using deep learning to solve optimization problems, especially for medical image reconstruction [3]. It has been shown that neural networks have the capability to learn gradients of the loss function [4], as well as proximal operators [5, 6], which is often used in optimization algorithms to update constraint terms, achieving greater accuracies and a faster reconstruction speed compared with convex optimization methods. Another advantage of using deep learning is that there is no parameter to tune during testing time.

MRI data are usually under-sampled to reduced acquisition time, and some efforts have been made to use CNN to reconstruct MR images using different network structures for different applications [7, 8, 9]. There are several reasons why CNN is well-suited for MRI reconstruction. One is that MRI data are acquired in frequency domain as described before, and some algorithms fill un-acquired data points by applying convolution in the frequency domain [10, 11, 12, 13]. It is expected that we can have a more accurate convolution kernel by training on large amounts of data. Secondly, the proximal operator of L1 regularization (compressed sensing), which is also known as soft-thresholding and is a very common regularization term for image reconstruction, is analogical to non-linear activation function in CNN. Together, the structure of an iterative reconstruction algorithm is similar to that of a CNN.

In this work, we apply CNN to reconstruct multi-shot DW images, in which data are uniformly under-sampled, and different acquisitions have different phase, thus leading to a more challenging non-linear reconstruction problem. A recent work is to use FISTA to solve a convex approximated problem with a so-called locally low-rank (LLR) constraint [14], but one limitation is the long reconstruction time required. Inspired by the idea of unrolled network with deep priors from the work of Wetzstein, et al. [6], we unrolled this algorithm, and replace the LLR constraint by a U-net [15]. The U-net is originally used for image classification, with the ability to capture images features of different scales. This way, we hope to accelerate the conventional reconstruction method while maintaining comparable accuracies.

3. Data

With approval from the institutional review board and written informed consent, data were acquired on a 3 Tesla (T) MRI system (Discovery MR750, GE Healthcare) using a 2D single-refocused Stejskal-Tanner diffusion-weighted spin-echo EPI sequence.

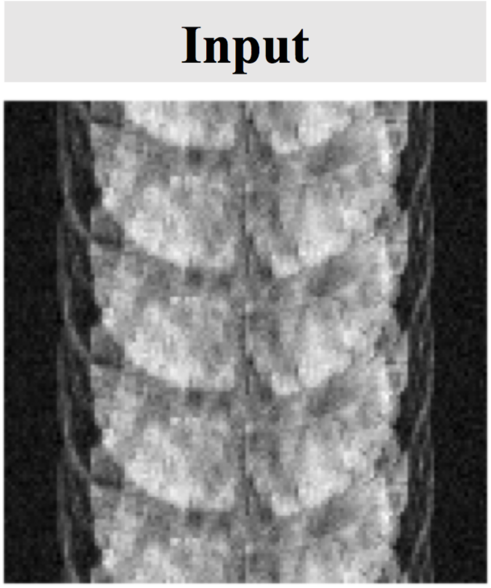

Figure 1. A typical input to our network.

2592 axial brain images have been acquired on seven healthy volunteers using a 32-channel head receive coil (Nova Medical) with the following parameters: TE/TR = 46/2000 ms, field-of-view (FOV) = 21 × 21 cm^2, matrix size = 248 × 244, slice thickness = 3/4 mm, number of shots = 4, b-value = $1000 s/mm^2$. Partial Fourier acquisition was used and the number of extra ky lines was 14 (fully-sampled), leading to 17 lines per shot. The readout direction was left-right.

The acquired data were first zero-filled to 256 × 256, and normalized based on the non-diffusion-weighted images. Thus, the size of input to the network is 256 × 256 × 32 (#channels) × 4 (#shots), and the size of output is 256 × 256 × 4. 1734 out of 2592 images and the left 858 images were used to train and test the network, respectively. Sensitivity maps were calculated from the multi-shot non-diffusion-weighted data using ESPIRiT [16] to construct the encoding matrix A. The acquired data were reconstructed by a convex optimization method, named shot-LLR, and the results were set as target images. A typical input image in real space is shown in Figure 1, and severe aliasing artifacts exists due to undersampling.

4. Methods

4.1. Conventional convex optimization method

Traditional MRI reconstruction usually involves solving an optimization problem as following,

$$\min_x \|Ax - y\|_2^2 + \lambda g(x) \tag{1}$$

The first term is known as data consistency term, in which A represents the encoding matrix, x is the image to be reconstructed and y is the acquired signal. The second term is a regularization term, in which lambda is the regularization parameter, and g(x) is a constraint on the image, such as L1-norm, L2-norm, total-variation and locally low-rank. These constraints are usually convex and their proximal operators are easy to calculate to make the problem easy to be solved, while many straightforward constraints are usually non-convex.

Solving these kinds of problems by conventional optimization problems, e.g. ISTA, FISTA and ADMM [17, 18], usually involve two steps, 1) updating x based on gradients calculated from the first term in eq. (1), and 2) updating variables based on the proximal operator of the regularization term. In the real world, the optimization algorithm is usually set to execute a predetermined number of iterations, a.k.a. unrolling optimization. A large amount of iterations is usually necessary to get a ”good” convergence, and this leads to low reconstruction efficiency. Another drawback of these kinds of algorithms is that they don’t make good use of the huge amount previous data, while this is especially the case for some medical imaging application. These model-based reconstruction methods are very convincing, for the reason that each term in the expression has a clear physical meaning and the solving process can be tracked pretty well.

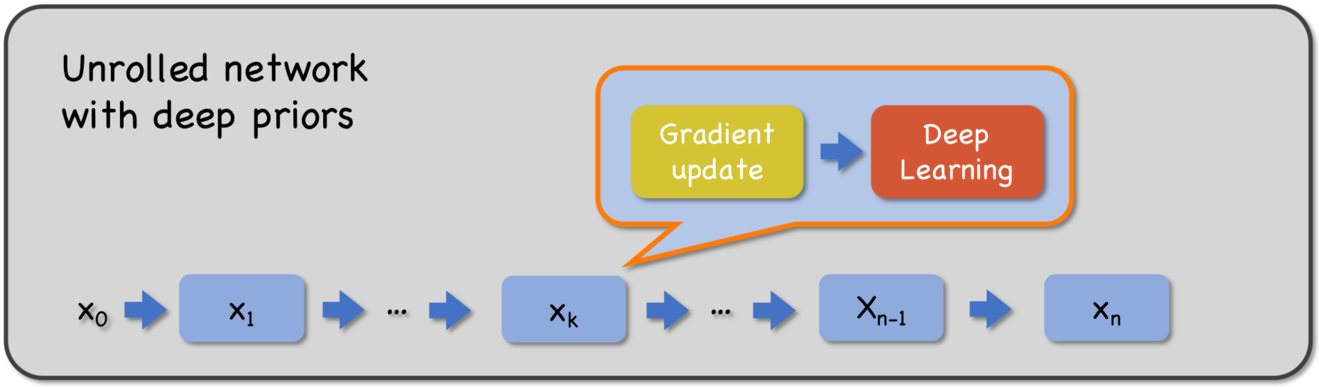

Figure 2. Schematic diagram of unrolled network with deep priors

4.2. Unrolled Network

In order to accelerate the image reconstruction speed while maintaining the advantages of these model-based methods, it is encouraged to replace the proximal operator in conventional optimization methods with a much more efficient function, e.g. a neural network. Within each iteration, the image is first updated based on the theoretical gradient from the data consistency term, and the updated image is then fed to the neural network to generate images for the next iteration. The schematic diagram of unrolled network with deep priors is shown in Figure 2. It has been shown that this has the capability to solve inverse problems in imaging, and only a small number of iterations, for example 5, is needed, thus the reconstruction time is significantly reduced [6].

The intuitive consideration is that using the regularization terms based on presumed assumptions is not the best choice. By training on huge amounts of data, we let the network itself to automatically learn a reasonable way to converge to the ”truth”, and networks in different iterations can be pretty different to achieve different functionalities, which might be another reason why it is more efficient than a fixed constraint (proximal operator).

However, everything has its opposite side: using a neural network, which is still a kind of black box, instead of a regularization term, gives us less knob to tune the update rule if the final results are not good enough. With this in mind, people might be less convinced that the MRI image reconstruction results from deep learning are accurate enough to for clinical diagnosis.

4.3. U-Net

The CNN part is where magic happens and the performance of the network will determine how fast the image can converge to the ground truth. One popular network structure in MRI field is U-Net [15], as shown in Figure 3. U-Net is a fancier version of ResNet and the idea is similar to FractalNet. The results from upstream of convolution layers will go different depths of downstream convolution layers and the outputs from layers in different levels are combined together to get the final output. By using this structure, layers at different levels tend to capture features in different scalings, i.e. convolution layers at higher levels focus more on local structures, while layers at bottom have a bigger vision.

The overall structure of our network is as follows: there are N iterations in total (N is a hyper-parameter to be tuned), and in each iteration, there is a specific U-net to update the image, and between these U-nets in different iterations, the output of previous network is updated based on the data consistency term (this can be seen as taking some prior knowledge to help improve the performance and make the results more convincing), before it is given to the following output.

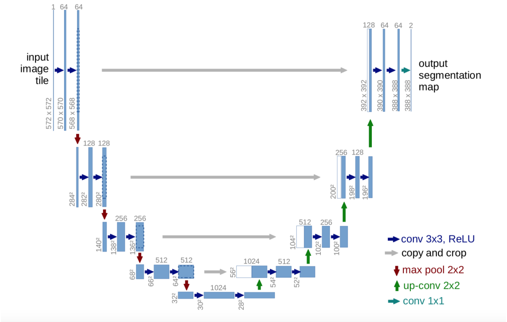

Figure 3. Schematic diagram of U-Net network

4.4. k-i domain input for CNN

The intrinsic raw data of MRI image are collected in k domain (frequency domain), while human understanding images are in i domain (inverse Fourier transform of frequency domain, i.e. image domain). One of the hyperparameters is to decide whether to use k domain or i domain

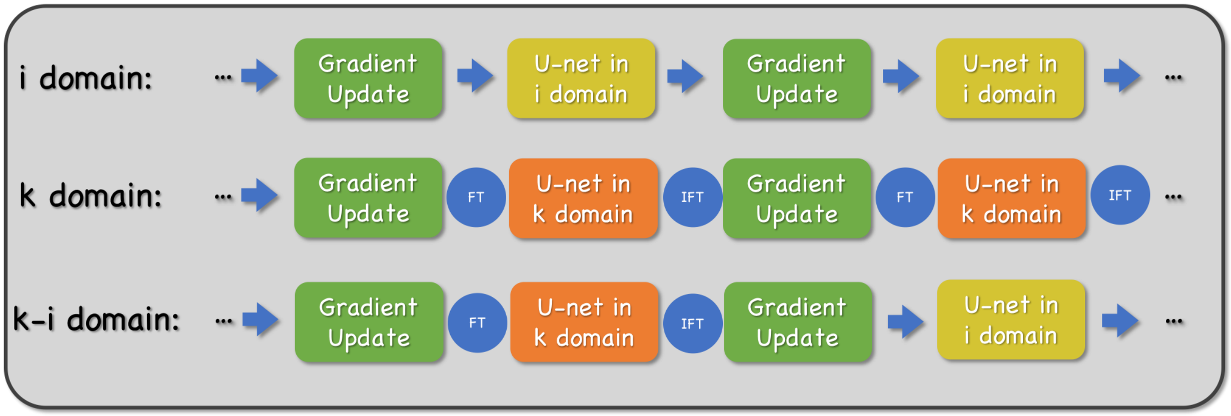

Figure 4. Schematic diagram of network structure with k and i domain input for CNN

data as the input of the network. We tried three different ideas of the network structures. The first is to use the k-space data as the input to the U-net in each iterations and therefore we need to perform Fourier transform and inverse Fourier transform before and after each U-net, respectively. The second idea is to use i domain for all the CNN input (which we refer to as the i net), which requires perform inverse Fourier transform first before raw data enters our network. The third idea is to alternatively use k domain or i domain data as the input of CNN network (which we refer to as the k-i net), while Fourier and inverse Fourier transform will be alternatively performed between before and after all CNN iterations.

Figure 4 shows the schematic diagram of the network structure that CNN processes data in either k domain, i domain or alternate between k and i domain. The physical intuition why we alternate input of CNN between i domain and k domain is that these two domains both have their own advantages and it would be a better idea to combine them. Since the signal is collected in k domain, CNN in k domain can better learn how to reconstruct the data that are not collected. Besides, the image is finally understood by doctor in i domain, so it would be much straightforward to remove structural artifacts in i domain.

5. Experiments

5.1. i net

As an initial experimentation, we implemented the i-net for our unrolled network model, namely the input to the U-net in each iteration is the real space input. The number of iterations in the network is one of the hyperparameters we played with. We have implemented our network with N = 5, 8, 10, 15 for the i-net. Our network with N = 5 is quite shallow especially compared with optimization algorithms with a few hundreds iterations and therefore the resulting reconstructed images are not very close to the ground truth values, which are from conventional convex optimization algorithms. As we increase the number of iterations in the unrolled net- work, the model space becomes larger and we should expect better reconstruction results (though this also makes the network harder to train and lose a little bit its advantage over conventional optimization approaches). With this in mind, we increased numbers of iterations to N = 8, N = 10 and N = 15. As expected, with deeper networks, the resulting images look better with fewer artifacts. Figure 5

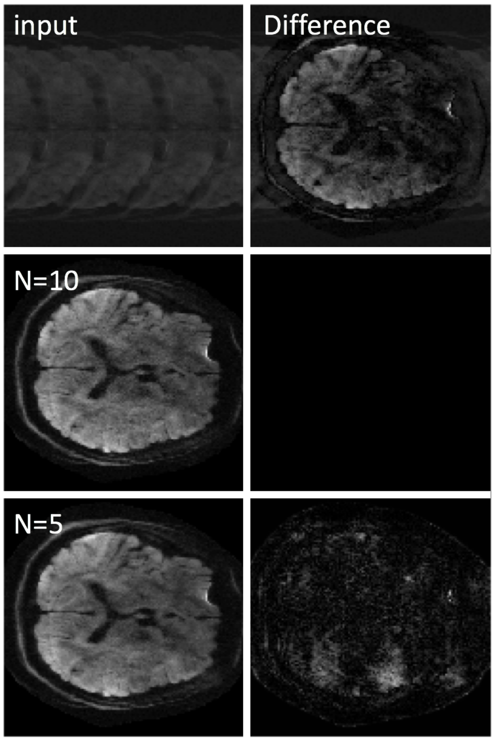

Figure 5. Representative output images for N=5 and N=10 i-net. Second row shows the difference from ground truth (not shown).

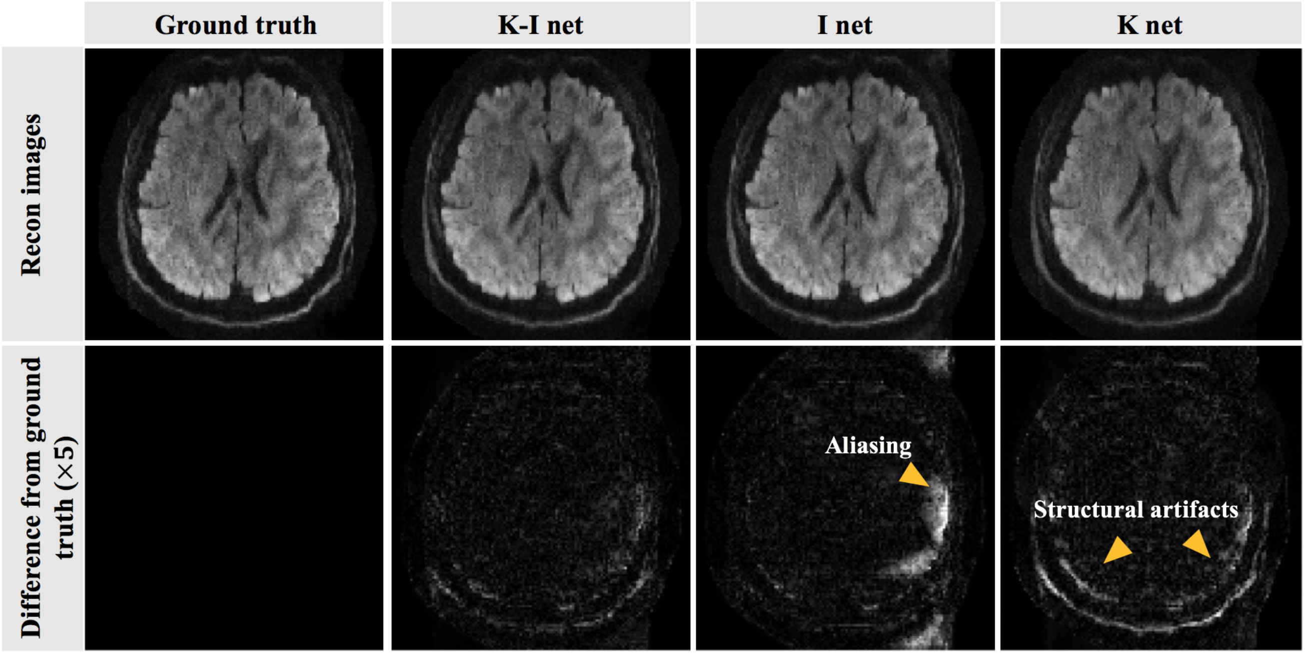

Figure 6. Representative reconstructed images from all three networks. Second row shows the difference from ground truth (amplified five times).

compares the results from N = 5 and N = 10. The first column shows the images of the input, and outputs from N = 5 and N = 10 networks respectively and the second column shows the L1 difference from the ground truth. Note that the L1 difference for N = 5 and N = 10 is amplified five times. However, we note that while increasing N from 5 to 10 yields significantly better images, further increasing to N = 10 and N = 15 does not improve the reconstructed images much. Even with 15 iterations in the unrolled network, $10\% \sim 15\%$ of the reconstructed images still have noticeable artifacts. This saturation suggests that this current model and architecture might be further modified and improved.

5.2. k net

The next thing we tried is the k-net, where we have a Fourier transform at the beginning of U-net so that the input to the U-net in each iteration is the k-space data and we have an inverse Fourier transform at the end of all the U-nets. As seen fromFigure 6, the results from the i-net typically suffer from aliasing problems while the results from the k-net sometimes have structural artifacts. We also implemented this network with different iteration depths, and increasing the iteration numbers helped reduce the structural artifacts, but not always.

5.3. k-i net

Having implemented both the i-net and the k-net and still having noticeable artifacts in the reconstructed images, we turned to the k-i net, whose input to the U-net alternates from the k-space data and real space data from iteration to iteration. The incentive is that by learning and operating on the input in both image and frequency domains, the network could better understand the underlying patterns and reconstruction rules in our data, eliminate artifacts in results of using a single net, and thus giving us better results. We implemented this k-i net. The only difference from the i-net is that before and after all the U-net in every other iteration where the input is the k-space data there are a Fourier transform step and an inverse Fourier transform step, respectively. We experimented with N = 4, 6, 8 for the k-i net. There is a significant improvement from N = 4 to N = 6, but this improvement saturates for N = 8. Therefore we used N = 6 in general since it is the shortest with good results.

Figure 7 shows the output of each of the six iterations for the k-i net, which gives us a rough idea of the effect of each iteration on the input data.

5.4. Comparison of three networks

The representative reconstructed images from all three different networks, i.e. the i net, the k net and the k-i net, are compared in Figure 6. The first image on the first row is the ground truth value from the convex optimization method as

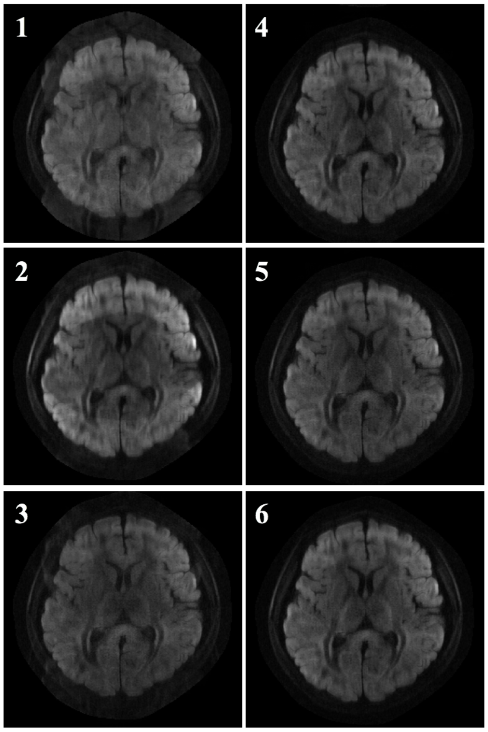

Figure 7. Representative output images from each of the 6 iterations in the network.

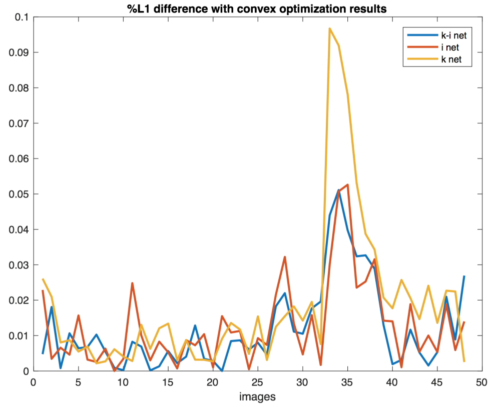

Figure 8. Percentage differences plot for 48 test images.

ground truth and the next three images are from these three networks. The second row shows the L1 difference from the ground truth amplified five times. As mentioned earlier, the i net and the k net often have noticeable artifacts (usually aliasing artifacts for the i-net and structural artifacts for the k-net). However, the reconstructed image from the k-i-net is quite close to the ground truth and the difference is very small and looks like uniform random noise. Therefore, we believe that the k-i net is the best performing model among the three. The reconstruction results and their difference from ground truth values for all three networks and for 48 test input images are shown in the supplementary material.

To better visualize the final loss for the test images from three different networks, we plotted their percentage differences from the ground truth values in Figure 8. The x-axis of this plot is the image index and the y-axis, the percentage difference, indicates how different the resulting images are from the targets. The curve for the k-i net in blue is the lowest which has an averaged percentage difference of around 1%.

6. Conclusion

In this project, we implemented unrolled network with deep priors (U-net as CNN) to accelerate DW MRI reconstruction process while trying to maintaining the same accuracy as conventional convex method. We experimented with three network models and the main distinction among the three is the type of input data we feed into the U-net in each iteration. We also played with different hyperparameters such as the number of iterations. The best performing model we have implemented and tested so far is the k-i net with 6 iterations. With this network trained on around 1700 MRI images of human brains, it makes the image reconstruction process nearly 60 times faster than conventional convex methods while the averaged L1 difference from ground truth is only around 1%.

We note that there are still limitations and things we might be able to further improve for our model. One major limitation is that our data were all taken in a relatively similar fashion and with the same resolution level. To be more specific, the encoding matrix A for all our data is quite similar while it could in principle be very different from case to case. However, our data were all taken by the same MRI facilities under the same sampling conditions. Therefore, new test data from other sources might not be reconstructed by our model as successfully as our current test data. Therefore, more data as well as data taken under different conditions could potentially improve the capability of our model.

So far, we have only tried the L1 and L2 norm from the ground truth values as our loss function. The ground truth values are calculated from conventional convex methods and can sometimes be non-ideal. Using the L1 loss as the loss function prevent our model from outperforming convex methods. One way of improving our model further is to modify our loss function. We could use the first term in Equation (1), the data consistent term, directly as the loss function, or a combination of different loss functions to instruct the model to optimize with a more direct goal. As a result, our network might generate reconstructed images even better than results from convex optimization algorithms. Also, though we have varied the structure of the U-net a little (depth, kernel size etc.), the parameter space is still largely unexplored. Different U-net architectures could be tested and might improve our model further.

7. Acknowledgments

We would like to thank Prof. Brian Hargreaves, Xinwei Shi and Dr. Qiyuan Tian for helpful discussions over the course of this project. We would also like to thank The Richard M. Lucas Center for Imaging for providing all data acquisition facilities and processing tools.

References

[1] Wenchuan Wu and Karla L Miller. Image formation in diffusion mri: a review of recent technical developments. *Journal of Magnetic Resonance Imaging*, 2017.

[2] Patrice Y Simard, David Steinkraus, John C Platt, et al. Best practices for convolutional neural networks applied to visual document analysis. 3:958–962, 2003.

[3] Ge Wang. A perspective on deep imaging. *IEEE Access*, 4:8914–8924, 2016.

[4] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017.

[5] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 2018.

[6] Steven Diamond, Vincent Sitzmann, Felix Heide, and Gordon Wetzstein. Unrolled optimization with deep priors, 2017.

[7] Morteza Mardani, Hatef Monajemi, Vardan Papyan, Shreyas Vasanawala, David Donoho, and John Pauly. Recurrent generative adversarial networks for proximal learning and automated compressive image recovery. *arXiv preprint arXiv:1711.10046*, 2017.

[8] Jian Sun, Huibin Li, Zongben Xu, et al. Deep admm-net for compressive sensing mri. pages 10–18, 2016.

[9] Bo Zhu, Jeremiah Z Liu, Stephen F Cauley, Bruce R Rosen, and Matthew S Rosen. Image reconstruction by domain-transform manifold learning. *Nature*, 555(7697):487, 2018.

[10] Mark A Griswold, Peter M Jakob, Robin M Heidemann, Mathias Nittka, Vladimir Jellus, Jianmin Wang, Berthold Kiefer, and Axel Haase. Generalized autocalibrating partially parallel acquisitions (grappa). *Magnetic resonance in medicine*, 47(6):1202–1210, 2002.

[11] Michael Lustig and John M Pauly. Spirit: Iterative self-consistent parallel imaging reconstruction from arbitrary k-space. *Magnetic resonance in medicine*, 64(2):457–471, 2010.

[12] Yuxin Hu, Tao Zhang, Kui Ying, and John Pauly. Generalized direct virtual coil (dvc) with spirit kernel for arbitrary sampling pattern. *ismrm*, 2015.

[13] Kang Wang, Philip J Beatty, Scott K Nagle, Scott B Reeder, James H Holmes, Mahdi S Rahimi, Laura C Bell, Frank R Korosec, and Jean H Brittain. Application of direct virtual coil to dynamic contrast-enhanced mri and mr angiography with data-driven parallel imaging. *Magnetic resonance in medicine*, 71(2):783–789, 2014.

[14] Yuxin Hu, Xiaole Wang, Evan Levine, Qiyuan Tian, Valentina Taviani, Frank Ong, Shreyas Vasanawala, Jennifer McNab, Bruce Daniel, and Brian Hargreaves. Multi-shot high-resolution brain diffusion-weighted imaging using phase regularized reconstruction. *ismrm*, 2018.

[15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[16] Martin Uecker, Peng Lai, Mark J Murphy, Patrick Virtue, Michael Elad, John M Pauly, Shreyas S Vasanawala, and Michael Lustig. Espiritan eigenvalue approach to autocalibrating parallel mri: where sense meets grappa, 2014.

[17] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[18] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.