Computational Imaging Lab Project Report

Author: Abdallah Eid
Supervisor: Dr. Zhenngo Tan

1 Introduction

Since the invention of MRI back in 1970s [1, 2], tremendous development has been made to the field. In particular developments in the preprocessing stage to allow for faster acquisition and better image quality. For instance the use of parallel imaging to exploit spatial information from multiple receiver coils [3] or the use of compressed sensing to exploit sparsity in the image domain [4].

Despite the success, especially of compressed sensing, the demand for deep learning based methods is growing. Applying compressed sensing requires meeting three conditions: sparsity, incoherence, and non-linear reconstruction. Sparsity is often achieved by transforming the signal into a domain where it is sparse or compressible, such as the wavelet or total variation (TV) domain [5, 6]. Incoherence is attained using random sampling patterns like poisson disc sampling or Cartesian undersampling. Non-linear reconstruction utilizes algorithms such as the iterative soft thresholding algorithm (ISTA) to recover the original signal from compressed measurements [7].

While compressed sensing shows promise in image reconstruction, it faces limitations. For example, solving optimization problems can be computationally intensive, particularly for large-scale ones, which limits its practicality in real-time applications [6]. Additionally, tuning parameters like regularization parameters or thresholds is necessary for optimal reconstruction performance, adding complexity [8]. Moreover, clinical MRI scans often use cartesian sampling, making it challenging to meet the incoherence criteria required by compressed sensing, especially for 2D sequences.

In this project, we aim to expedite MRI acquisition using deep learning methods, focusing on three approaches. The first approach combines variational regularization with deep learning to map undersampled k-space measurements to fully sampled MRI images [9]. Addressing inverse problems in imaging, like reconstruction from undersampled or noisy data, is the focus of the second approach, achieved by merging model-based methods with deep learning [10]. Lastly, we plan to train a neural network on undersampled data to forecast missing k-space data [11].

2 Methods

In this section, we'll walk you through the methods employed in this project. First, we'll cover the data preprocessing steps, and then we'll dive into the three approaches we're exploring.

2.1 Data preprocessing

In this project, we utilize two distinct datasets. The default dataset comes fully equipped with reconstructed images, undersampling masks, and coil sensitivity maps, making it ready for seamless training and testing. On the other hand, the fastMRI dataset, focused on brain MRI data with T2-weighted sequences, only provides fully sampled k-space data. To ready this dataset for training and testing using the aforementioned approaches, we must undertake the following steps:

- Reconstruct images from the kspace data and crop the images to the desired size.
- Normalize the reconstructed images to stable the training.
- Estimate coil sensitivity maps using ESPIRIT algorithm. [12]
- Generate different undersampling masks to test the performance of each approach.

It's worth noting that I exclusively utilized the first 8 slices (Fig. 1.) of the brain volume. This decision stemmed from the realization that the subsequent slices (Fig. 2.) contain less pertinent image content, thus making a negligible contribution to the training process.

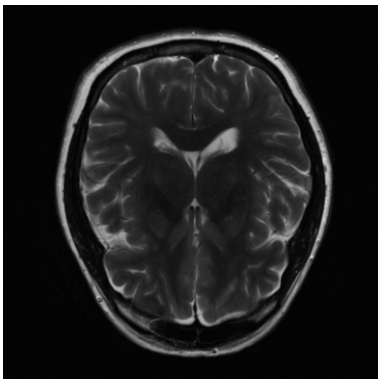


Fig. 1 Slice from the top

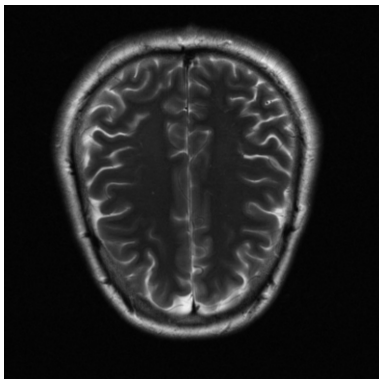


Fig. 2 Slice from the bottom

2.2 VarNet Approach

Variational Network is designed to learn a complete reconstruction procedure for complex-valued multichannelMR data, including all free parameters which would otherwise have to be set empirically. The entire appraoch is based on the following optimization problem:

$$u^{t+1} = u^t - \sum_{i=1}^{N_k} (K_i^t)^T \Phi_i^{t\prime}(K_i^t u^t) - \lambda^t A^*(Au^t - f), \quad 0 \leq t \leq T-1. \tag{1}$$

Where u^t is the current estimate of the image, K_i^t is the linear operator, $\Phi_i^{t\prime}$ are the activation functions defined by the first derivative of potential functions, A^* is the adjoint linear sampling operator, f is noisy data, and λ^t is the regularization parameter. This optimization problem is depecited in Fig. 3.

The training process of the VarNet is shown in Fig. 4. To obtain a reconstruction, we feed the undersampled k-space data, coils senstivity maps and zero filled image to the network. The goal is to learn a set of parameters of the VN during training procedure. For this purpose, we compare the current reconstruction of the VN to an artifact-free reference using a similarity measure. This gives us the reconstruction error which is propagated back to the VN to compute a new set of parameters.

In the VarNet experiments, I conducted training over 20 epochs using the Adam optimizer [13] with a learning rate set at 0.001. Additionally, I explored variations in network performance by altering several parameters, including the type of masks, acceleration rates, number of cascades, loss function, and the incorporation of different transformer-based networks. Notably, I switched to the IIPG optimizer [9] on a single occasion, aligning with the recommendation outlined in the original paper.

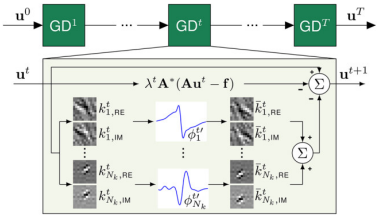


Fig. 3 VarNet Building block.

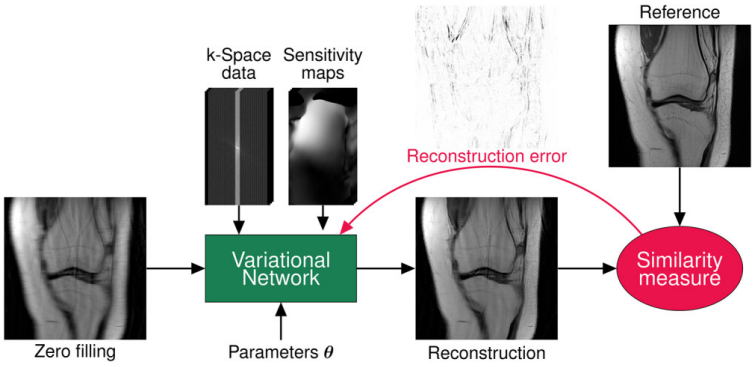


Fig. 4 VarNet training process.

2.3 MoDL Approach

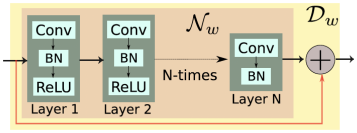
Model-Based Deep Learning (MoDL) is a combination of model-based reconstruction schemes and deep learning. The authors used a variational model for the data consistnecy term and a CNN-based denoiser to capture image redundancy. The main inverse problem can be formulated in the following optimization problem:

$$\boldsymbol{x}_{rec} = \arg\min_{\boldsymbol{x}} ||A(\boldsymbol{x}) - \boldsymbol{b}||_2^2 + \lambda||\boldsymbol{x} - \mathcal{D}_w(\boldsymbol{x})||^2, \tag{2}$$

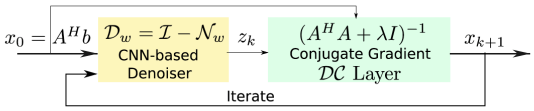
Where $\boldsymbol{x}_{rec}$ is the reconstructed image, A is the forward operator, $\boldsymbol{b}$ is the undersampled k-space data, λ is the regularization parameter, $\mathcal{D}_w$ is "denoised" version of x, after the removal of alias artifacts and noise and $\boldsymbol{x}$ is the input image.

The CNN-based denoiser conists of N layers of convolutional layers, batch normalization and ReLU activation functions as shown in Fig. 5. (a) The Nth-layer does not have ReLU to avoid truncating the negative part of the learned noise patterns.

Fig. 5. (b, c) shows the architecture of the MoDL network. The network consists of two main components: the data consistency layer and the denoising layer. The data consistency layer is a linear layer that enforces the data consistency constraint. The denoising layer is a CNN-based denoiser that removes alias artifacts and noise from the input image.



(a) The Residual learning based denoiser



(b) Proposed Model-based Deep Learning (MoDL) architecture

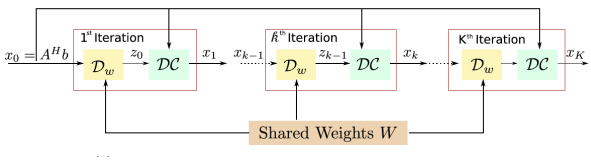


Fig. 5 MoDL architecture.

For experiments related to MoDL, I used the same configurations as VarNet.

2.4 SSDU Approach

This study introduces a novel self-supervised learning approach for training physics-guided DLMRI reconstruction without fully sampled reference data. SSDU splits acquired kspace indices into two sets, with one used in the network's data consistency unit and the other for defining the loss function. The network is trained and evaluated solely on acquired measurements, without assumptions about image output.

Fig. 6. illustrates the training procedure of the SSDU. Initially, the fully sampled k-space undergoes undersampling via an equispaced cartesian sampling mask. Subsequently, the undersampled k-space is divided into two distinct sets. Both sets are further undersampled using either gaussian or uniform masks, ensuring that the center of the k-space remains intact in the set utilized for the data consistency unit.

In this phase, the network's output is transformed into k-space, where a subset of available measurements at a specific location ($\wedge$) is compared with the corresponding reconstructed k-space values. The training loss is derived from this comparison, facilitating the adjustment of network parameters accordingly.

For the experimentes related to SSDU, I trained the network for 30 epochs, with Adam optimzer [13] and learning rate of 0.001. I also tested how the performance of the network changes with changing the type of masks, acceleration rates, rho (ρ) values. where ρ is the ratio which determines how many measurements are used for the training loss metric.

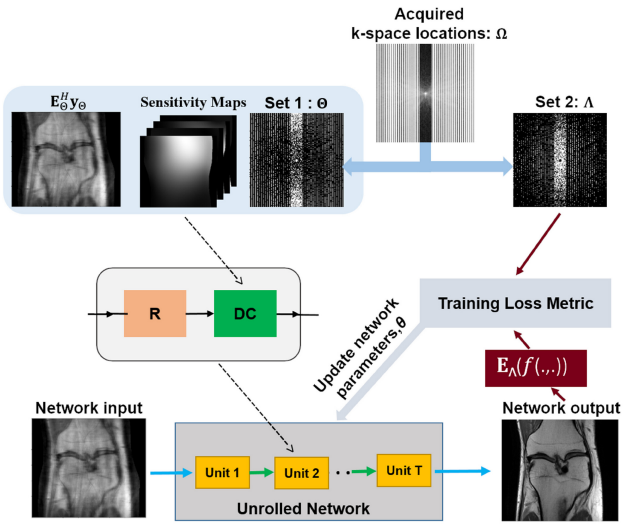


Fig. 6 SSDU architecture.

3 Results

In this section, I will present the results of the experiments conducted in this project. I will start by presenting the results of the VarNet approach, followed by the MoDL approach and finally the SSDU approach.

3.1 VarNet Results

3.1.1 Using different masks & cascades (k)

In the first experiment, I analyzed how VarNet performs with different masks and cascades. For both 1D gaussian and spiral the performance decreases with increasing cascades. Whereas radial masks reached its peak performance at 5 cascades (Fig. 7.).

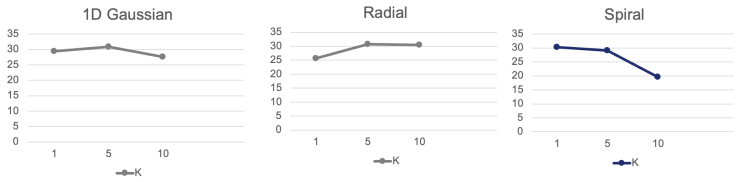


Fig. 7. Results of VarNet training with different masks.

3.1.2 Replacing UNET with transformer-based networks

Upon substituting Unet with AttentionUnet and SwinUNETR with a spiral mask, notable improvements were observed, particularly with SwinUNETR. It surpassed Unet in both PSNR and SSIM values, showcasing promising advancements. Results can be found in fig. 8. and fig. 9.

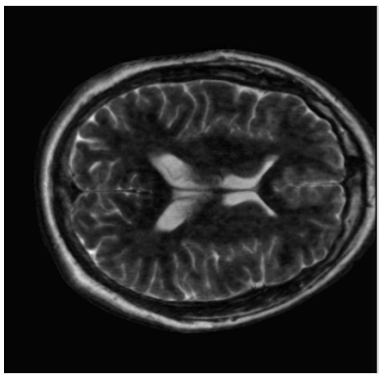


Fig. 8 AttentionUnet result.

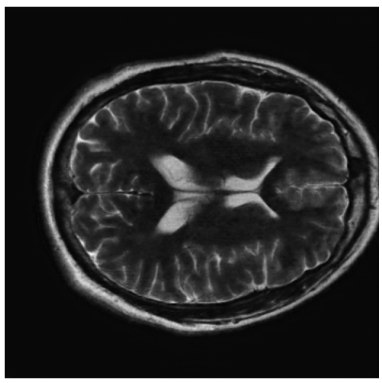


Fig. 9 SwinUNETR result.

3.1.3 Using SSIMLoss and IIPG optimizer

In the last two experiments, I switched the loss function to SSIMLoss and changed the optimizer to IIPG. Unlike MoDL, VarNet training showed success; however, it resulted in noticeable bubble-like artifacts in the reconstructed images. Implementing the IIPG optimizer, as outlined in the paper, didn't lead to significant improvements in the network's performance.

3.2 MoDL Results

3.2.1 Using different masks & iterations (k)

Employing different masks with MoDL doesn't seem to impact the overall performance significantly. A clear pattern emerges: as we increase the number of iterations, the network's performance tends to decrease. Surprisingly, the best results were achieved with just one iteration.

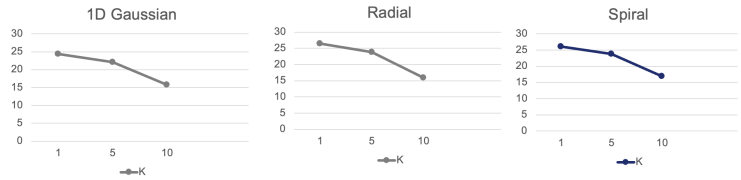


Fig. 10 Results of MoDL training with different masks.

3.2.2 Replacing UNET with transformer-based networks

Replacing the denoiser in MoDl with either AttentionUnet or SwinUNETR did work well and the maximum PSNR value was approximately **4.2** in both as also shown visually in Fig. 11. and Fig. 12.

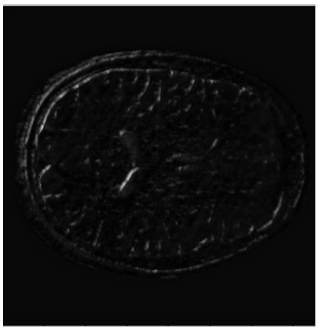


Fig 11. AttentionUnet result.

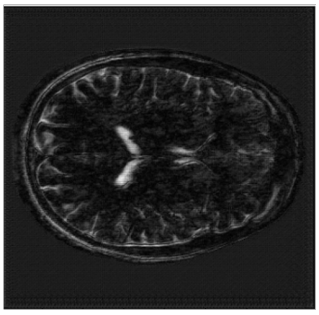


Fig 12. SwinUNETR result.

3.2.3 Using SSIMLoss function

When utilizing SSIMLoss as the loss function in MoDL, the outcomes were unexpected. Despite training the network for numerous epochs, I consistently obtained negative PSNR values, which was quite surprising.

3.3 SSDU Results

3.3.1 Equispaced mask & different acceleration rates

In this experiment, the evaluation metric used is the SSIM as PSNR is quite high and I was not able to get a good comparison between the different masks. The results are shown in Fig. 13. and Table 1. As we can observe from the results, the SSIM value decreases as the acceleration rate increases.

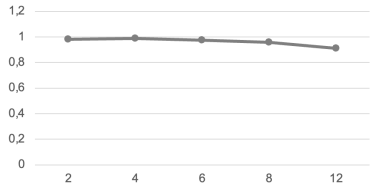


Fig. 13 SSDU results with different acceleration rates.

rate	2	4	6	8	12
SSIM	0.9805	0.9884	0.9738	0.958	0.9101

Table 1 SSDU results with different acceleration rates.

3.3.2 Different masks and same acceleration rate

In this experiment, I evaluated the performance using radial and spiral masks with identical acceleration ratios (8), rho = 0.4, and a training duration of 20 epochs. The results were quite comparable for both masks, with the spiral mask yielding a slightly higher SSIM value (0.9896) compared to the radial mask (SSIM=0.9884).

3.3.3 Same equispaced mask and different rho

For the last experiment ragrading SSDU, I changed only rho value keeping epochs =30, acceleration rate = 4, and unrolls = 10. The result is approximately the same for all rho values as indicated in Table 2.

ρ	0.1	0.4	0.9
SSIM	0.9950	0.9868	0.9946

Table 2 SSDU results with different rho values.

4 Discussion

Initially, training MoDL and VarNet posed challenges due to using unnormalized data, leading to poor performance (negative PSNR). Normalizing the data significantly improved model performance.

Due to some brain slices lacking content, I opted to utilize only the initial 8 slices of the volume. While training, there was no notable impact on model performance, but visualization improved significantly.

Incorporating transformer-based networks into both MoDL and VarNet proved to be a promising endeavor. While models like AttentionUnet[14] and SwinUNETR[15] did not enhance the performance of MoDL, I achieved optimal results by training VarNet using the SwinUNETR model. This approach yielded the highest performance between MoDL and VarNet.

Using SSIMLoss for evaluation was crucial. MoDL had poor results (negative PSNR), while VarNet showed bubble-like artifacts.

5 Conclusion

In conclusion, it's evident that the SSDU approach surpasses both MoDL and VarNet in terms of SSIM values and the quality of reconstructed images. While VarNet exhibited promising results, particularly with the SwinUNETR model, it generally outperformed MoDL. However, MoDL's performance faltered notably when employing the SSIMLoss.

References

[1] P C Lauterbur. Image formation by induced local interactions: Examples employing nuclear magnetic resonance. *Nature*, 242:190–191, 1973. doi:10.1038/242190a0.

[2] Peter Mansfield. Multi-planar image formation using NMR spin echoes. *J Phys C*, 10:55–58, 1977. doi:10.1088/0022-3719/10/3/004.

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

[4] M. Lustig, D. Donoho, and J.M. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magn. Reson. Med.*, 58:1182–1195, 2007. doi:10.1002/mrm.21391.

[5] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52:1289–1306, 2006. doi:10.1109/TIT.2006.871582.

[6] E. J. Candes and M. B. Wakin. An Introduction To Compressive Sampling. *IEEE Signal Processing Magazine*, 25:21–30, 2008. doi:10.1109/MSP.2007.914731.

[7] I. Daubechies, M. Defrise, and C. De Mol. An Iterative Thresholding Algorithm for Linear Inverse Problems with a Sparsity Constrains. *Comm. Pure Appl. Math.*, 57:1413–1457, 2004. doi:10.1002/cpa.20042.

[8] Figueiredo MA and Nowak RD. An EM algorithm for wavelet-based image restoration. *IEEE Trans Image Process.*, 12:906–16, 2003. doi:10.1109/TIP.2003.814255.

[9] Kerstin Hammernik, Teresa Klatzer, Erich Kobler, Michael P Recht, Daniel K Sodickson, Thomas Pock, and Florian Knoll. Learning a variational network for reconstruction of accelerated MRI data. *Magn. Reson. Med.*, 79:3055–3071, 2018. doi:10.1002/mrm.26977.

[10] H. K. Aggarwal, M. P. Mani, and M. Jacob. MoDL: Model-Based Deep Learning Architecture for Inverse Problems. *IEEE Transactions on Medical Imaging.*, 38:394–405, 2019. doi:10.1109/TMI.2018.2865356.

[11] Yaman B, Hosseini SAH, Moeller S, Ellermann J, Uğurbil K, and Akçakaya M. Self-supervised learning of physics-guided reconstruction neural networks without fully sampled reference data. *Magn Reson Med.*, 84:3172–3191, 2020. doi:10.1002/mrm.28378.

[12] Uecker M., Lai P., Murphy M.J., Elad M. Virtue P., Pauly J.M., Vasanawala S.S., and Lustig M. ESPIRiT—an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA. *Magn. Reson. Med.*, 71:990–1001, 2020. doi:10.1002/mrm.24751.

[13] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv*, 2017. arXiv:1412.6980.

[14] Ozan Oktay, Jo Schlemper, and et al. Attention U-Net: Learning Where to Look for the Pancreas. *arXiv*, 2018. arXiv:1804.03999.

[15] Ali Hatamizadeh, Vishwesh Nath, and et al. Swin UNETR: Swin Transformers for Semantic Segmentation of Brain Tumors in MRI Images. *arXiv*, 2022. arXiv:2201.01266.