

# Dynamic MRI Reconstruction

沈哲灏\*

Shanghaitech University

2021533110

shenzhh@shanghaitech.edu.cn

张逸清\*

Shanghaitech University

2022591020

zhangyq22022@shanghaitech.edu.cn

曹熠辉\*

Shanghaitech University

2021533029

caoyh3@shanghaitech.edu.cn

黄之杰\*

Shanghaitech University

2020533147

huangzhj1@shanghaitech.edu.cn

## ACM Reference Format:

Zhehao Shen\*, Yihui Cao\*, Yiqing Zhang\*, and Zhijie Huang\*. 2024. Dynamic MRI Reconstruction.

## 1 INTRODUCTION

Magnetic Resonance Imaging (MRI) reconstruction is a medical imaging technique that creates visual representations from data obtained through MRI scans, extensively utilized in disease diagnosis and pathological analysis.

Dynamic Magnetic Resonance Imaging (Dynamic MRI) excels at capturing and illustrating changes in the internal structures of the human body over time, providing significant insights for the identification and understanding of rapid physiological processes. Compared to static MRI, dynamic MRI faces the substantial challenge of capturing these internal changes within short time intervals to maintain high temporal resolution. This requirement often results in scans with fewer signals than their static counterparts, leading to increased noise in the reconstructed images, which can degrade the quality of the images.

Recent studies [1] [5] [6] have demonstrated the effectiveness of using neural network approaches in static MRI reconstruction. These studies have utilized Convolutional Neural Networks (CNNs) and U-Nets to model the denoising process. While these CNN-based methods have achieved impressive results in reconstructing images, they are capable of restoring relatively clear organ information from undersampled MRI images. However, they struggle to accurately represent information at the edges of images and in areas with rapid grayscale changes.

Edges and regions with rapid grayscale changes hold significant practical importance in medical diagnostics. Clear delineation of edge regions helps in the precise identification of lesion locations and shapes. Rapid changes in grayscale typically indicate dramatic shifts in tissue density or composition, which are crucial for distinguishing between different types of tissues and pathological states, such as tumors, inflammation, or degenerative changes. Therefore, improving the representation of MRI images' edge regions and areas with rapid grayscale changes is a critical issue with substantial value in the field of medical diagnostics.

In this project, we were inspired by the effectiveness of CNN-based neural networks in static MRI reconstruction. We utilized a 3D-CNN that encodes both temporal and spatial information in image sequences to achieve dynamic MRI reconstruction. To better guide the network's learning at image edges, we incorporated an edge loss based on the Sobel operator. Additionally, drawing

from advancements in the super-resolution and image stylization fields [2] which enhance neural networks' capabilities in high-frequency areas, we employed a pre-trained VGG19 [4] CNN to extract high-dimensional feature information for implementing Perceptual Loss. This helps the network more accurately represent areas of rapid grayscale changes (high-frequency areas). Extensive experiments were conducted to validate our network's performance and the effectiveness of commonly used training tricks in CNN training. Finally, to further enhance network performance, we were inspired to incorporate Cascade operations [3] into our training process.

In summary, the main achievements of our project include:

- We implemented a neural network architecture based on 3D-CNNs for the reconstruction of dynamic MRI images.
- Innovatively, we integrated both edge loss and perceptual loss into the MRI image reconstruction process, enhancing the network's ability to accurately depict edge regions and areas with significant grayscale changes. This significantly improved the clarity of these regions in the reconstructed images.
- We conducted extensive experiments to validate the network's performance and analyzed the effectiveness of various training tricks used during the CNN training process.
- We further enhanced the reconstruction quality by incorporating Cascade layers into the network architecture.

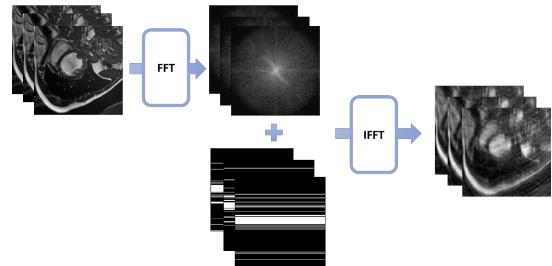


Figure 1: Simulation pipeline

## 2 METHOD

### 2.1 Undersample Simulation

In the training of this network using the designated Cine dataset, we aimed to mimic potential undersampling challenges typical in dynamic MRI reconstruction. This was achieved by converting images to K-space via FFT (Fast Fourier Transform). A series of masks,

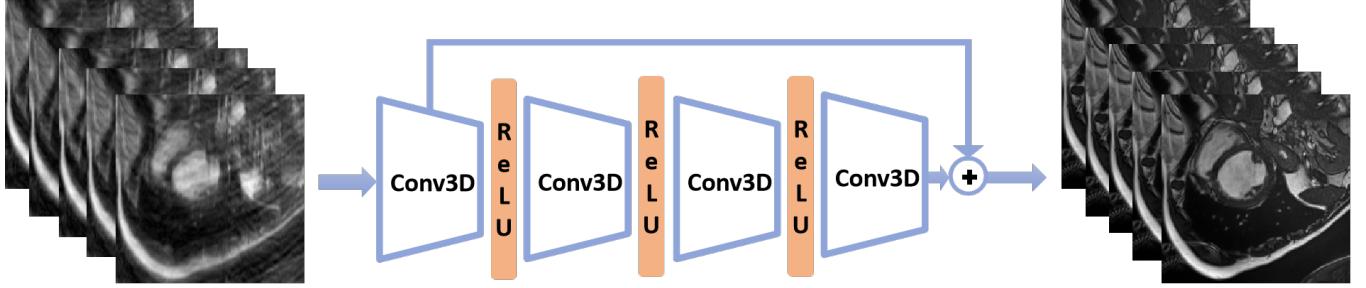


Figure 2: Naive network architecture

created from a constant random seed, were then used to sample these K-space images. Subsequently, these sampled images were reverted to their original form using IFFT (Inverse Fast Fourier Transform) to simulate the appearance of undersampled MRI images. The undersampling simulation process is depicted in Fig. 1

## 2.2 Problem Modeling

In dynamic MRI reconstruction, our problem is thus defined: starting with an undersampled image, the goal is to reconstruct an MRI image that most accurately approximates the ground truth. This objective is encapsulated by the following model equation:

$$y = F_u x + e \quad (1)$$

where:

- $y$  represents the ground truth MRI image,
- $F_u$  denotes the function modeled by the deep learning network, which is trained to reconstruct MRI images from undersampled data,
- $x$  is the blurred MRI image derived from undersampled K-space data,
- $e$  encapsulates the acquisition noise present in the MRI image-capturing process.

## 2.3 Network Architecture Design

**2.3.1 Naive Network.** For the purpose of fulfilling our reconstruction aims, we first designed a naive 3D-CNN network that utilizes a 3D convolutional layer to extract features from time-sequential MRI images. Concurrently, we employed residual connections from the ResNet framework to safeguard against gradient vanishing due to the depth of the network. The architecture of the network is illustrated in Fig. 2

**2.3.2 Complex Network.** In order to better harness the spatiotemporal data within dynamic MRI image sequences, we have evolved our initial network architecture. We have incorporated the upsampling and downsampling techniques inspired by U-net to more effectively amalgamate features from different resolutions. Furthermore, we have increased the depth of our 3D CNN layers to detect more complex, high-dimensional image features. Our enhanced network architecture is showcased in Fig. 6

## 2.4 Loss Design

In addition to the standard Mean Squared Error (MSE) Loss used in MRI reconstruction, we have innovatively integrated Edge Loss and Perceptual Loss. This approach enhances the reconstruction of vital information pertaining to edge definition and areas with swift grayscale transitions, which are essential for medical diagnostics.

**2.4.1 Edge loss.** The Edge Loss employed in our network leverages the Sobel operator for edge detection on both the network's input and the ground truth, subsequently calculating the loss which is shown in Fig. 11. Acknowledging the Sobel operator's limitations with blurred MRI images, we have designed the following piecewise loss function:

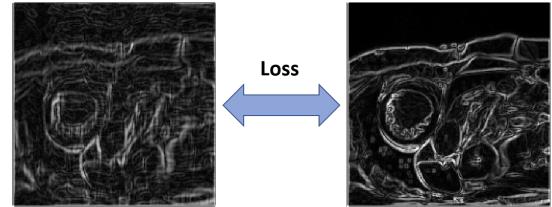


Figure 3: Edge Loss

$$\text{loss} = \begin{cases} \|\text{output} - \text{gt}\|^2, & \text{if iteration} < 100, \\ (1 - \alpha) \cdot \|\text{output} - \text{gt}\|^2 \\ + \alpha \cdot \|\text{Sobel}(\text{output}) - \text{Sobel}(\text{gt})\|^2, & \text{if iteration} \geq 100, \end{cases} \quad (2)$$

where output represents the network's output, gt stands for the ground truth, and  $\alpha$  is a hyperparameter that balances the contribution of the standard loss and the edge loss.

**2.4.2 Perceptual loss.** Through our experimentation, we discovered that relying solely on MSE (Mean Squared Error) loss to constrain our network often results in the equalization and subsequent loss of detail in areas with significant grayscale transitions, leading to blurred image information. This issue arises because MSE loss prioritizes 'similarity' in image space, which can inadvertently smooth over regions containing textures or high-frequency information. Although a high PSNR (Peak Signal-to-Noise Ratio) may be achieved, a high PSNR does not necessarily equate to superior image quality, especially in expressing areas with large grayscale variations.

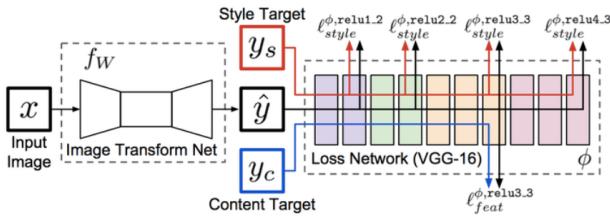


Figure 4: Style-transfer Pipeline

Inspired by the style transfer and super-resolution fields[2], which utilize the VGG-16 image classification network to extract image features for high-quality texture-rich area results on a high-dimensional feature level, shown in Fig. 5, we too have employed a pre-trained VGG-16 network. This enables us to compute the loss on high-dimensional features of the network's output and ground truth, combined with MSE loss. Consequently, we define our loss function as follows:

$$\text{Loss} = (1 - \alpha) \times \| \text{output} - \text{gt} \|^2 + \alpha \times \| F_i(\text{output}) - F_i(\text{gt}) \|^2 \quad (3)$$

where  $F_i$  is a pre-trained VGG-16 network and  $\alpha$  is a hyperparameter that balances the contribution of the standard loss and the perceptual loss.

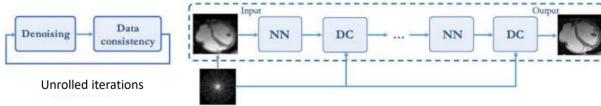


Figure 5: Cascade Pipeline

## 2.5 Cascade

Inspired by the work presented in [3], we have integrated a cascade operation into our pipeline, as depicted in Fig. 5. The cascade approach involves a sequential application of convolutional neural networks (CNNs) where each stage refines the output of the previous one. This method effectively leverages the strengths of multiple processing layers to enhance feature extraction and reconstruction accuracy. By employing a cascading sequence of CNNs, each tailored to progressively refine the reconstruction, we significantly boost the performance of our network. The cascading architecture not only accelerates the reconstruction process but also improves the preservation of anatomical structures, particularly in scenarios of high undersampling. This enhancement is crucial for dynamic MRI where rapid and accurate image reconstruction is essential.

## 3 IMPLEMENTATION DETAILS

Our Naive Network contains 4 Conv3d layers and a residual connection while our complex contains a downsample layer, 8 Conv3d layers, an upsample layer a residual connection.

We all use Adam optimizer with `weight_decay`. We split the 200 data into 5:1:2 train(125), val(25), and test(50) since 200 cannot be divided evenly by 7.

Since the edge loss and perceptual loss will somehow make the

training speed slower, we experiment by training 300 iterations (which almost converge). In other experiments, we train 500 iterations.

We adjusted our hyperparameters based on the performance observed on the validation set as  $\alpha = 0.2$ ,  $\text{weight\_decay} = 1e - 8$ , and subsequently computed the metrics for our subsequent experiments on the test set.

## 4 EXPERIMENTS

### 4.1 Evaluation of PSNR and SSIM Metrics for Simple and Complex Models

In this section, we evaluate the performance metrics of Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) for simple and complex models used in image reconstruction, compared with the baseline metrics measured on images with artifacts. The detailed results are presented in Table 1.

Table 1: PSNR and SSIM Metrics in Test Set Before and After Reconstruction

Model Type	Metric	Mean	Std Dev
Original	PSNR	20.75	0.582
	SSIM	0.695	0.079
Simple Model	PSNR	28.66	0.73
	SSIM	0.886	0.042
Complex Model	PSNR	31.12	0.87
	SSIM	0.934	0.020

The baseline images displayed initial PSNR and SSIM values of 20.75 and 0.695, respectively, with the PSNR exhibiting a standard deviation of 0.582. After applying the simple model for image reconstruction over 500 epochs, improvements were evident with a mean PSNR rising to 28.66 and SSIM to 0.886, with corresponding standard deviations of 0.73 and 0.042. The complex model demonstrated even more substantial enhancements, achieving a mean PSNR of 31.12 dB and an SSIM of 0.934, while maintaining lower variability with standard deviations of 0.87 for PSNR and 0.020 for SSIM. These results highlight the effectiveness of both models in enhancing the quality of images post-reconstruction.

Following the quantitative analysis, we provide a qualitative comparison to demonstrate the distinct effects of the simple and complex models on image reconstruction. Fig. 7 and Fig. 8 display a series of images that showcase the effects of each model separately. This side-by-side comparison underscores the differences in image quality enhancements achieved by each model, highlighting the superior clarity and detail recovery offered by the complex model as compared to the simple model.

In addition to the visual comparisons, we also present the training dynamics of each model through their respective loss curves, illustrated in Fig. 9 and Fig. 10. These curves plot the reduction in loss over the 500 epochs of training for each model, providing insights into their learning efficiency and stability. The loss curve shows that the loss of the complex model is smaller than that of the simple model at the time of convergence.

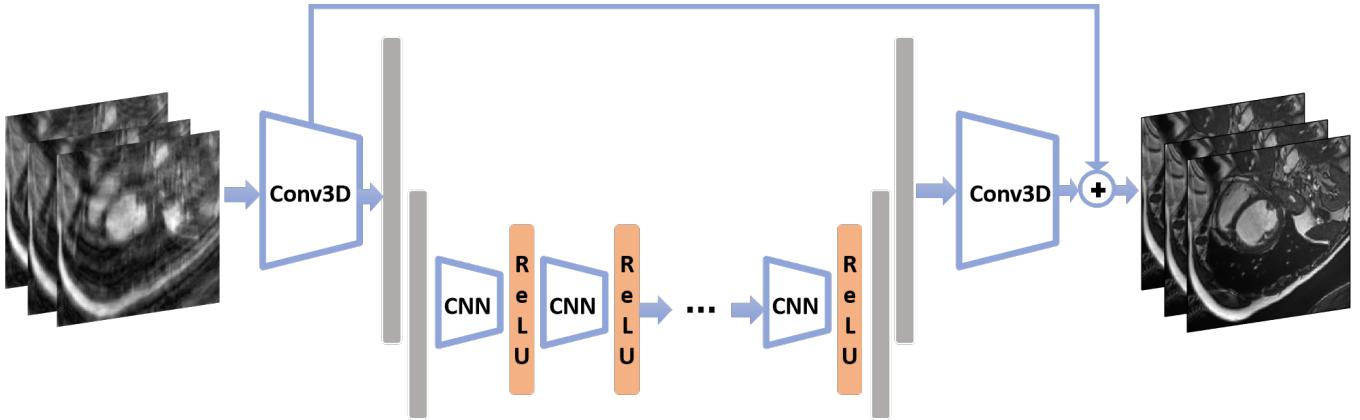


Figure 6: Improved network architecture

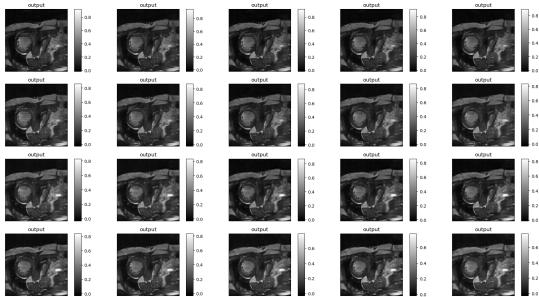


Figure 7: reconstruction results with simple moodel

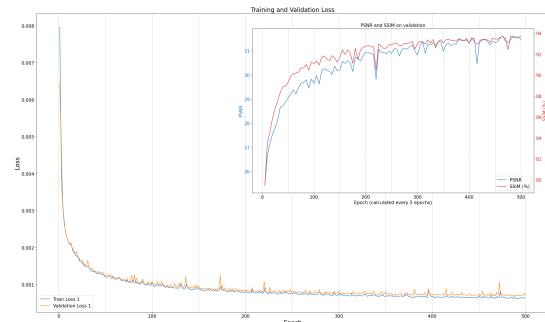


Figure 10: complex model loss curve

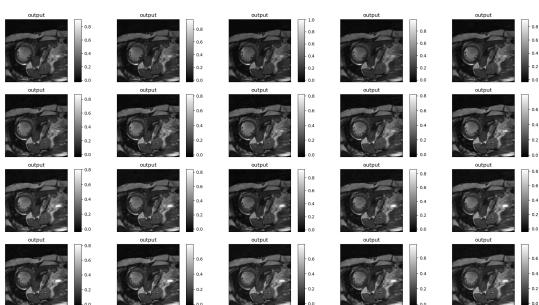


Figure 8: reconstruction results with complex moodel

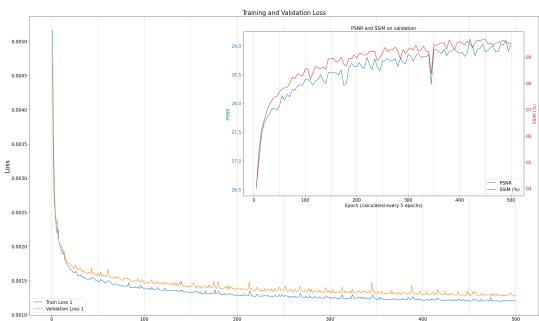


Figure 9: simple model loss curve

## 4.2 Evaluation for Edge loss and Perceptual loss

In this section, we evaluate the performance metrics of Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) for the complex models with and without edge loss and perceptual loss. The detailed results are presented in Table 2, in this experiment we train the model in 300 iterations.

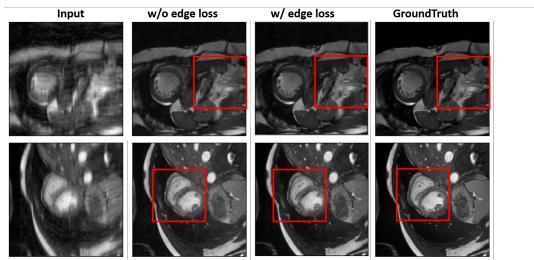
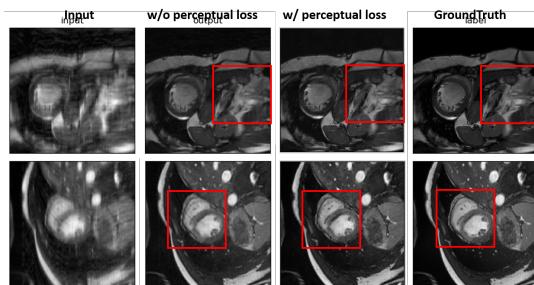
After training 300 times with the addition of Perceptual loss and Edge loss, the results as shown in Fig. 11 and Fig. 12 indicate significant improvements. Incorporating Edge loss has notably enhanced the clarity at the boundaries of organ edges. Furthermore, the inclusion of Perceptual loss has eliminated the previous blurriness in areas of rapid grayscale changes, leading to a more accurate restoration of the original image's rapid grayscale transitions.

## 4.3 Impact of Dropout and Dynamic Learning Rate on Model Performance

In this analysis, we examine the impact of implementing a dropout rate of 0.4 on the performance of simple and complex models after 300 epochs of training. Dropout, a regularization technique, is used to prevent overfitting by randomly omitting a portion of feature detectors during the training process. The results, detailed in Table 3, illustrate the effects of this approach on image reconstruction metrics such as PSNR and SSIM. And

**Table 2: PSNR and SSIM Metrics in Test Set Before and After Reconstruction**

Model Type	Metric	Mean
Complex Model	PSNR	30.311
	SSIM	0.904
Model With Edge Loss	PSNR	30.612
	SSIM	0.928
Model With Perceptual Loss	PSNR	30.939
	SSIM	0.959

**Figure 11: Qualitative evaluation of edge loss****Figure 12: Qualitative evaluation of Perceptual loss****Table 3: Impact of Dropout (0.4) on PSNR and SSIM in Test Set After 300 Epochs**

Model Type	Metric	Mean	Std Dev
Complex Model	PSNR	28.74	0.42
	SSIM	0.889	0.038
Simple Model	PSNR	27.39	0.92
	SSIM	0.861	0.041

The complex model exhibited a mean PSNR of 28.74 and an SSIM of 0.889 after employing a dropout rate of 0.4, with standard deviations indicating moderate variability in the reconstruction quality. Similarly, the simple model recorded a mean PSNR of 27.39 and an SSIM of 0.861, demonstrating slightly higher variability in results, as reflected in the larger standard deviations. These metrics suggest that while dropout helps in regularizing the training process, the extent of its impact can vary between different model complexities.

To assess the effectiveness of learning rate strategies, we compared the performance of a complex model trained with a fixed learning rate using Stochastic Gradient Descent (SGD) to previous results obtained with a dynamic learning rate optimizer, Adam. The complex model was trained for 500 epochs using an SGD optimizer with a fixed learning rate of 0.01. The results are summarized in Table 4.

**Table 4: Impact of Fixed Learning Rate SGD on PSNR and SSIM After 500 Epochs**

Learning Rate Strategy	Metric	Mean	Std Dev
SGD (lr=0.01)	PSNR	22.28	0.48
	SSIM	0.708	0.076

The performance metrics indicate that using a fixed learning rate of 0.01 with SGD resulted in a significantly lower mean PSNR of 22.28 and SSIM of 0.708 when compared to the outcomes achieved with Adam, which dynamically adjusts the learning rate during training. The standard deviations also suggest a higher variability in the results, further highlighting the limitations of using a fixed learning rate for complex model training. This comparison underscores the importance of choosing an appropriate learning rate strategy to enhance model performance and stability during training.

#### 4.4 Evaluating the Impact of Batch Normalization on Model Performance

In this study, we explore the impact of Batch Normalization (BatchNorm) on the performance of simple and complex models trained over 500 epochs. Batch Normalization, a technique aimed at improving training speed and model stability, normalizes the input layer by adjusting and scaling activations. The results, presented in Table 5, demonstrate the effects of BatchNorm on image reconstruction metrics, specifically Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM).

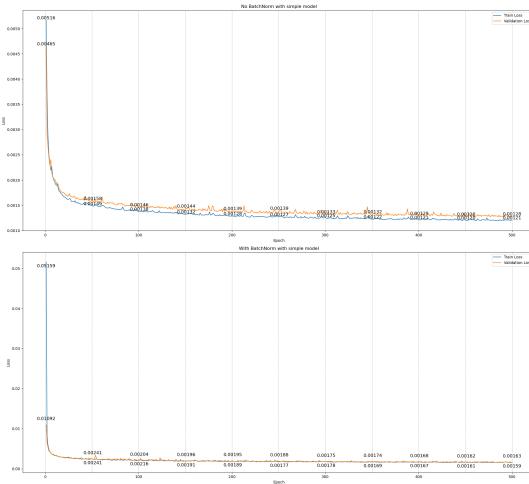
**Table 5: Impact of Batch Normalization on PSNR and SSIM in Test Set After 500 Epochs**

Model Type	Metric	Mean	Std Dev
Simple Model	PSNR	27.68	0.627
	SSIM	0.865	0.043
Complex Model	PSNR	30.91	0.765
	SSIM	0.927	0.030

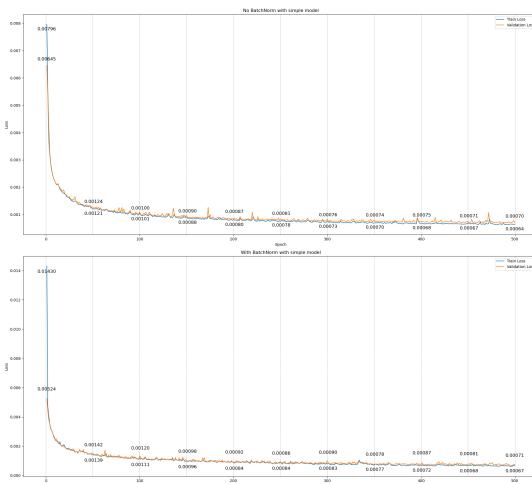
The implementation of Batch Normalization in the simple model resulted in a mean PSNR of 27.68 and an SSIM of 0.865, with moderate variability in the results. For the complex model, BatchNorm led to a higher mean PSNR of 30.91 dB and an SSIM of 0.927, indicating a significant improvement in image quality and consistency. These findings underscore the efficacy of Batch Normalization in enhancing the overall performance and robustness of deep learning models during the reconstruction process.

And we find that after incorporating Batch Normalization, the convergence speed of the model significantly improved. Batch

Normalization normalizes the inputs for each mini-batch, helping to reduce internal covariate shift and thus stabilizing the training process. This stability not only allows the use of higher learning rates but also reduces the dependency on fine-tuning the initial weights and parameters. As a result, the model can achieve or exceed previous performance levels in fewer training epochs, leading to faster convergence and enhanced training efficiency, the training loss curves are shown in Fig. 13 and Fig. 14.



**Figure 13:** loss curve with batch normalization(simple model)



**Figure 14:** loss curve with batch normalization(complex model)

#### 4.5 Assessing the Impact of Cascade Networks on Image Reconstruction Performance

In this section, we investigate the impact of employing cascade networks within a simple model framework on image reconstruction

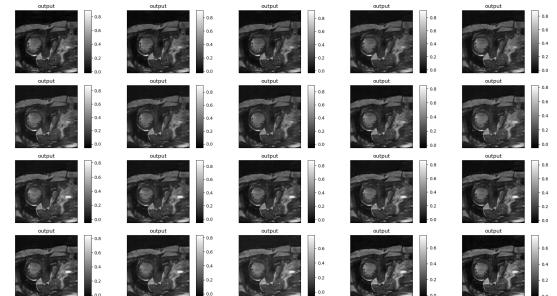
quality. Cascade networks enhance reconstruction performance by iteratively refining the outputs at each stage. We conducted experiments with different levels of iteration to assess how cascading influences the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). The experimental results are presented in Table 6.

**Table 6: Impact of Cascade Networks on PSNR and SSIM in Test Set After 100 Epochs**

Iteration Level	Metric	Mean	Std Dev
No Iteration	PSNR	28.04	0.975
	SSIM	0.877	0.042
2 Iterations	PSNR	28.96	0.633
	SSIM	0.899	0.038
3 Iterations	PSNR	29.76	0.379
	SSIM	0.913	0.033

The data indicates a noticeable improvement in image quality metrics with increasing iterations in the cascade architecture. Starting from a baseline with no iteration, which achieved a PSNR of 28.04 dB and an SSIM of 0.877, the model shows slight fluctuation in performance with one iteration. However, with the addition of two and three iterations, particularly with the integration of DC layers, both PSNR and SSIM significantly improve. The highest mean PSNR recorded was 29.76 dB, and the highest SSIM was 0.913, both with three iterations, demonstrating the model's enhanced ability to recover high-quality images. These findings suggest that cascade networks, particularly when combined with DC layers, effectively improve the robustness and quality of the reconstruction process.

To further elucidate the practical benefits of using cascade networks in image reconstruction, we present a visual comparison of reconstructed images obtained at various iteration levels within the cascade framework. Fig. 15, Fig. 16 and Fig. 17 showcase reconstructed images corresponding to each iteration level discussed in the previous analysis. This visual comparison allows for a direct assessment of the improvements in image quality and detail as the number of iterations increases. And the Metrics curves during training and the training loss curves are shown in Fig. 18 and Fig. 19



**Figure 15: No Iteration**

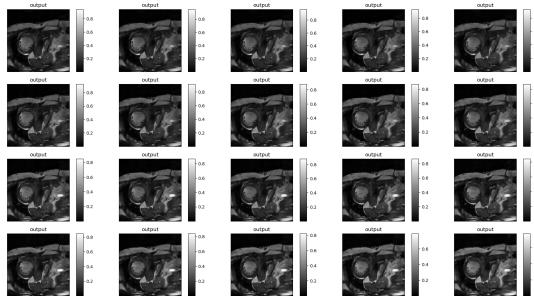


Figure 16: 2 Iterations

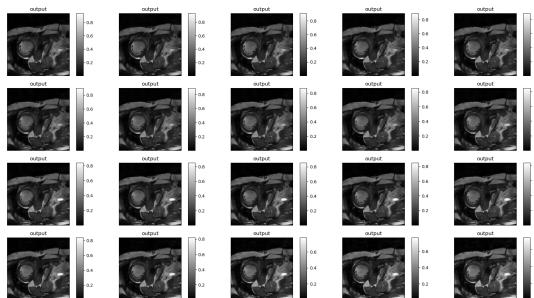


Figure 17: 3 Iterations

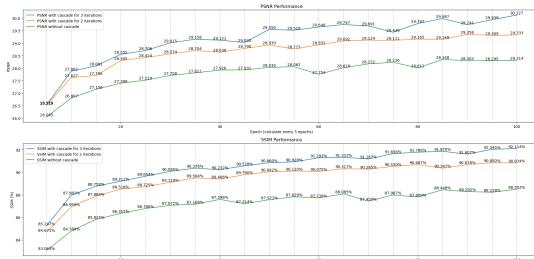


Figure 18: Metrics during training in different iterations

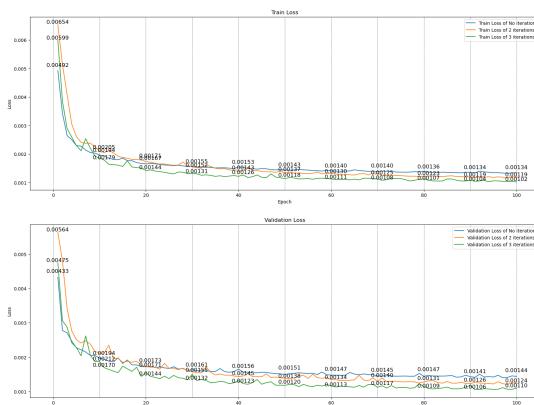


Figure 19: Loss during training in different iterations

This series of images vividly illustrates the incremental improvements in image clarity, detail, and overall quality. The images reconstructed with no iteration serve as the baseline, showing a decent reconstruction. However, each subsequent iteration, especially with the inclusion of DC layers, progressively refines and enhances the visual quality, culminating in the most detailed and clear images at three iterations. This visual evidence strongly supports the effectiveness of the cascade network approach in improving the fidelity and accuracy of image reconstruction tasks.

## REFERENCES

- [1] Andreas Hauptmann, Simon Arridge, Felix Lucka, Vivek Muthurangu, and Jennifer A Steed. Real-time cardiovascular mr with spatio-temporal artifact suppression using deep learning—proof of concept in congenital heart disease. *Magnetic resonance in medicine*, 81(2):1143–1156, 2019.
- [2] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II*, pages 694–711. Springer, 2016.
- [3] Jo Schlemper, Jose Caballero, Joseph V Hajnal, Anthony N Price, and Daniel Rueckert. A deep cascade of convolutional neural networks for dynamic mr image reconstruction. *IEEE transactions on Medical Imaging*, 37(2):491–503, 2017.
- [4] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.
- [5] Shanshan Wang, Zhenghang Su, Leslie Ying, Xi Peng, Shun Zhu, Feng Liang, Da-gan Feng, and Dong Liang. Accelerating magnetic resonance imaging via deep learning. In *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, pages 514–517, 2016.
- [6] 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):487–492, 2018.