Deep convolutional neural networks for accelerated dynamic magnetic resonance imaging

Christopher M. Sandino[1] Neerav Dixit[1] Joseph Y. Cheng[2] Shreyas S. Vasanawala[2]
[1]Department of Electrical Engineering, Stanford University [2]Department of Radiology, Stanford University
{sandino, ndixit}@stanford.edu

Abstract

Dynamic magnetic resonance imaging (MRI) scans can be accelerated by utilizing compressed sensing (CS) reconstruction methods that allow for diagnostic quality images to be generated from undersampled data. Unfortunately, CS reconstruction is time-consuming, requiring hours between a dynamic MRI scan and image availability for diagnosis. In this work, we train a convolutional neural network (CNN) to perform fast reconstruction of severely undersampled dynamic cardiac MRI data, and we explore the utility of CNNs for further accelerating dynamic MRI scan times. Compared to state-of-the-art CS reconstruction techniques, our CNN achieves reconstruction speeds that are 150x faster without significant loss of image quality. Additionally, preliminary results suggest that CNNs may allow scan times that are 2x faster than those allowed by CS.

1. Introduction

In dynamic magnetic resonance imaging (MRI), organs such as the heart must be continuously scanned over long periods of time to obtain a series of images that characterize anatomy and motion over time. To accelerate the long scan times, compressed sensing (CS) reconstruction schemes, which exploit the redundancy of dynamic MRI acquisitions in space and time, are utilized to achieve high spatiotemporal resolution with undersampled data (see Figure 1) [1, 2, 3]. However, CS reconstruction times are long because these schemes are based on non-linear inverse problems, which must be iteratively solved. This presents a problem for clinical translation of CS techniques, as they extend the period of time between exam and diagnosis by hours.

Convolutional neural networks (CNNs) have previously been used to model medical image reconstruction problems in computed tomography [4] and MRI [5]. CNN-based reconstruction is non-iterative after training occurs and is therefore much faster than conventional CS methods (by $\sim$100-1000x). These techniques also leverage previous exam data to learn spatial structure of anatomy and typical image artifacts caused by undersampling. These attributes allow CNN-based methods to reconstruct highly undersampled data at higher fidelity than CS schemes in certain cases. While deep learning techniques have been applied to a variety of medical imaging reconstruction problems, they have not yet been used to reconstruct dynamic MRI data.

In this paper, we extend previous work done by Jin et al. [4] and propose a deep dynamic MRI reconstruction framework that uses CNNs to learn a mapping between trivial reconstructions of undersampled data and diagnostic quality reconstructed images. We show that our framework accelerates reconstruction time and potentially scan time when compared against a state-of-the-art compressed sensing algorithm.

2. Background

2.1. MR Image Reconstruction

In MRI, images are reconstructed from raw data acquired in the Fourier domain. If an image's Fourier data is fully

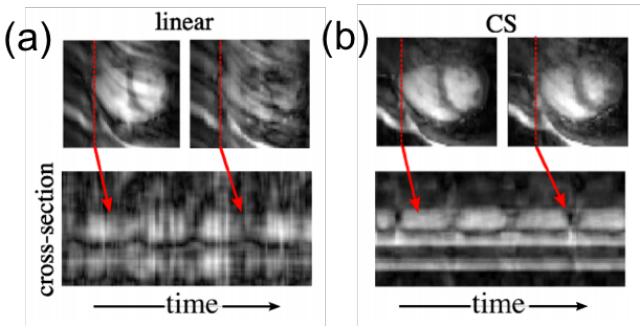

Figure 1. The MRI acquisition process is not fast enough to take fully sampled snapshots in real-time. Instead, data is acquired over multiple heartbeats, and a CS reconstruction is used to suppress image artifacts caused by sub-Nyquist sampling. [Figure courtesy of Michael Lustig, UC Berkeley]

J. Y. Cheng and S. S. Vasanawala are not enrolled in CS231N

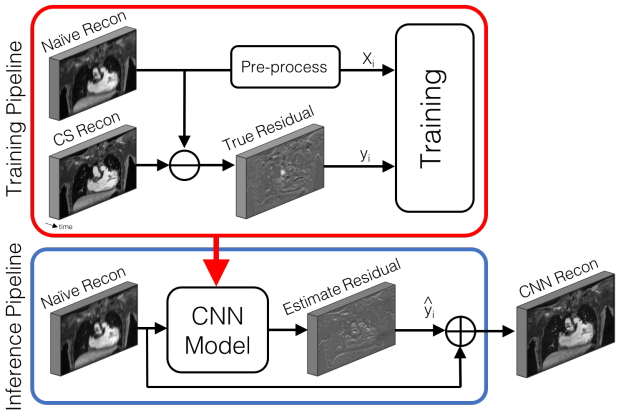

Figure 2. Workflow for CNN training and inference

sampled (i.e. sampled at the Nyquist rate), then the image can be linearly reconstructed without aliasing artifacts with a simple 2-D Fourier transform (2DFT). When the Fourier data is sampled below the Nyquist rate, CS techniques employ prior information to estimate the true image data m from raw data b. This is done by iteratively solving an optimization problem of the following form:

$$\underset{m}{\text{minimize}} \, ||\mathcal{F}_s m - b||_2^2 + \lambda ||\phi(m)||_1 \qquad (1)$$

where $\mathcal{F}_s$ is the undersampled Fourier transform operator, and ϕ is some sparsifying transform [6]. Some commonly used functions for ϕ (priors) include finite differences and multiresolutional Wavelet transforms. A global optimum $\hat{m}$ can be found by iterative algorithms such as ISTA [7] or ADMM [8]; however, these become slow when reconstructing high-dimensional data. The goal of this project is to train a CNN to quickly approximate the image data m from $|\mathcal{F}_s^{-1} b|$, the magnitude of undersampled raw data that has been linearly reconstructed using the 2DFT.

2.2. Related Work

Recent work suggests that neural networks are mathematically well-suited for efficiently solving compressed sensing problems. The basic unit of a neural net consists of a multiplicative layer followed by a point-wise linear rectifier (ReLU). Analogously, each iteration of the ISTA algorithm consists of an affine transformation and a point-wise soft-thresholding step (similar to ReLU). Therefore, a neural network is structured similarly to an unrolled iterative shrinkage algorithm. Gregor et al. used this idea to train a network to approximate solutions to Eqn 1 with 20x fewer iterations than ISTA [9]. This analysis/result has provided the mathematical basis for deep learning approaches to be applied to various classes of inverse problems such as denoising [10], deconvolution [11], and super-resolution [12].

Additionally, neural networks have been used to efficiently and accurately reconstruct medical imaging data. For example, Jin et al. used a CNN (FBPConvNet) to model computed tomography (CT) reconstruction by learning a mapping between undersampled data and ground truth images [4]. They showed that FBPConvNet is able to reconstruct 2-D CT data at 1000x faster speeds than state-of-the-art methods without loss of image quality. They also showed that it can reconstruct images more accurately than compressed sensing methods for further rates of undersampling. This is likely because CNNs are able to learn better priors by training on previous exam data, whereas traditional CS methods rely on simpler priors (i.e. sparse finite differences). Similar methods have been successfully applied to MR image reconstruction using variational networks [5] and generative adversarial networks [13, 14]. Despite these advances, deep learning methods have not yet been applied to model dynamic MR image reconstruction. In this paper, we apply the technique proposed by Jin et al. to dynamic cardiac MR image reconstruction and compare it to CS techniques.

3. Dataset

The dataset used for this work was provided by Dr. Shreyas Vasanawala from Lucille Packard Children's Hospital. The dataset contains 3-D cardiac time series that were acquired from six patients with IRB approval. All patients were between 4-6 years old, except for one who was 3 months old. Three of these time series were used for training, one for validation, and two for testing. Each 3-D dataset contains 896 2-D slices, and each time series contains 20 frames representing different phases of the cardiac cycle. Because our CNN treats each 2-D time series independently, we have 2688 samples for training, 896 for validation, and 1792 for testing.

It is important to note that one limitation of this dataset is that it is not fully sampled per the Nyquist limit. All of the images were acquired using a 15x factor of acceleration during scanning and therefore cannot be considered as "ground truth". However, each dataset is considered by a radiologist to be of diagnostic quality when reconstructed using l_1-ESPIRiT, a popular CS technique [3].

3.1. Data Pre-processing

Dynamic cardiac MRI acquisitions are done using parallel imaging, in which Fourier data is acquired by multiple receivers at different spatial locations. For the purposes of CNN reconstruction, the raw Fourier data is transformed into the image domain since it is a more natural space for convolutional layers to exploit spatial structure. This also provides a warm start to the CNN, as it does not have to learn the Fourier transform to reconstruct images. To convert the raw Fourier data to the image domain, we performed a naive reconstruction in which a simple 2DFT was performed after substituting zeros for the data points that

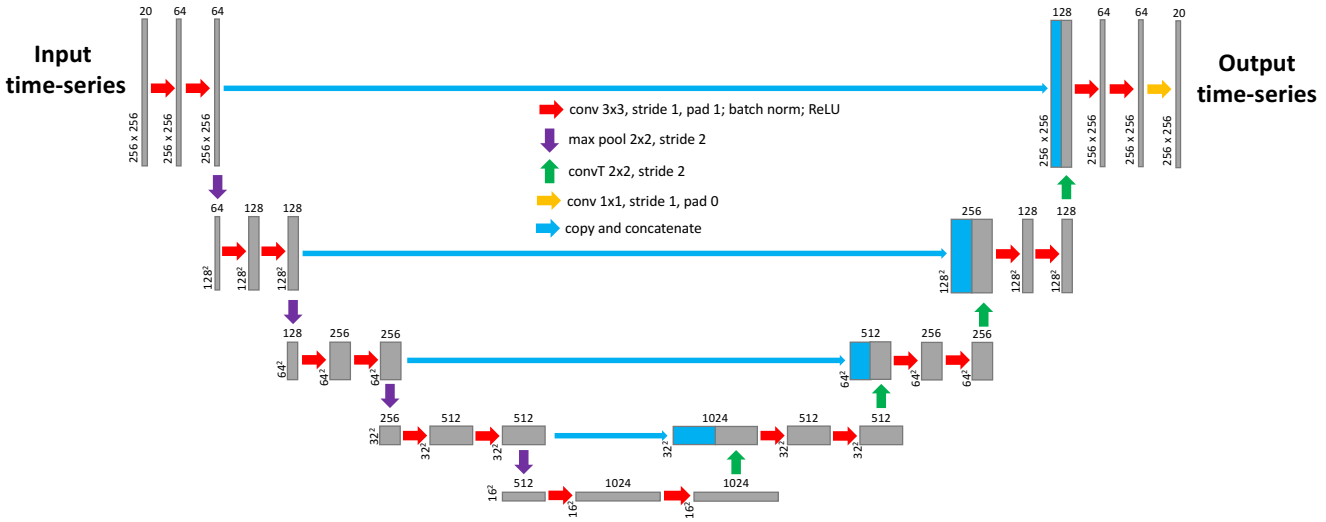

Figure 3. Modified U-Net CNN architecture

were not acquired due to undersampling. This reconstruction is very fast when the raw data is sampled on a Cartesian grid in Fourier space, as it was for the examples shown in this work. We also sought to simplify the CNN input by combining the data from the multiple receiver coils into a single input image. This was done using a sum-of-squares combination of the naively reconstructed images from each receiver.

This manipulation of the raw data left us with a single 3-D time series in the image domain for each patient. Due to limited dataset size and computational expense of processing 3-D image data with CNNs, we separated the 3-D volume into a series of 2-D slices that were evaluated independently. The temporal dimension was used as the channel dimension of the CNN input. The input data dimensions (H, W, T) were made consistent across examples by linear interpolation. Each example was normalized to have signal intensities in the range $[0, 1000]$ and then zero-centered by subtracting the empirical mean across all training examples to stabilize neuron behavior.

3.2. Restrospective Undersampling

To evaluate the utility of CNN reconstruction in accelerating scan times, raw MRI data was discarded before performing the naive reconstruction used to create the input to the CNN. Discarding MRI data simulates performing a shorter scan in which less data in acquired. Therefore, the factor by which data is undersampled is equivalent to the factor by which scan time is accelerated. This restrospective undersampling was done by applying a variable-density sampling mask in the Fourier domain.

4. Methods

4.1. Training and Inference Workflow

For the purposes of training the CNN, we considered the l_1-ESPIRiT reconstruction as the ground truth reconstruction of the undersampled data. During training, we sought to learn a set of residual images of dimension (H, W, T) corresponding to the difference between the l_1-ESPIRiT ground truth reconstruction and the naive reconstruction discussed above. During inference, the full CNN reconstruction is obtained by adding the learned residual to the naively reconstructed image (see Figure 2).

4.2. CNN Architecture

U-Net is a CNN architecture that was initially applied to biomedical image segmentation but has performed well on medical image reconstruction problems [4, 15]. We implemented a modified version of the U-Net architecture, shown in Figure 3. Each level in the first half of the U-Net passes its input through two blocks consisting of a 3x3 convolution layer, batch normalization, and a ReLU layer. With each successive level, height and width dimensions are halved by 2x2 max pooling, and the depths of the convolution filters double. In the second half of the U-Net, these changes in dimension are reversed in a symmetric fashion using transpose convolution. Two blocks of 3x3 convolution, batch normalization, and ReLU are again applied at each level. Some filter outputs are passed directly from the first half of the U-Net to the second half and concatenated to the transpose convolution outputs. This may improve gradient behavior by creating shorter paths for backpropagation of gradients to the input. The last step in the network uses a 1x1 convolution layer to produce the output images.

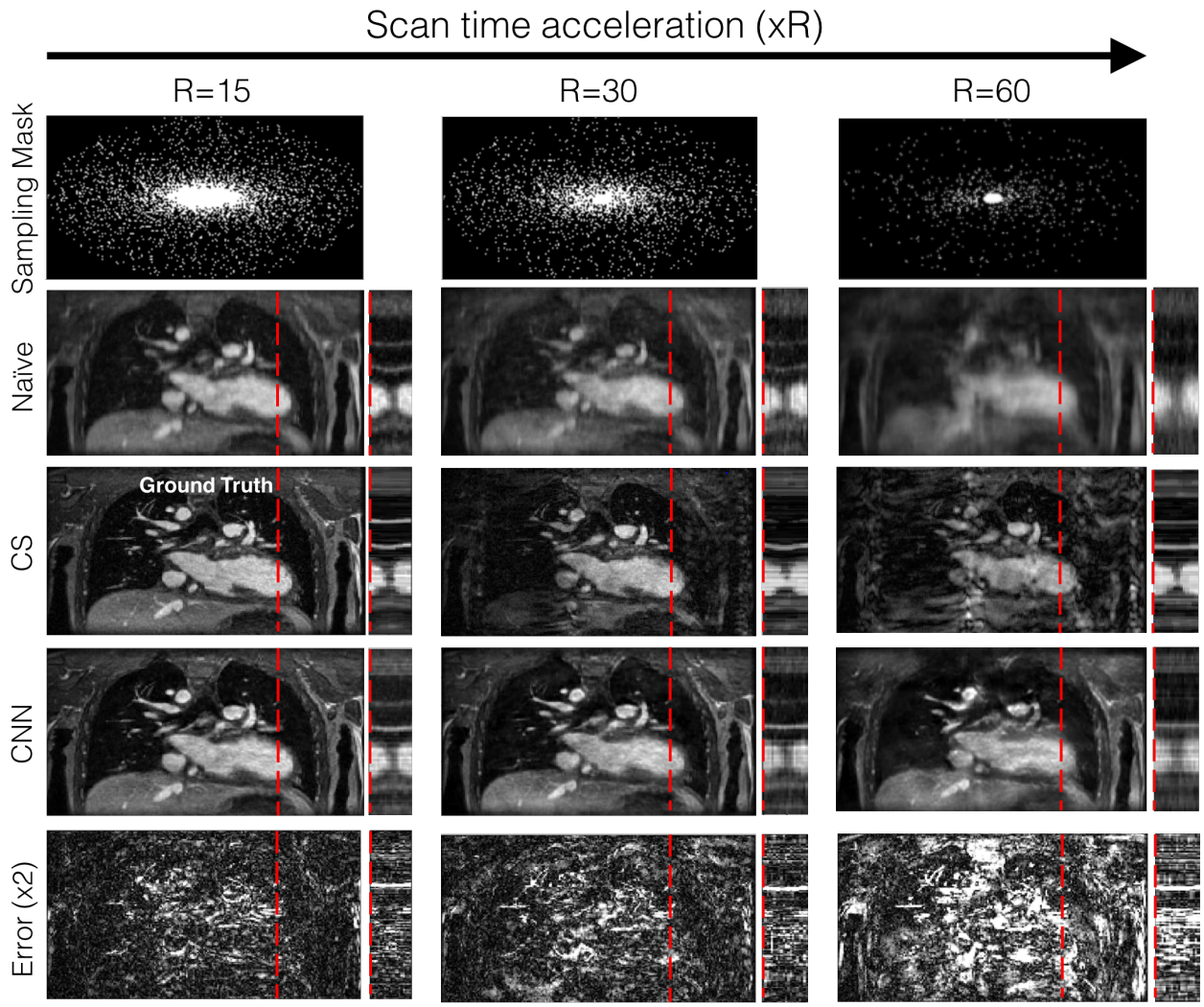

Figure 4. Cardiac images generated using various reconstruction techniques for different factors of scan time acceleration. The images are displayed alongside x-t plots to visualize how pixels along the dashed line change through the time series. Error images show the absolute value of the difference between the CNN reconstructions for each acceleration factor and the CS reconstruction for R=15.

4.3. Training Parameters

We evaluated the use of several different loss functions to quantify the difference between the learned residuals $\hat{y}_i$ and the actual residuals y_i. In certain cases, a regularization term that penalizes the squared l_2 norm of the weights of the convolution and transpose convolution layers was also used to prevent the CNN model from overfitting to the training data.

One loss function was based on the l_2 norm of the difference between the learned and actual residuals.

$$loss_{l_2} = \frac{1}{N}\sum_{i=1}^{N} ||y_i - \hat{y}_i||_2^2 + \lambda||W||_2^2 \qquad (2)$$

Another was based on the l_1 norm of the difference between the residuals in the Fourier domain. This loss function was used with the intention of preventing blurriness in the reconstructed images by providing equal weighting to both high and low spatial frequencies.

$$loss_{l_1(\mathcal{F})} = \frac{1}{N}\sum_{i=1}^{N} ||\mathcal{F}y_i - \mathcal{F}\hat{y}_i||_1 + \lambda||W||_2^2 \qquad (3)$$

A common metric used to quantify the perceptual similarity between images is the structural similarity (SSIM) index [16]. A loss function that penalized structural dissimilarity (DSSIM) was also used to train the model.

$$loss_{SSIM} = \frac{1}{N}\sum_{i=1}^{N} \frac{1 - \text{SSIM}(y_i, \hat{y}_i)}{2} \qquad (4)$$

The Adam algorithm [17] with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ was used for gradient descent when training. In

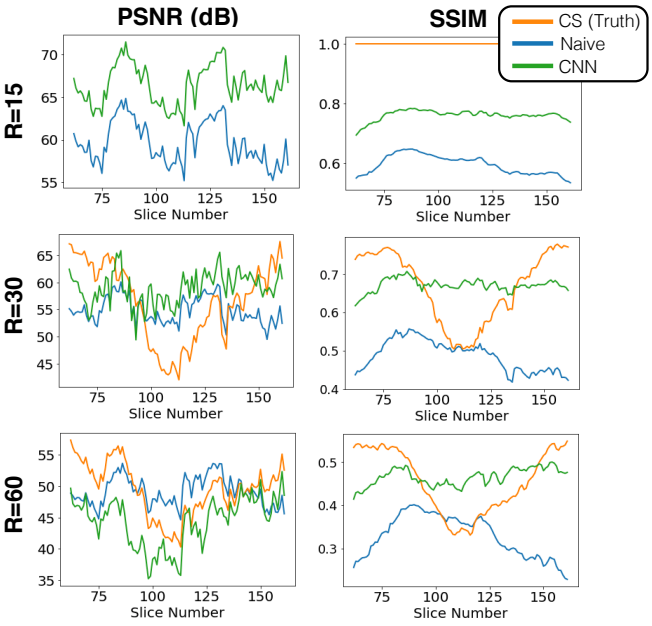

Figure 5. Plots of PSNR and SSIM over the center 100 slices of a test dataset for different factors of scan time acceleration (R).

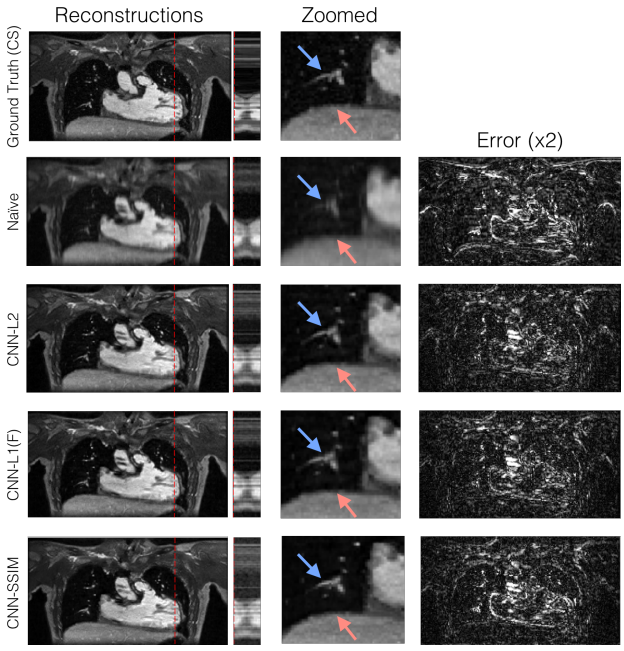

Figure 6. Cardiac images reconstructed by CNNs that were trained using different loss functions. Alongside the images are x-t plots to visualize temporal dynamics, zoomed images showing the details of a pulmonary vessel (blue arrows) and the diaphragm border (red arrows), and an image of the error of each reconstruction with respect to the l_1-ESPIRiT reconstruction.

Adam, the step size taken at each iteration of the gradient descent is influenced by the accumulated first and second moments of the gradient in a manner determined by the β parameters. The learning rate for gradient descent and the regularization coefficient λ used in the loss function were chosen by observing training and validation loss curves over multiple epochs of training. The CNN architecture was created in TensorFlow [18], and the code for training and inference was built from the framework provided in Assignment 2 of CS231N at Stanford University.[1]

4.4. Image Quality Evaluation

To quantitatively evaluate the CNN reconstructions, SSIM values between the resulting images and the l_1-ESPIRiT reconstruction were calculated. Peak signal-to-noise ratios (PSNR) of the CNN-reconstructed images with respect to the l_1-ESPIRiT-reconstructed images were also computed. In MRI, qualitative evaluations of images are especially important to ensure that key features that impact diagnosis are reconstructed accurately. We qualitatively noted the differences in behavior between the CNN and CS cardiac reconstructions by closely observing the features of several 2-D time series.

5. Experiments/Results

CS and CNN reconstructions were performed on NVIDIA GTX Titan X and NVIDIA Tesla K80 graphics cards respectively. On average, CS reconstructions took 2 hours and 19 minutes, whereas CNN reconstructions took 55 seconds.

5.1. CNN Reconstruction for Scan Time Acceleration

To determine whether using a CNN reconstruction can allow for scan time acceleration, we trained CNNs to generate images of diagnostic quality from naive reconstructions of retrospectively undersampled data. Our dataset contains raw data that is already undersampled by a factor of 15, and datasets with undersampling factors of 30 and 60 were obtained by applying the sampling masks shown in Figure 4 in the Fourier domain. For each of these undersampling factors, a CNN was trained to produce the residual between the naive reconstruction of the raw data and the l_1-ESPIRiT reconstrution of the R=15 data. The l_2-based loss function in Eqn 2 was used when training the CNNs. To compare the CS and CNN performance, l_1-ESPIRiT reconstructions were also computed for each undersampling factor.

Example slices of the CNN and CS reconstructions are shown in Figure 4. For R=15, CS and CNN reconstructions are visually similar. The CNN is able to accurately depict small, stationary vessels in the lung. However, moving structures such as the heart wall are slightly blurred compared to the ground truth. For R=30 and 60, the CS

[1] cs231n.stanford.edu/assignments/2017/spring1617_assignment2.zip

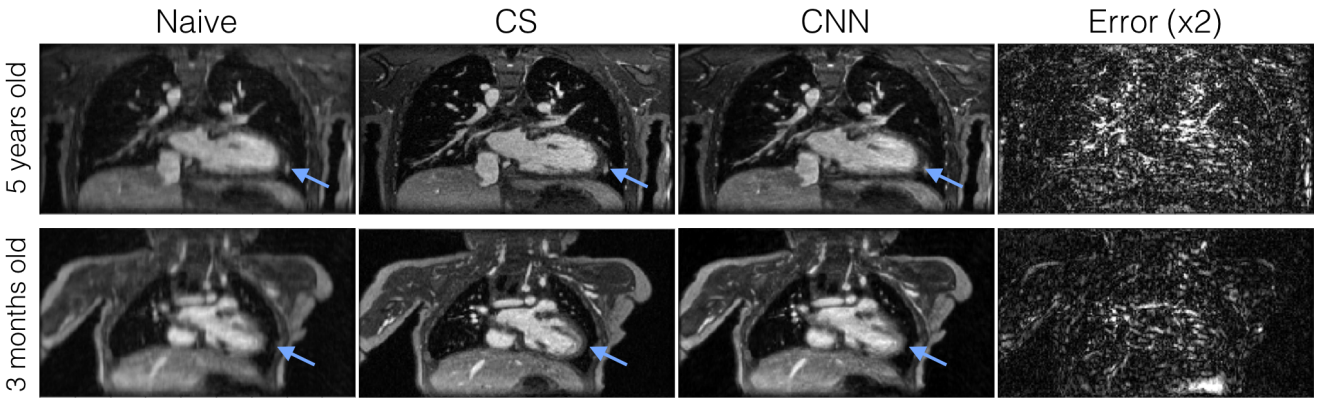

Figure 7. Reconstructed cardiac images of a 5 year-old patient and a 3 month-old patient from data acquired with R=15 acceleration. CNN training was performed on data from patients between 4 and 6 years old. Degraded image quality is depicted in the heart wall (blue arrows) of the 3 month-old patient data.

reconstruction image quality begins to degrade. This is because l_1-ESPIRiT is not able to accurately estimate a calibration region from the low frequency components due to excessive undersampling. The image quality in the CNN reconstructions stays relatively constant, although images become blurrier for increasing R.

These reconstructions were tested on MRI data corresponding to a 3-D cardiac time series of a single patient. To evaluate how the reconstructions performed at different 2-D slice layers within this 3-D volume, in Figure 5 PSNR and SSIM are plotted over a series of 100 2-D slices located near the center of the volume. We chose to plot these because typically the clinical region of interest is placed at the center of the scan field of view. The quality of the CS reconstructions varies over the slice number for acceleration rates of 30 and 60, degrading mostly at the center slices. CNN reconstructions are generally better than CS reconstructions except at the outer slices.

5.2. Impact of Loss Function on Image Quality

When training the CNN, the choice of loss function used to quantify the difference between the learned and actual residual determines how the weights in the model are updated. We compared the performance of CNNs trained using different loss functions to determine which one resulted in the best CNN reconstruction.

Images reconstructed with CNNs trained on the loss functions detailed in Eqns 2, 3, and 4 are shown in Figure 6. Each of the CNN reconstructions look visually similar. We noticed that in the CNN-SSIM reconstruction, certain edges and structures (i.e. diaphragm and lung vessels) are better defined. Quantitative results are shown in Table 1. The l_1-based loss had the highest PSNR and SSIM values of the three tested loss functions, although by a small margin.

	PSNR (dB)	SSIM
CS (Truth)	∞	1
Naive	59.56	0.596
CNN-l_2	66.18	0.760
CNN-$l_1(\mathcal{F})$	**66.63**	**0.764**
CNN-SSIM	65.46	0.744

Table 1. Impact of Loss Function Quantitative Results. Shown here are the average PSNR (dB) / SSIM values over the center 100 slices in the R=15 3-D volume reconstruction.

5.3. Generalizing to Varied Patient Anatomies

The medical community is highly skeptical of using deep learning approaches to reconstruct images for diagnostic purposes due to their "black-box" nature. It is unclear whether pre-trained models will be able to accurately reconstruct images depicting patient anatomies and pathologies that were not represented in the training data. A robust model must generalize to highly varied anatomies since including a training example for every possible patient type and disease is infeasible. To explore this idea of generalization, we trained a model on data acquired from 4-6 year-old patients, and then used it to reconstruct data acquired from a 3 month-old patient. The two sets of images are visually different; therefore, we expected reconstruction results for the 3 month-old to look worse than that of the 5 year-old.

The reconstructed images shown in Figure 7 prove our hypothesis to be correct. Although the CNN reconstruction is able to resolve minute vessels in the lungs, it is unable to form definite edges at the heart wall. Quantitatively, the 5 year-old and 3 month-old patient reconstructions have similar average PSNR values (66.18 and 66.61 dB respectively). However, the SSIM of the 5 year-old reconstruction is much higher than that of the 3 month-old (0.760 and 0.579 respectively).

6. Discussion/Conclusion

In this work we have shown that CNNs can be used to model dynamic MR image reconstruction of 2-D cardiac data. Reconstruction time is accelerated by a factor of 150 when compared against l_1-ESPIRiT. Preliminary results show that CNNs may allow scan time to be accelerated by a factor of 2. The CNN produced visually better reconstructions than CS techniques for larger factors of acceleration. However, further studies are necessary to determine if these reconstructions are reliable.

The time plots shown in Figures 4 and 6 indicate that the CNN is not able to resolve temporal dynamics as well as l_1-ESPIRiT. Temporal profiles are visibly sharper for the CS reconstruction because l_1-ESPIRiT uses a strong temporal finite differences regularizer in its cost function. This also explains the streaking artifacts present in both of the temporal profiles of the CS and CNN reconstructions. The available datasets were highly undersampled in time and therefore may not have been adequate ground truths to train on. We hypothesize that having a fully sampled ground truth may improve temporal image quality in our CNN reconstructions. Additionally, image quality may be improved by using 3-D (2-D space and time) convolution kernels, which could locally exploit both spatial and temporal structure. However, a 3-D CNN would be much more computationally intensive and require fewer layers to be able to run on modern GPUs. Alternatively, a recurrent neural network could be used to exploit temporal relationships between successive frames and would be less computationally intensive than a 3-D CNN.

The different loss functions used for CNN training appeared to produce models that performed quantitatively and qualitatively similar CNN reconstructions. In the future, we would like to try other more complex loss functions such as perceptual loss [19] or train GANs for reconstruction [13]. Ultimately, none of these different loss metrics can necessarily capture how radiologists assess image quality. It is important to note that although we tune algorithms to reconstruct images that minimize a certain loss function, they may not produce the best image for diagnosis. The ideal method would be to train the deep reconstruction on a "radiologist loss" network, which maps images to corresponding scores given by a radiologist or committee of radiologists. However, this would likely require a tremendous amount of both data and radiologists' time.

The degradation in the performance of CNN reconstruction when applied to data from a 3 month-old patient rather than a 5 year-old patient indicates that CNN reconstruction does not generalize well to cases that are significantly different than those used for training. Rather than learning the underlying structure of the MRI reconstruction, the CNN appears to be learning priors regarding spatial structure from the training data and applying these priors when performing inference on new data. This is a significant limitation of CNN reconstruction for medical imaging, as there is no guarantee of sufficient performance when presented with novel cases. Changes in the loss function may be able to force the CNN to place more emphasis on learning a more generalizable reconstruction. For example, adding a data consistency term like the one in Eqn 1 to the loss function would place some importance on ensuring that the raw MRI data matches the reconstructed image.

One notable limitation of the proposed CNN method is that it does not make use of parallel imaging. In accelerated MR image reconstruction, most algorithms exploit redundancy across data acquired by different receivers in parallel. Although we were initially provided raw data from the thirty-two receiver channels, this data was combined into one channel using a sum-of-squares operation to simplify computation. Theoretically, a neural network should be able to learn parallel imaging if given multi-channel data, although this would make the model more complex and harder to train.

Deep learning techniques can potentially accelerate the entire magnetic resonance imaging workflow, but there are still many questions about their accuracy and robustness that must be addressed before these techniques can become clinically useful.

7. Acknowledgments

We would like to thank Morteza Mardani and Feiyu Chen for their helpful insights and discussions over the course of this project.

References

[1] J Tsao and P Boesiger. k-t BLAST and k-t SENSE: dynamic MRI with high frame rate exploiting spatiotemporal correlations. *Magn Reson Med*, 2003.

[2] H Jung, K Sung, KS Nayak, and EY Kim. k-t FOCUSS: a general compressed sensing framework for high resolution dynamic MRI. *Magn Reson Med*, 2009.

[3] M Uecker, P Lai, M Murphy, P Virtue, M Elad, JM Pauly, SS Vasanawala, and M Lustig. ESPIRiT eigenvalue approach to autocalibrating parallel MRI: where SENSE meets GRAPPA. *Magn Reson Med*, 71(3):990–1001, 2014.

[4] K Jin, MT McCann, E Froustey, and M Unser. Deep convolutional neural network for inverse problems in imaging. *arXiv:1611.03679*, 2016.

[5] K Hammernik, T Klatzer, E Kobler, and MP Recht. Learning a variational network for reconstruction of accelerated MRI data. *arXiv:1704.00447*, 2017.

[6] M Lustig, D Donoho, and JM Pauly. Sparse MRI: the application of compressed sensing for rapid MR imaging. *Magn Reson Med*, 58(6):1182–1195, 2007.

[7] A Beck and M Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2009.

[8] S Boyd, N Parikh, E Chu, and B Peleato. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 2011.

[9] K Gregor and Y LeCun. Learning fast approximations of sparse coding. *Proceedings of the 27th International Conference on Machine Learning*, pages 399–406, 2010.

[10] J Xie, L Xu, and E Chen. Image denoising and inpainting with deep neural networks. *Advances in Neural Information Processing Systems*, 2012.

[11] L Xu, JSJ Ren, C Liu, and J Jia. Deep convolutional neural network for image deconvolution. *Advances in Neural Information Processing Systems*, 2014.

[12] Z Wang, D Liu, J Yang, and W Han. Deep networks for image super-resolution with sparse prior. *Proceedings of the IEEE*, 2015.

[13] M Mardani, E Gong, JY Cheng, S Vasanawala, G Zaharchuk, M Alley, N Thakur, JM Pauly, and Xing L. Deep generative adversarial networks for compressed sensing automates MRI. *arXiv:1706.00051*, 2017.

[14] S Yu, H Dong, G Yang, G Slabaugh, and PL Dragotti. Deep De-Aliasing for fast compressive sensing MRI. *arXiv:1705.07137*, 2017.

[15] O Ronneberger, P Fischer, and T Brox. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention*, 9351:234–241, 2015.

[16] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[17] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.

[18] M Abadi, A Agarwal, P Barham, and E Brevdo. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv:1603.04467*, 2016.

[19] Justin Johnson, Alexandre Alahi, and Fei-Fei, Li. Perceptual losses for real-time style transfer and super-resolution. *arXiv:1603.08155*, 2016.