The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Compressed Sensing MRI Using a Recursive Dilated Network

Liyan Sun,[†] Zhiwen Fan,[†] Yue Huang, Xinghao Ding,[⋆] John Paisley,[‡]
Fujian Key Laboratory of Sensing and Computing for Smart City, Xiamen University, Fujian, China
[†]The co-first authors contributed equally.
[⋆]Correspondence: dxh@xmu.edu.cn.
[‡]Department of Electrical Engineering, Columbia University, New York, NY, USA

Abstract

Compressed sensing magnetic resonance imaging (CS-MRI) is an active research topic in the field of inverse problems. Conventional CS-MRI algorithms usually exploit the sparse nature of MRI in an iterative manner. These optimization-based CS-MRI methods are often time-consuming at test time, and are based on fixed transform bases or shallow dictionaries, which limits modeling capacity. Recently, deep models have been introduced to the CS-MRI problem. One main challenge for CS-MRI methods based on deep learning is the trade-off between model performance and network size. We propose a recursive dilated network (RDN) for CS-MRI that achieves good performance while reducing the number of network parameters. We adopt dilated convolutions in each recursive block to aggregate multi-scale information within the MRI. We also adopt a modified shortcut strategy to help features flow into deeper layers. Experimental results show that the proposed RDN model achieves state-of-the-art performance in CS-MRI while using far fewer parameters than previously required.

Introduction

Magnetic resonance imaging (MRI) is a non-invasive medical imaging technique with the advantages of being high resolution and low radiation. The major limitation of MRI is its relatively slow imaging speed. This problem is critical because patients must remain still in the scanner for long periods of time to reduce motion artifacts. Different hardware and software strategies exist to accelerate MRI. Among them, compressed sensing (CS-MRI) is popular because no additional hardware is required, thus reducing extra costs.

According to compressed sensing theory (Candès, Romberg, and Tao 2006; Donoho 2006), for MRI signals that can be sparsely represented in an appropriate transform basis, far fewer k-space (Fourier transform) measurements are necessary for accurate MRI reconstruction than suggested by the traditional Nyquist sampling theorem. In previous work on CS-MRI, focus has been on proposing better objective functions and efficient optimization algorithms to exploit this fact. Our contribution will be to the former task, but we observe that processing of new MRI using deep learning will require no new optimizations, and so be very fast.

CS-MRI was pioneered by SparseMRI (Lustig, Donoho, and Pauly 2007). Since then, contributions have been made to efficient optimization of the SparseMRI objective, such as TVCMRI (Ma et al. 2008), RecPF (Yang, Zhang, and Yin 2010), and FCSA (Huang, Zhang, and Metaxas 2011). Other objectives choose different wavelet regularizations based on geometric information, such as PBDW/PBDWS (Qu et al. 2012; Ning et al. 2013) and GBRWT (Lai et al. 2016), or dictionary learning techniques such as DLMRI (Ravishankar and Bresler 2011) and BPTV (Huang et al. 2014).

Three potential issues arise with these methods:

1. Most methods are performed "in situ," meaning only information from the MRI of interest is used and any other potential MRI data ignored. While some methods are able to consider recurrent structure across MRI, such as NLR (Dong et al. 2014), PANO (Qu et al. 2014) and BM3D-MRI (Eksioglu 2016), they are usually implemented on a single MRI.
2. Data complexity is not sufficiently modeled. For example, with dictionary learning methods (Ravishankar and Bresler 2011; Huang et al. 2014) the patches of MRI are reconstructed using linear combinations of learned atoms and sparse coefficients. While structural sparsity has been introduced toward fixing this problem (Chen and Huang 2012), linear-based models are inherently less complex.
3. Optimizing the objective function for new MRI may time-consuming because they require iterative procedures. This may limit their use for real applications.

To address these limitations, deep learning models have been adopted for CS-MRI to exploit the large amount of synthetically available training data. Deep neural networks can also help model the intricate structures across various MRI. Finally, after training the network, the feed-forward testing of new MRI does not require new iterations and is extremely fast. The first work devoted to introducing deep learning to CS-MRI is (Wang et al. 2016). In their paper, a vanilla CNN model is directly used to learn the mapping from a zero-filled input MRI to the fully-sampled output MRI, which is also corrected with an additional data fidelity term. Subsequent papers have proposed a modified U-Net architecture (Lee, Yoo, and Ye 2017), and deep cascading of the CNN

(DC-CNN) (Schlemper et al. 2017), which is currently the state-of-the-art CS-MRI inversion technique.

The major problem in deep learning for CS-MRI is the trade-off between model performance and the number of network parameters. It has been shown that deeper network architectures can result in better network performance. But increasing model complexity requires more intensive computational and memory requirements. To achieve a balance between the number of network parameters and model performance, we propose a *recursive dilated network* (RDN) for CS-MRI. Using recursive computations reduces the number of network parameters, while dilated convolutions increase the receptive field to aggregate multi-scale information. A modified shortcut strategy is also employed to help propagate features to deeper layers directly. Our experiments show that the proposed RDN model improves upon the DC-CNN model while using far fewer parameters.

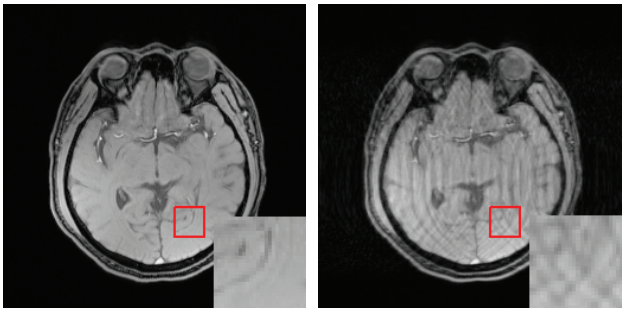
Related Work

In this section, we review the previous work and concepts from deep learning that are used in our proposed objective function for CS-MRI reconstruction.

Recursive learning Achieving good performance with a moderate number of network parameters is an important goal for designing deep neural networks. A design strategy called recursive learning aims at learning hierarchical feature representations by *reusing* the same convolutional filter weights, e.g., (Socher et al. 2012; Eigen et al. 2013; Liang and Hu 2015). Sharing filters can significantly reduce the storage requirements for very large deep models. For example, Deeply Recursive Convolution Networks (DRCN) (Kim, Kwon Lee, and Mu Lee 2016) use a very deep recursive layer (up to 16 recursions) to improve the model capacity without introducing more parameters. Similarly, Deep Recursive Residual Networks (DRRN) (Tai, Yang, and Liu 2017) use recursive learning to achieve state-of-the-art performance in single-image super resolution with fewer parameters.

Residual learning A major challenge for deep learning models is the optimization. To address the gradient vanish problem in back propagation, shortcuts have been proposed to stabilize the gradient flow in deep residual networks (ResNet) (He et al. 2016b; 2016a). In this model, every two convolutional layers include a shortcut as the basic residual block. ResNet has achieved very impressive performance on the ImageNet task.

Dilated convolutions Dilated convolutions were first proposed by (Yu and Koltun 2015) for the dense prediction task. Because an image contains repeated structural patterns at different scales, it is reasonable to model these features for visual tasks. Dilated convolutions, in which the same filter is increased in scale, can increase the receptive field to learn multi-scale features without sacrificing resolution or introducing additional parameters. (Yu, Koltun, and Funkhouser 2017) modified ResNet by proposing a residual dilated network, where dilated convolutions take the place of the under-sampling operation.



(a) Full-sampled image (b) Zero-filled reconstruction

Figure 1: A fully-sampled MRI and a zero-filled reconstruction using a 55% under-sampling Cartesian mask. We observe that artifacts in the zero-filled reconstruction are highly structured and significantly impair diagnostic information.

Method

Problem formulation

MRI machines directly measure the Fourier coefficients of an object (called k-space data), after which an inverse transform produces the image. We denote the vectorized fully-sampled k-space data as $y_{fs} \in \mathbb{C}^{N\times 1}$, and the resulting fully-sampled MRI as $x_{fs} \in \mathbb{R}^{N\times 1}$. The under-sampled (i.e., compressively sensed) k-space measurements can be represented by $y \in \mathbb{C}^{M\times 1}$, where $M \ll N$. The measured vector y is simply a sub-vector of y_{fs}. Replacing all unmeasured values with a zero followed by the inverse transform produces a "zero-filled MRI" $x_{zf} \in \mathbb{C}^{N\times 1}$. Zero-filled MRI are usually significantly degraded with artifacts and unusable for diagnostic applications, as shown in Figure 1.

The conventional CS-MRI inversion problem can be formulated as follows,

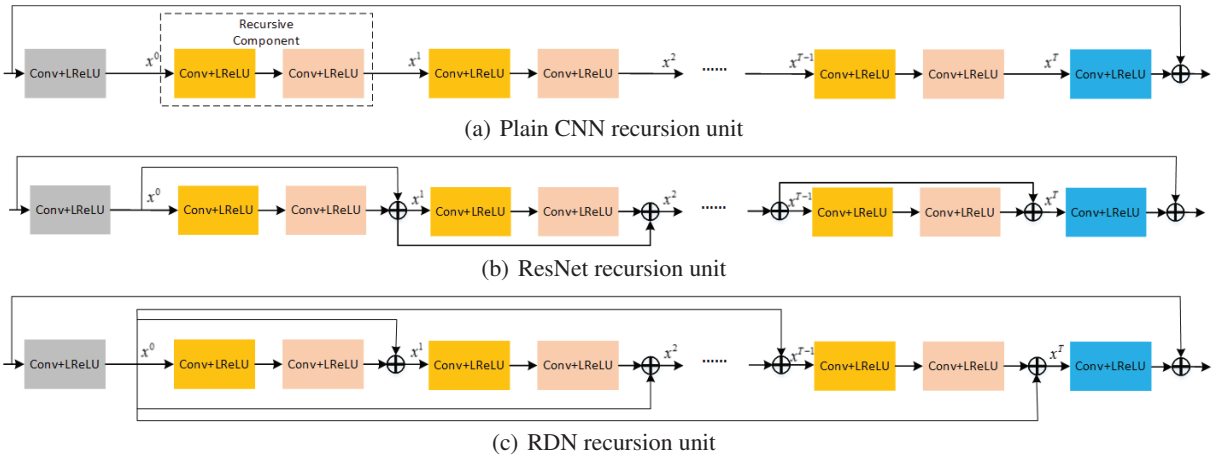
$$\hat{x} = \arg\min_{x} \|F_u x - y\|_2^2 + \sum_i \alpha_i \Psi_i(x), \qquad (1)$$

where $F_u \in C^{M\times N}$ is the under-sampled Fourier operator matrix corresponding to the measured k-space locations. Thus the zero-filled MRI x_{zf} can be calculated as $F_u^H y$. The first term is the data fidelity which enforces consistency between k-space values of the reconstructed image and the measured data. The second term Ψ_i regularizes the solution space, while α_i balances the importance of the two terms. Usually sparse regularization is imposed on Ψ_i via an l_1 norm or l_0 norm.

When the CS-MRI problem is addressed using deep learning models, a nonlinear mapping from the zero-filled MRI x_{zf} to the fully-sampled MRI x_{fs} is learned via a deep network model using many synthetically generated training examples. A new MRI can then be reconstructed through a fast feed-forward process on the input data. This requires no iterations, and so MRI reconstruction with deep learning is often much faster than conventional methods.

Building blocks of the proposed formulation

We propose a recursive dilated network (RDN) by cascading a series of basic blocks, which each consists of a recursive



(a) Plain CNN recursion unit

(b) ResNet recursion unit

(c) RDN recursion unit

Figure 2: We illustrate three different recursive methods. The first and last Conv+LReLU structure are designed to map the image into feature maps and aggregate feature maps into an image, respectively. The same color indicates shared filters.

dilated unit and a data fidelity unit. We next introduce how the recursive strategy and dilation are combined and give details about the recursive method and dilation convolutions. Then we briefly talk about the data fidelity unit and give the architecture of the proposed RDN model.

Recursive methods As we mentioned above, the trade-off between the number of network parameters and the model performance can be overcome using a recursive strategy, where the nonlinear mapping operator is shared within a network unit.

An intuitive idea is to use a recursive component T times, which consists of convolutional layers with activation function as shown in Figure 2(a). For ease of training, we adopt the global residual learning with a long global shortcut. Each recursive component consists of several convolutional layers using Leaky Rectified Linear Units (Conv+LReLU). The number of Conv+LReLU is not restricted; for illustration we use 2 recursive Conv+LReLUs per sharing component. In the figure we denote the sharing of the convolutional filters by using the same color. The first Conv+LReLU structure is used as a feature extraction to transform the image into multiple feature maps. The last Conv+LReLU is used as a reconstruction to aggregate the multiple feature maps into an image. We can notationally write the structure of the input and output relationship in the t^{th} and $(t+1)^{th}$ recursion $(1 \leq t < T)$ as

$$\text{CNN}: \quad \mathrm{x}^{\mathrm{t}} = \mathrm{f}\left(\mathrm{x}^{\mathrm{t}-1}\right), \quad \mathrm{x}^{\mathrm{t}+1} = \mathrm{f}\left(\mathrm{x}^{\mathrm{t}}\right). \tag{2}$$

Because the filters are reused recursively, each recursive component represents the *same* mapping function $f(\cdot)$. However as the recursions continue, the network depth increases, which introduces a severe gradient vanish problem that makes training difficult.

Naturally, a modified recursive strategy is to use the standard residual learning structure shown in Figure 2(b), where the input of each recursive component is also feeded to the output of the recursive component via a shortcut connection. Similar to the Figure 2(a), we also adopt 2 Conv+LReLU structures in each recursive component for illustration. Previous work has shown that residual learning with such a shortcut has superior performance compared with the vanilla CNN model because the mapping function is easier to learn. We similarly formulate the structure as

$$\text{ResNet}: \quad \mathrm{x}^{\mathrm{t}} = \mathrm{f}\left(\mathrm{x}^{\mathrm{t}-1}\right) + \mathrm{x}^{\mathrm{t}-1}, \quad \mathrm{x}^{\mathrm{t}+1} = \mathrm{f}\left(\mathrm{x}^{\mathrm{t}}\right) + \mathrm{x}^{\mathrm{t}}. \tag{3}$$

The function mapping $f(\cdot)$ is now only required to learn a mapping from an input to the residual signal. However, details contained in feature maps for low layers may be lost if they are passed through too many layers of the network. Therefore, our aim in the proposed CS-MRI model is to propagate the original information from lower levels to deeper levels directly.

An alternative approach is to send the feature maps from low layers to the deeper layers as shown in Figure 2(c). We observe that the output of the feature extraction Conv+LReLU structure x^0 is fed into not only the input of the adjacent recursive component, but also all subsequent outputs. This propagates information from the lower feature maps directly to deeper layers. Again, we can formulate the structure as

$$\text{RDN}: \quad \mathrm{x}^{\mathrm{t}} = \mathrm{f}\left(\mathrm{x}^{\mathrm{t}-1}\right) + \mathrm{x}^{1}, \quad \mathrm{x}^{\mathrm{t}+1} = \mathrm{f}\left(\mathrm{x}^{\mathrm{t}}\right) + \mathrm{x}^{1}. \tag{4}$$

We evaluated the benefit of transitioning from recursive Plain CNN (Equation 2, Figure 2(a)), recursive ResNet (Equation 3, Figure 2(b)) to our adopted recursive strategy (Equation 4, Figure 2(c)) using 10 components recursions. We show this performances in Figure 3 (The evaluation index PSNR and SSIM will be elaborated in experiment section). This experiment is conducted on 61 real-valued brain MR images using a 1D 30%-sampled Cartesian mask. As is evident, propagating all information forward directly from the input slightly outperforms a recursive version of ResNet. (We also note that for all test cases, ResNet outperformed the plain CNN.)

Dilated convolution In many deep neural network models, under-sampling operations such as pooling are used to

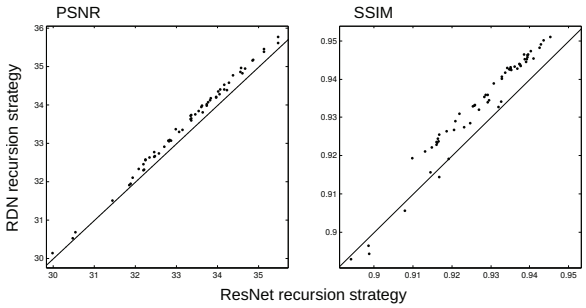


Figure 3: A comparison of recursive strategies using ResNet (Figure 2(b)) and the proposed RDN (Figure 2(c)). We do not use dilation in this experiment in order to directly compare the two shortcut approaches as visualized in Figure 2.

increase the receptive field, which increases the ability to use larger contextual information contained in an image. However, the pooling operation can create loss in resolution and degrade model performance for pixel-level problems. To avoid this loss, some models use convolutional kernels of varying size to alter the receptive field, which comes at the expense of a dramatic increase in parameters.

To achieve a balance between the receptive field size and the number of filters, dilated convolutions have been proposed. In Figure 4, we show how we use this technique in combination with recursive learning to form the recursive dilated unit. The kernels in the figure are of the same filter size (FS), but have different dilation factor (DF).

In low-level vision tasks like image restoration and reconstruction, contextual information at different scales is important for removing noise or artifacts. The same analysis applies to the CS-MRI problem, since under-sampling k-space brings significant artifacts as shown previously in Figure 1. In this circumstance, a larger receptive field can help distinguish between the true structural details and artifacts.

We also compared the proposed RDN framework in Equation 4 with and without using dilated filters. These results are shown in Figure 5. The results give support for using multi-scale filtering for the CS-MRI problem based on PSNR and the SSIM measure. For this comparison, we cascade 5 blocks, and each block includes recursive components used 3 times, feature extraction and reconstruction. The recursive components differ among the blocks. Each recursive component consists of two Conv+LReLU structures. For experiments with dilation, we use 1-dilation in first Conv+LReLU unit and 2-dilation in the second. For pure recursive strategy, we only use 1-dilation convolution.

Data fidelity unit Image details can be degraded as the network becomes deeper. For classification problems this is ideal, since the goal is to "boil down" the information to a linearly classifiable vector, but for MRI a high-quality output image is required. In addition to the modified shortcut method described previously, which transmits the input information across all layers, we also introduce a "data fidelity unit" to correct for the reconstruction error. From the sub-sampled measurement y, we have accurate values for the

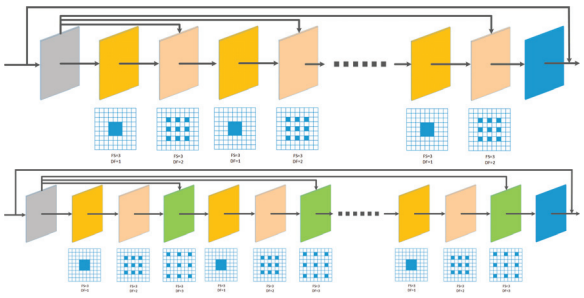


Figure 4: Top: The dilated convolutions corresponding to Figure 2(c). Bottom: A similar example when the dilation factor grows up to 3 with the number of Conv+LReLU structures in each recursive component.

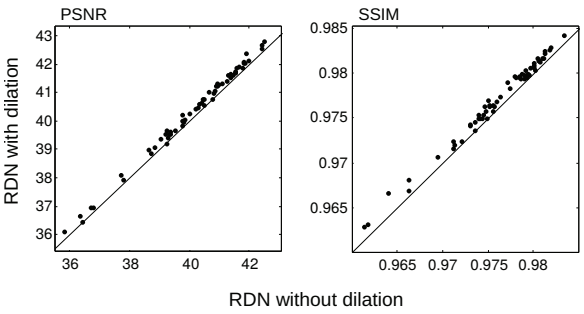


Figure 5: A comparison of the proposed RDN shortcut strategy with and without dilated filters. Multi-scale modeling with dilation gives an improvement on 61 test images.

sampled positions in k-space space. This is valuable for correcting any biases that accrue as the signal flows through the nonlinear components down the neural network.

Similar to Equation 1, we solve the following objective function with the data fidelity unit,

$$\hat{x} = \arg\min_{x} \frac{\lambda}{2} \left\| F_u x - y \right\|_2^2 + \left\| x - x_{in} \right\|_2^2, \qquad (5)$$

where x_{in} is the input to the data fidelity unit and λ is the regularization parameter. To enforce consistency between the reconstruction and the measurements y, we set λ a large value, e.g., 10^6. The second term can be viewed as the prior guess, where the input image x_{in} is reconstructed by the deep neural network.

In practice, this can be automatically solved via back-propagation using available software such as TensorFlow. However, we note that the above least squares problem does have a closed-form solution. We can simplify the problem by working in Fourier domain, after which the solution is

$$\hat{x} = F^H \frac{\lambda F F_u^H y + F x_{in}}{\lambda F F_u^H F_u F^H + I}, \qquad (6)$$

where the term $F F_u^H y$ is the Fourier transform of the zero-filled reconstruction, the term $F F_u^H F_u F^H$ is a diagonal matrix with ones at the sampled locations and zeros otherwise.

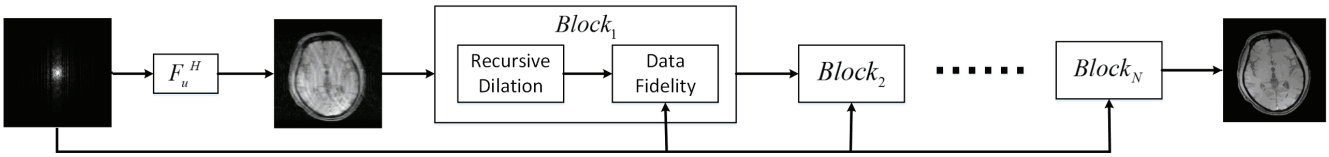


Figure 6: The flowchart of the proposed RDN network. Together the recursive dilation and data fidelity units comprise the basic blocks. The blocks are cascaded to form an end-to-end network. The input is the k-space measurements (left-most image) and the output is the reconstructed MRI (right-most image). The zero-filled reconstruction (second-left image) forms the basis for reconstruction, but accuracy of the reconstruction in k-space is enforced at each block by the data fidelity unit.

Calling the feed forward function for this unit $g\left(x_{in}; y; \lambda\right)$, the relevant gradient for model training is

$$\frac{\partial g}{\partial {x_{in}}^T} = \frac{I}{\lambda F F_u^H F_u F^H + I}. \tag{7}$$

We present these calculations for completeness, but again note that current software can automatically handle the inclusion of this unit during network training.

Network architecture

As discussed, the proposed RDN framework consists of repeated blocks. Each block includes two units, one called a recursive dilation unit and the other a data fidelity unit. We show this network architecture of the RDN model in Figure 6. In our experiments, we use the following naming convention: If we cascade N blocks, reuse the recursive component T times in each block where the dilation factor goes up to F, we refer to the model as RDN-NB-FD-TR.

We can train the RDN network in an end-to-end manner with training data consisting of multiple pairs of a k-space measurement vector y and its corresponding fully-sampled MRI x_{fs}. We first feed the zero-filled MRI x_{zf} constructed from y into the RDN network. We denote the function of the N^{th} block as B_N. Thus the output of the model can be written as

$$B\left(y\right) = B_N\left(B_{N-1}\left(\cdots\left(B_1\left(F_u^H y\right)\right)\right)\right). \tag{8}$$

The corresponding loss function is

$$L = \frac{1}{2P}\sum_{i=1}^{P}\left\|B\left(y_i\right) - x_{fs}^{(i)}\right\|_2^2, \tag{9}$$

where the sum is over a mini-batch with P data. This network can be efficiently optimized using, e.g., TensorFlow.

Experiments

We compare our algorithm with several state-of-the-art deep and non-deep techniques on several brain MRI.

Implementation details We train and test the algorithm using TensorFlow for the Python environment on a NVIDIA GeForce GTX 1080Ti with 11GB GPU memory. We use the Xavier method to initialize the network parameter and ADAM with momentum for parameter learning. We select the initial learning rate to be 0.001, the first-order momentum to be 0.9 and the second momentum to be 0.999. We set the weight decay regularization parameter to 0.0001 and the size of training batch to 8. 70000 stochastic iterations of training were required to train the RDN model. For all experiments we set the filter size to be 3×3 and each convolution layer to have 32 feature maps, except for the first and last layers which map from and to the image space, respectively.

Dataset Our training data consists of 2800 normalized real-valued brain MRI. The testing set consists of 61 brain MRI. We collected the images using a 3T MR scanner. All MRI are T1 weighted and of the same size (256×256). Unless otherwise indicated, we use a Cartesian mask with 30% sub-sampling rate.

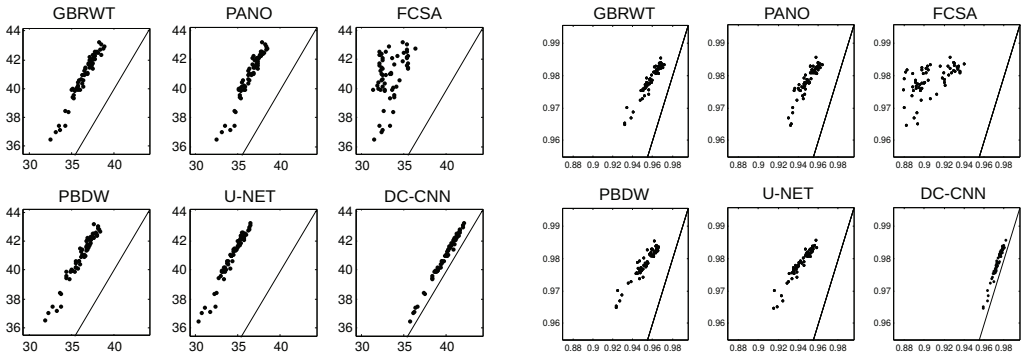
Quantitative evaluation

We compare the RDN model with non-deep models FCSA, PBDW, PANO, GBRWT, and two deep models U-Net (Lee, Yoo, and Ye 2017) and DC-CNN (Schlemper et al. 2017). For the non-deep methods, we use the software provided by the authors on their homepage. For the deep models, we re-implement according to the original paper specifications using TensorFlow. In the original paper of DC-CNN, the authors implement a deep network with 5 blocks with each block containing 5 convolution layers with ReLU activation. For fair comparison, we adopt the RDN-5B-3D-3R to keep the total number of network parameters the same, because the number of convolution layer is same in both methods and dilation brings no additional parameters. For U-Net, we follow the same network architecture as the original paper, which contains more parameters than DC-CNN.

For image reconstruction quality assessment we use two popular measures: peak signal-to-noise ratio (PSNR) and the structural similarity index measure (SSIM) (Wang et al. 2004). We compare testing results with other state-of-the-art algorithms in Figure 7. In each plot, the x-axis corresponds to the algorithm listed in the title and the y-axis is the proposed RDN framework. We see that the points lie above the diagonal line for each comparison, indicating that RDN outperforms each algorithm. We also observe that the current state-of-art DC-CNN model clearly performs the second best, while the other deep model U-Net does not provide an obvious improvement over other non-deep methods.

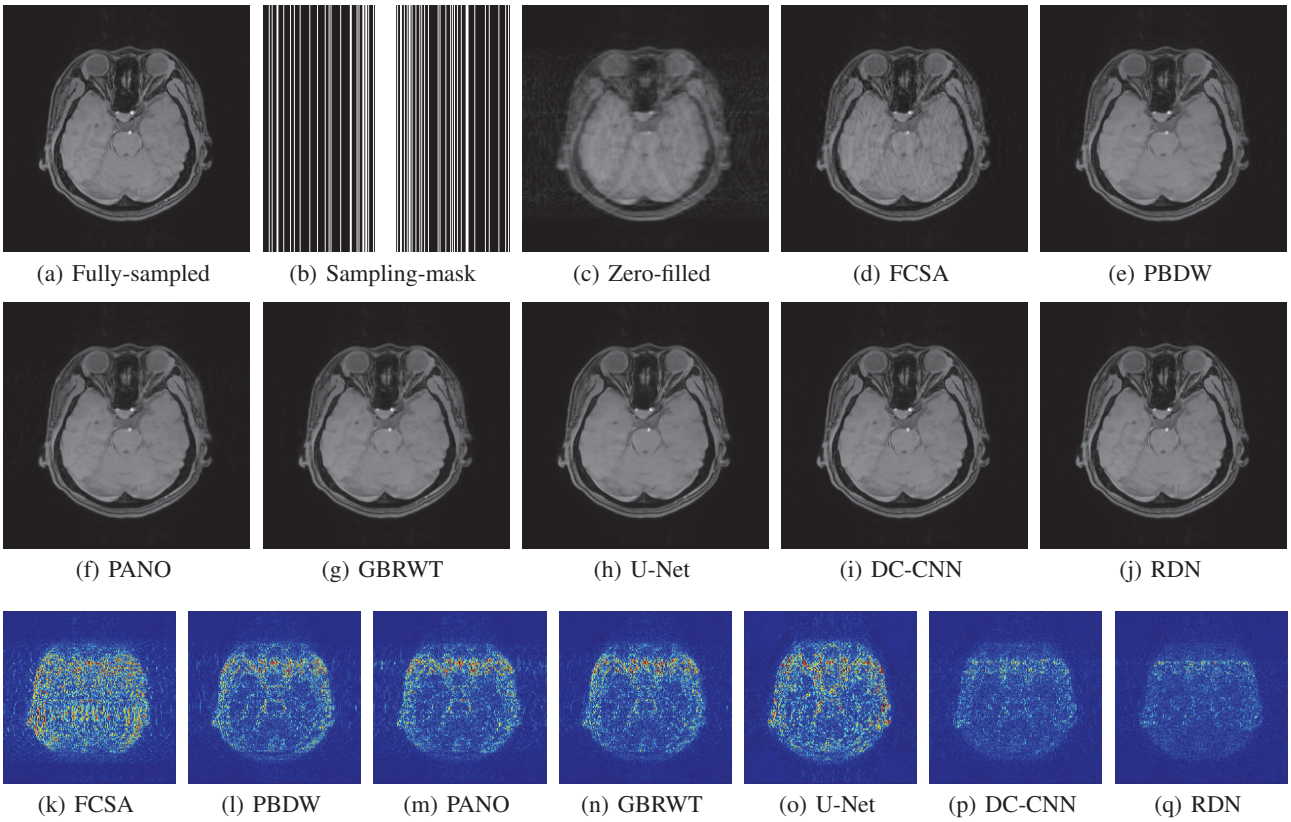
Qualitative evaluation

We show the results on a brain MRI in Figure 8. As is clear from the error images, RDN has state-of-the-art performance. The colorbar ranges form [0 0.08]. To check per-



(a) PSNR: RDN vs $-$ (b) SSIM: RDN vs $-$

Figure 7: A comparison of the proposed recursive dilation (RDN) network (y-axis) with several CS-MRI inversion methods (x-axis) on 61 test images. A point above the diagonal line indicates that RDN outperforms the indicated algorithm for that test image according to PSNR (left) or SSIM (right).



(a) Fully-sampled (b) Sampling-mask (c) Zero-filled (d) FCSA (e) PBDW

(f) PANO (g) GBRWT (h) U-Net (i) DC-CNN (j) RDN

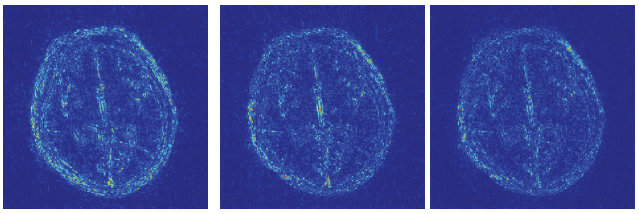
(k) FCSA (l) PBDW (m) PANO (n) GBRWT (o) U-Net (p) DC-CNN (q) RDN

Figure 8: Results on a brain MRI using 30% Cartesian under-sampling.

formance for other sampling masks, we experiment with a 10% sub-sampling radial mask and show qualitative results on another MRI data in Figure 9. Results for radial sampling were consistent with those using Cartesian sampling.

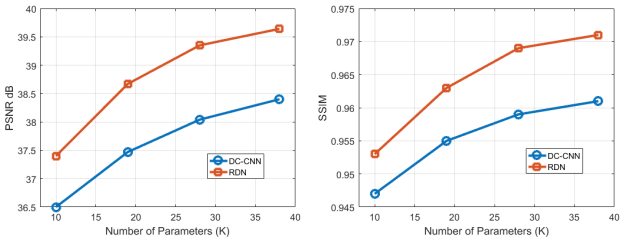
Discussion on the number of network parameter

The proposed RDN model achieves good performance while keeping the overall network parameter number reasonably low. We show performance of RDN and DC-CNN as a function of parameter number in Figure 10. For both, we use 3 blocks and only vary the number of parameters in each block by changing the number of Conv+LReLU structure in each recursive component, i.e., "FD" in our notation. In Figure 11 we show a reconstructed test MRI using the state-of-the-art deep model DC-CNN having 38K parameters and our RDN having 28K parameters (i.e., 10K fewer parameters).



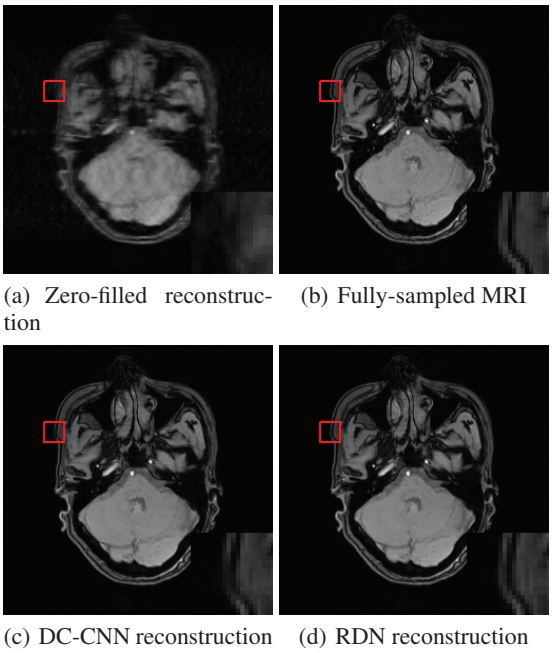
(a) GBRWT (b) DC-CNN (c) RDN

Figure 9: Reconstruction errors using a 10% sub-sampled 2D radial mask.



(a) PSNR (b) SSIM

Figure 10: The comparison of model performance of DC-CNN and RDN conditioned on the number of parameters.



(a) Zero-filled reconstruction (b) Fully-sampled MRI

(c) DC-CNN reconstruction (d) RDN reconstruction

Figure 11: The comparison for zero-filled reconstruction, full-sampled image, DC-CNN reconstruction and RDN reconstruction The experiment is conducted on a real-valued brain MRI using 30% Cartesian under-sampling.

While both significantly improve the zero-filled result, our RDN model captures additional details while using fewer parameters. From Figure 10, we see that quantitative performance becomes comparable when RDN reduces further to about 18K parameters, i.e., about half of DC-CNN. If the depth of the proposed model is the same as vanilla CNN, their computational complexity is almost the same.

Discussion on the number of recursions

In Table 1 we show how a 5-block RDN model performs as a function of the number of recursions within each recursive dilated unit. Here we give the comparison of RDN-5B-3D-1R, RDN-5B-3D-2R, RDN-5B-3D-3R, RDN-5B-3D-4R and RDN-5B-3D-5R. The performance of the RDN model improves as the number of recursion increases. However, because parameters are reused in each recursion, their number is fixed as a function of recursion number.

Table 1: Performance as a function of number of recursions.

Recursion	1	2	3	4	5
PSNR	39.91	40.37	40.87	40.97	41.09
SSIM	0.975	0.977	0.979	0.980	0.981

Conclusion

We have proposed a recursive dilated network for CS-MRI. The network is formed by cascading a series of basic blocks consisting of a recursive dilation unit and a data fidelity unit. The recursive reuse of filters can reduce the number of parameters in the network while maintaining good model performance. We propagate the measured k-space values to all levels in the network to reduce information loss. We also use dilated convolutions to increase the receptive field without introducing more parameters. The modified shortcut strategy helps the information flow from lower layers to deeper layers. Our experiments show that the proposed RDN model can achieve state-of-the-art performance for CS-MRI while keeping number of free parameter relatively low.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China grants 61571382, 81671766, 61571005, 81671674, U1605252, 61671309 and 81301278, Guangdong Natural Science Foundation grant 2015A030313007, Fundamental Research Funds for the Central Universities grants 20720160075 and 20720150169, the CCF-Tencent research fund, and the Science and Technology funds from the Fujian Provincial Administration of Surveying, Mapping and Geoinformation.

References

Candès, E. J.; Romberg, J.; and Tao, T. 2006. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory* 52(2):489–509.

Chen, C., and Huang, J. 2012. Compressive sensing MRI

with wavelet tree sparsity. In *Advances in neural information processing systems*, 1115–1123.

Dong, W.; Shi, G.; Li, X.; Ma, Y.; and Huang, F. 2014. Compressive sensing via nonlocal low-rank regularization. *IEEE Transactions on Image Processing* 23(8):3618–3632.

Donoho, D. L. 2006. Compressed sensing. *IEEE Transactions on information theory* 52(4):1289–1306.

Eigen, D.; Rolfe, J.; Fergus, R.; and LeCun, Y. 2013. Understanding deep architectures using a recursive convolutional network. *arXiv preprint arXiv:1312.1847*.

Eksioglu, E. M. 2016. Decoupled algorithm for MRI reconstruction using nonlocal block matching model: BM3D-MRI. *Journal of Mathematical Imaging and Vision* 56(3):430–440.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016a. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016b. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, 630–645. Springer.

Huang, Y.; Paisley, J.; Lin, Q.; Ding, X.; Fu, X.; and Zhang, X.-P. 2014. Bayesian nonparametric dictionary learning for compressed sensing MRI. *IEEE Transactions on Image Processing* 23(12):5007–5019.

Huang, J.; Zhang, S.; and Metaxas, D. 2011. Efficient MR image reconstruction for compressed MR imaging. *Medical Image Analysis* 15(5):670–679.

Kim, J.; Kwon Lee, J.; and Mu Lee, K. 2016. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1637–1645.

Lai, Z.; Qu, X.; Liu, Y.; Guo, D.; Ye, J.; Zhan, Z.; and Chen, Z. 2016. Image reconstruction of compressed sensing MRI using graph-based redundant wavelet transform. *Medical image analysis* 27:93–104.

Lee, D.; Yoo, J.; and Ye, J. C. 2017. Deep residual learning for compressed sensing MRI. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, 15–18. IEEE.

Liang, M., and Hu, X. 2015. Recurrent convolutional neural network for object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3367–3375.

Lustig, M.; Donoho, D.; and Pauly, J. M. 2007. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic resonance in medicine* 58(6):1182–1195.

Ma, S.; Yin, W.; Zhang, Y.; and Chakraborty, A. 2008. An efficient algorithm for compressed MR imaging using total variation and wavelets. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 1–8. IEEE.

Ning, B.; Qu, X.; Guo, D.; Hu, C.; and Chen, Z. 2013. Magnetic resonance image reconstruction using trained geometric directions in 2D redundant wavelets domain and non-convex optimization. *Magnetic resonance imaging* 31(9):1611–1622.

Qu, X.; Guo, D.; Ning, B.; Hou, Y.; Lin, Y.; Cai, S.; and Chen, Z. 2012. Undersampled MRI reconstruction with patch-based directional wavelets. *Magnetic resonance imaging* 30(7):964–977.

Qu, X.; Hou, Y.; Lam, F.; Guo, D.; Zhong, J.; and Chen, Z. 2014. Magnetic resonance image reconstruction from undersampled measurements using a patch-based nonlocal operator. *Medical image analysis* 18(6):843–856.

Ravishankar, S., and Bresler, Y. 2011. MR image reconstruction from highly undersampled k-space data by dictionary learning. *IEEE transactions on medical imaging* 30(5):1028–1041.

Schlemper, J.; Caballero, J.; Hajnal, J. V.; Price, A.; and Rueckert, D. 2017. A deep cascade of convolutional neural networks for MR image reconstruction. In *International Conference on Information Processing in Medical Imaging*, 647–658. Springer.

Socher, R.; Huval, B.; Bath, B.; Manning, C. D.; and Ng, A. Y. 2012. Convolutional-recursive deep learning for 3D object classification. In *Advances in Neural Information Processing Systems*, 656–664.

Tai, Y.; Yang, J.; and Liu, X. 2017. Image super-resolution via deep recursive residual network. In *Proceeding of IEEE Computer Vision and Pattern Recognition*.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13(4):600–612.

Wang, S.; Su, Z.; Ying, L.; Peng, X.; Zhu, S.; Liang, F.; Feng, D.; and Liang, D. 2016. Accelerating magnetic resonance imaging via deep learning. In *Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on*, 514–517. IEEE.

Yang, J.; Zhang, Y.; and Yin, W. 2010. A fast alternating direction method for TVL1-L2 signal reconstruction from partial fourier data. *IEEE Journal of Selected Topics in Signal Processing* 4(2):288–297.

Yu, F., and Koltun, V. 2015. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Yu, F.; Koltun, V.; and Funkhouser, T. 2017. Dilated residual networks. In *Proceeding of IEEE Computer Vision and Pattern Recognition*.