Supplementary Material

1. Network architecture

The proposed network architecture is shown in Supplementary Material Figure S1 with its specifications listed in Supplementary Material Table S1. The first step in processing is a data normalization step (referred to as pre-processing in Supplementary Material Figure S1) applied to the subspace filtered input data. Let $\mu_{i,n}$ and $\delta_{i,n}$, $1 \leq n \leq N$, denote the empirical mean and standard deviation of the n^{th} contrast-weighted magnitude input image across the training data, respectively. Similarly, let $\mu_{t,n}$ and $\sigma_{t,n}$, $1 \leq n \leq N$, denote the empirical mean and standard deviation for the target training data. Consider each input patch $\mathbf{p}_i \in \mathbb{R}^{P \times P \times N}$ to the network as a multi-dimensional array with $P \times P$ spatial pixels and N contrasts. The input patch $\mathbf{p}_i$ is normalized using quantities $\mu_{i,n}$ and $\sigma_{i,n}$ as follows:

$$\bar{\mathbf{p}}_i(k,l,n) = \frac{\mathbf{p}_i(k,l,n) - \mu_{i,n}}{\sigma_{i,n}},\ 1 \leq k,l \leq P,\ 1 \leq n \leq N. \tag{1}$$

It is important to emphasize that, although data normalization is commonly carried out in neural network training, its importance is even greater in this application. The dynamic range of the input signal can vary drastically across the multi-contrast images. Without data normalization, the network may be biased towards the contrasts with large signal amplitudes, such as the early TE images in T_2 mapping, and neglect learning the contrasts with smaller signal amplitudes, such as the late TE images in T_2 mapping. This pre-processing step largely eliminates the scaling discrepancies in multi-contrast images within the network.

Symmetrically, the proposed network architecture includes a post-processing step at the end of the network as shown in Supplementary Material Figure S1. Instead of normalizing the target patches used in the training, the network processes the patches $\bar{\mathbf{p}}_o$ that are input to the post-processing step to yield the output patches $\mathbf{p}_o$ using

$$\mathbf{p}_o(k,l,n) = \bar{\mathbf{p}}_o(k,l,n)\sigma_{t,n} + \mu_{t,n},\ 1 \leq k,l \leq P,\ 1 \leq n \leq N. \tag{2}$$

where $\mu_{t,n}$ and $\sigma_{t,n}$ denote the empirical mean and standard deviation values for the target images as described earlier. The post-processing step together with the pre-processing step are referred to as contrast normalization.

The proposed network architecture was constructed in a multi-scale fashion as displayed in Supplementary Material Figure S1(a). The parameters associated with each layer of the network are provided in Supplementary Material Table S1. The pre-processing layer performs the contrast normalization as described earlier. The resulting patches are then propagated at two spatial scales so that the receptive field size of the network is enlarged. The Conv_1 layer converts the contrasts of inputs to a fixed number of features. It was shown that residual networks are easier to optimize and provide improved accuracy at increased network depth [1] compared to convolutional neural networks. Therefore, at each spatial scale, a stack of residual blocks (ResBlocks) was adopted. The ResBlock consists of a rectified linear unit (ReLU) layer that lays in between two convolutional layers as illustrated in Supplementary Material Figure S1(b). Note that no batch normalization layer was deployed since our network pre-processing procedure has partially accounted for improper dynamical range of input and target data. Instead, a constant scaling factor of 0.1 was used inside each residual block for training stability [2]. The Conv_2 layer together with stacked residual blocks are intended to learn features at the original resolution. The additional global skip-connection results in substantial gains in performance without increasing inference time as reported in [3]. The downsampling layer, Conv_3, uses a stride-2 convolution rather than stride-2 pooling since a trainable convolutional layer provides more representability and flexibility in comparison to a non-parameterized max or average pooling. The Conv_4 layer combined with ResBlocks via a skip-connection is used to learn representations in lower spatial resolution. The up-sampling layer, Conv_5, uses computationally efficient sub-pixel convolution [4]. High resolution features from the first group of ResBlocks and low-resolution features from the up-sampling layer are combined by weighted summation. This single weighting factor in this summation is learnable and

auto-tuned by the network at each iteration to balance the contributions of high and low resolution features. A final convolution layer, Conv_6, reconstructs multi-contrast outputs from the fixed number of features. These multi-contrast outputs are post-processed to recover their original dynamic ranges using contrast normalization.

2. Network component validation

A series of experiments were performed to validate the components of MS-ResNet, such as the preprocessing step or architectural hyper-parameters. Experiment using the T_2 brain data is first conducted to justify the major components of our MS-ResNet from three aspects: multi-scale structure, residual blocks, and its depth. To evaluate the multi-scale structure, we designed a single-scale network where the downsampling (Conv_3), upsampling (Conv_5), and FeatAdd layers were removed. To evaluate the contribution of the residual blocks, we designed a network where the residual block (consisting of the "Conv-ReLU-Conv-Scale-SkipConnection" steps as shown in Supplementary Material Figure S1(b)) was replaced by its CNN counterpart, "Conv-ReLU-Conv", with the same number of parameters. Three different depths, 16, 32, and 48, were used to evaluate the contribution of the depth of the network. Second, the efficacy of proposed subspace filtering was evaluated in the T_1 brain datasets. Third, the proposed contrast normalization was compared to batch normalization on the T_2 brain data. Lastly, the size of required data for training MS-ResNet was also investigated using T_1 brain data. Using the MS-ResNet architecture tuned with above experiments, we gradually increased the number of training subjects by randomly including an additional subject from the remaining training data pool.

Supplementary Material Figure S2 illustrates the validation Normalized Mean Square Error (NMSE) evolution of networks with (a) CNN and (b) ResNet structures on the T_2 brain data. The networks with ResNet structures outperform the ones with CNN structures by a large margin. Similarly, multi-scale networks outperform single-scale networks. Overall, MS-ResNet48 and MS-ResNet32 (used in the paper) converge to roughly the same NMSE, but MS-ResNet32 has lower memory requirements and is therefore preferable. Supplementary Material Figure S3 shows the training and validation NMSE with and without subspace filtering over the training iterations for the T_1 brain dataset. Including subspace filtering as a pre-processing step reduces NMSE during training and validation. Supplementary Material Figure S4 shows the TI images and T_1 maps obtained with and without the subspace filtering step using NUFFT and MS-ResNet, together with the reference NLR3D results. The subspace filtering step effectively reduce blurring in the resulting contrast-weighted images and parameter maps. Representative slices of T_2 maps obtained from these three networks together with the T_2 map from the NLR3D method were shown in Supplemental Information Figures S5. We find that contrast normalization greatly accelerates network learning since it normalizes the dramatic dynamic range variations between multi-contrast images and equalizes the learning weight on each contrast-weighted image. Residual artifacts remain in the T_2 maps of MS-ResNets with no normalization or with batch normalization, while these artifacts are significantly reduced with contrast normalization. Supplementary Material Figure S6 shows that MS-ResNet validation NMSE gradually improves as the number of training subjects increase from 1 to 4, but saturates at the convergent NMSE when more than 4 training subjects are used. These experiments demonstrate that it is sufficient to train MS-ResNet using a limited amount of data.

3. Statistical Analysis

In order to compare the T_2 estimation performances of MS-ResNet-FR and MS-ResNet-AR, statistical analysis was performed. $P < 0.05$ was considered statistically significant in all statistical tests. The full width at half maximum (FWHM) and the area under the curve within the [-5, 5] ms ($\text{AUC}_{[-5,5]}$) metrics were computed on a per slice basis over the entire T_2 knee testing cohort, resulting in 60 samples for each metric at each acceleration ratio. These metrics were first evaluated

using the Shapiro–Wilk [5] test for normality. The null hypotheses corresponding to each acceleration ratio were rejected due to p-values below 0.05. Thus, we can state that the data samples do not come from a normal distribution. Therefore, non-parametric Kruskal-Wallis tests [6] were conducted to test the null hypotheses that the mean ranks of these metrics for MS-ResNet-FR and MS-ResNet-AR are the same. Support Information Table S2 reports the Chi-square and p-values of the Kruskal-Wallis tests for different acceleration ratios. In all tests, the null hypothesis could not be rejected, suggesting that there is no evidence that the mean ranks of the metrics FWHM and $AUC_{[-5,5]}$ are different for MS-ResNet-FR and MS-ResNet-AR.

References

[1] He, K., Zhang, X., Ren, S., Sun, J.. Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE. ISBN 9781467388511; 2016, p. 770–778. doi:10.1109/CVPR.2016.90.

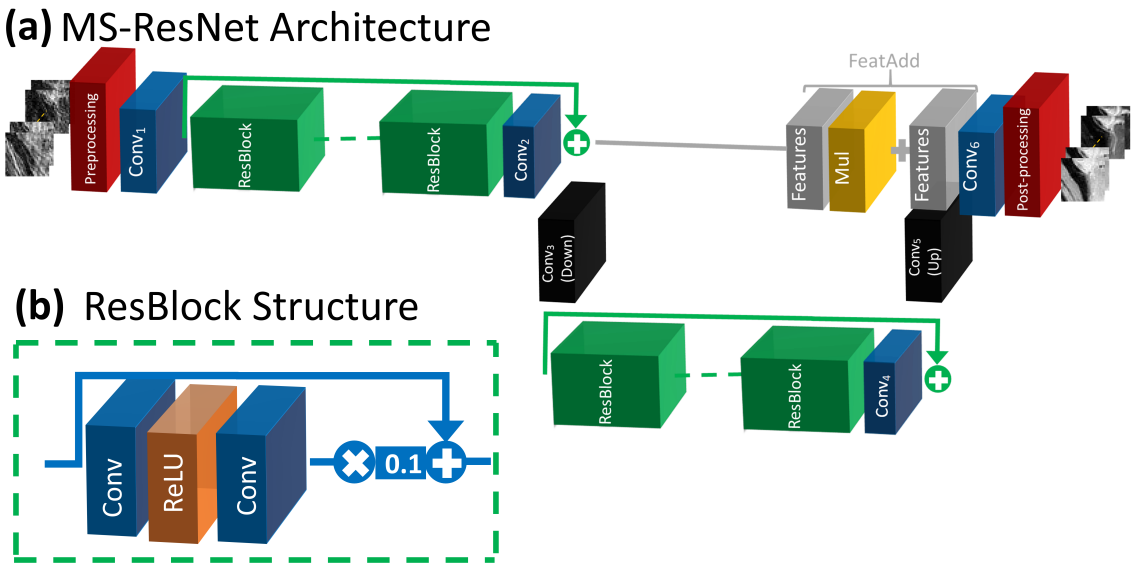
[2] Szegedy, C., Ioffe, S., Vanhoucke, V.. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. CoRR abs/1602.07261 (2016). 2016.

[3] Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI: IEEE. ISBN 978-1-5386-0457-1; 2017, p. 105–114. doi:10.1109/CVPR.2017.19.

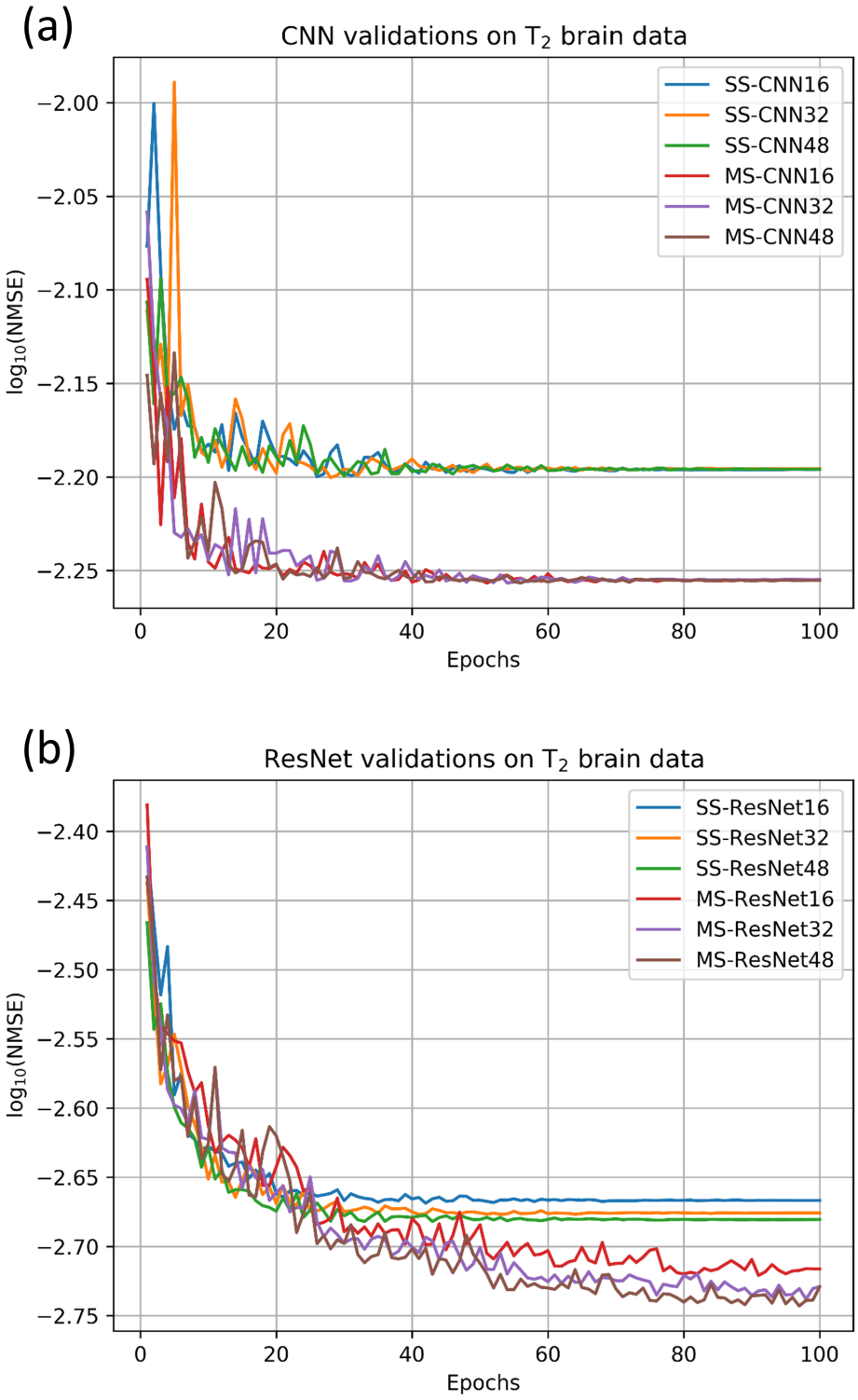
[4] Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016, p. 1874–1883. doi:10.1109/cvpr.2016.207.

[5] SHAPIRO, S.S., WILK, M.B.. An analysis of variance test for normality (complete samples)†. Biometrika 1965;52(3-4):591–611. doi:10.1093/biomet/52.3-4.591.

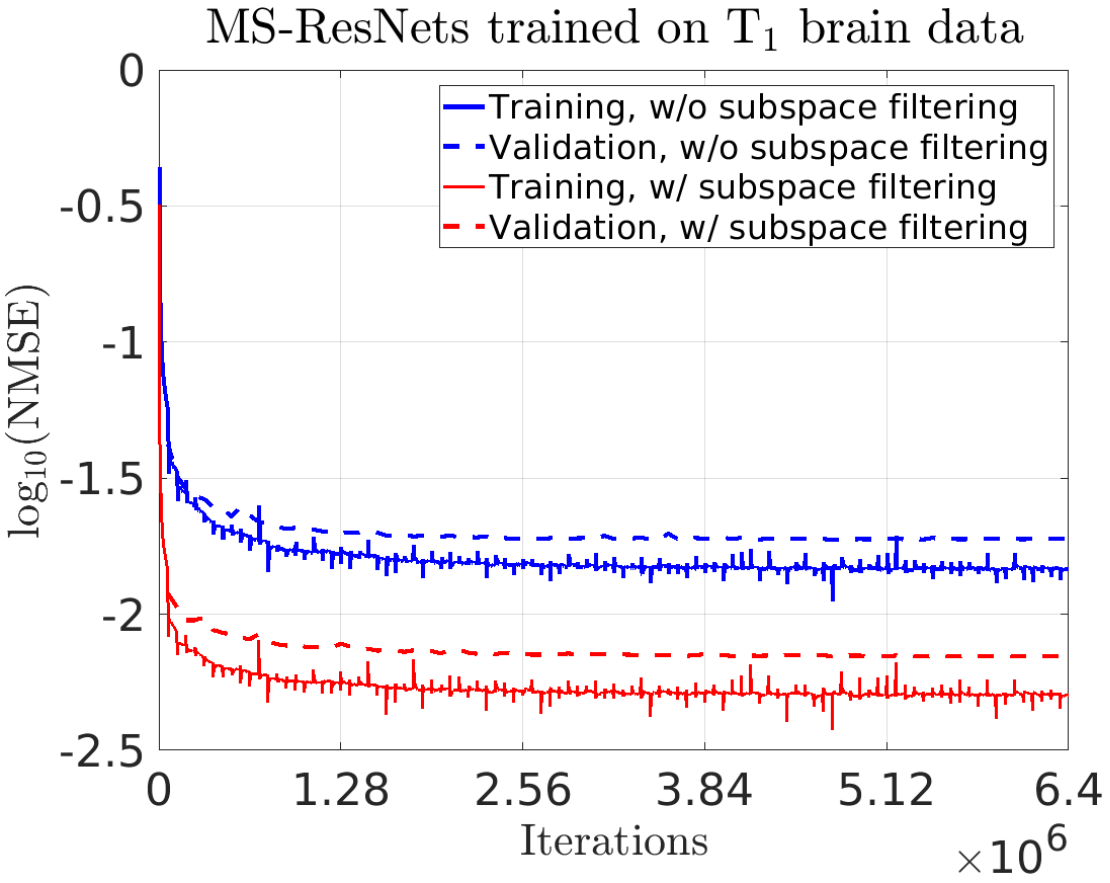
[6] Kruskal, W.H., Wallis, W.A.. Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association 1952;47(260):583–621. doi:10.1080/01621459.1952.10483441.



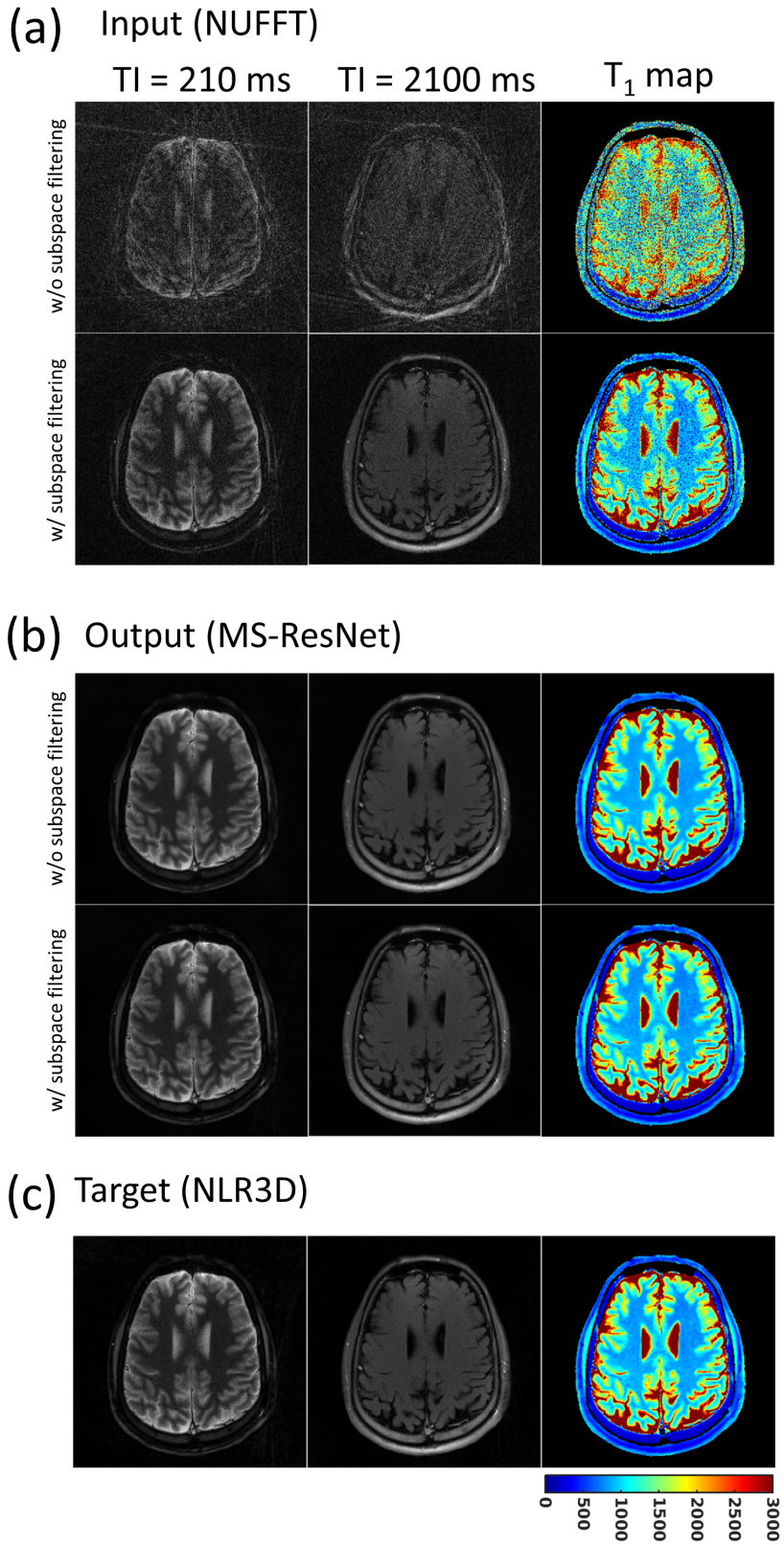
Supplementary Material Figure S1: Network architecture of the proposed multi-scale ResNet. (a) The input patches are pre-processed before they are propagated through the network at multiple spatial scales. At each scale, stacked residual blocks are used to learn features and express the patches at that resolution. Low-resolution features and high-resolution features are fused using weighted summation which has a single learnable weight. The output patches are obtained after post-processing the network output. The specifications of the network are provided in Supplementary Material Table S1. (b) The structure of a residual block is shown.



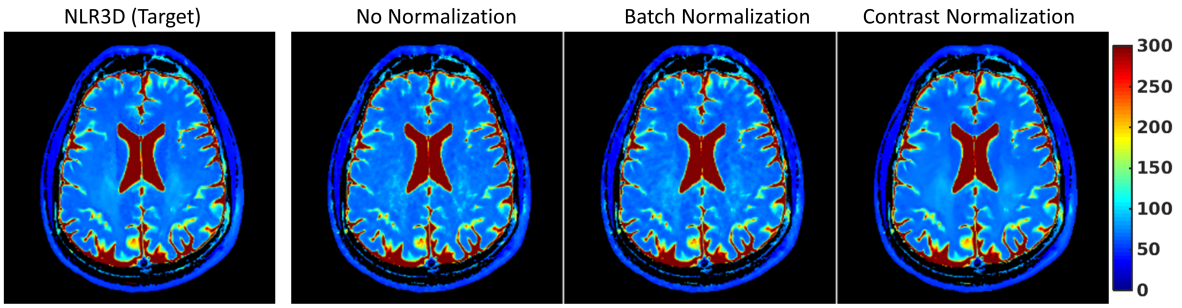
Supplementary Material Figure S2: Validation NMSE evolution of networks with (a) CNN and (b) ResNet structures on T_2 brain data. The networks starting with "MS" and "SS" represent multi-scale and single-scale networks, respectively. Similarly, "ResNet" and "CNN" in the network name corresponds to networks with residual and CNN blocks, respectively. Finally, the numbers included as suffixes to the network name correspond to the depth of the network.



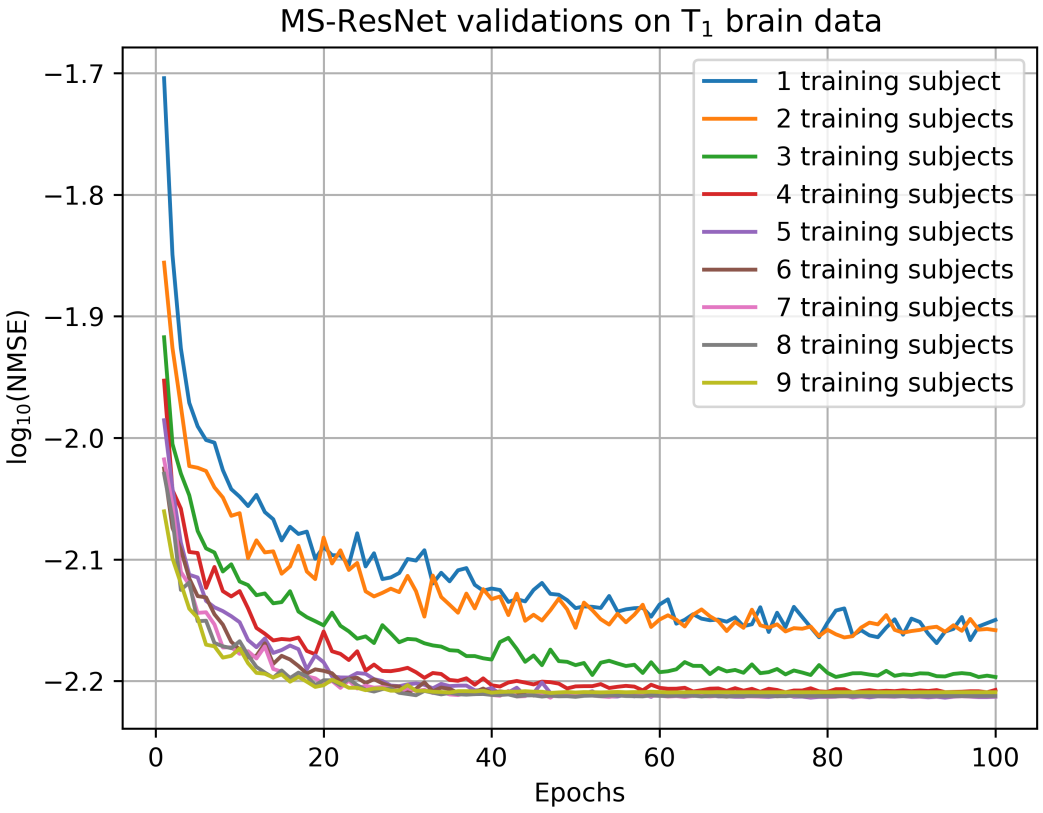
Supplementary Material Figure S3: Training and validation NMSEs of MS-ResNet trained on T_1 brain data with and without subspace filtering over training iterations.



Supplementary Material Figure S4: TI images and T_1 maps of the brain with and without subspace filtering using (a) NUFFT and (b) MS-ResNet are shown. The reference (c) NLR3D TI images and T_1 map are also provided.



Supplementary Material Figure S5: T_2 maps of the brain reconstructed by NLR3D, MS-ResNet with no normalization, MS-ResNet with batch normalization and MS-ResNet with proposed contrast normalization are shown.



Supplementary Material Figure S6: Validation NMSE evolution of MS-ResNet with varying number of training subjects on T_1 brain data.

Layer	**Activation**
Input	$32 \times 32 \times N$
Subspace Filtering	$32 \times 32 \times N$
Contrast Normalization	$32 \times 32 \times N$
Conv_1, $3 \times 3 \times 64$, stride 1, pad 1	$32 \times 32 \times 64$
16 ResBlocks, $3 \times 3 \times 64$	$32 \times 32 \times 64$
Conv_2, $3 \times 3 \times 64$, stride 1, pad 1	$32 \times 32 \times 64$
Conv_3 (Down), $6 \times 6 \times 64$, stride 2, pad 2	$16 \times 16 \times 64$
16 ResBlocks, $3 \times 3 \times 64$	$16 \times 16 \times 64$
Conv_4, $3 \times 3 \times 64$, stride 1, pad 1	$16 \times 16 \times 64$
Conv_5 (Up), $6 \times 6 \times 64$, stride 1/2, pad 2	$32 \times 32 \times 64$
[a] FeatAdd	$32 \times 32 \times 64$
Conv_6, $3 \times 3 \times N$, stride 1, pad 1	$32 \times 32 \times N$
Contrast Normalization	$32 \times 32 \times N$

[a] $w \times \text{act}(\text{Conv}_2) + \text{act}(\text{Conv}_5)$. w is a learnable weighting factor and $act(l)$ represents the activations of layer l.

Supplementary Material Table S1: Multi-scale Resnet Architecture Specification.

	$R = 8$	$R = 12$	$R = 16$
	[a] $AUC_{[-5,5]}$		
Chi-square	2.82	1.12	0.03
p-value	0.093	0.289	0.863
	FWHM		
Chi-square	1.99	0.52	0.04
p-value	0.158	0.472	0.850

[a] $AUC_{[-5,5]}$ denotes the normalized area under the curve within the interval $[-5, 5]$ ms.

Supplementary Material Table S2: Results of Kruskal-Wallis tests comparing $\text{AUC}_{[-5,5]}$ and FWHM metrics on the T_2 knee test cohort. Chi-square and p-values are provided for each metric and acceleration rate. Significance level α was set to 0.05.