Multiscale brain MRI super-resolution using deep 3D convolutional networks

Chi-Hieu Pham, Carlos Tor-Díez, Hélène Meunier, Nathalie Bednarek, Ronan Fablet, Nicolas Passat, François Rousseau

Multi-scale brain MRI super-resolution using deep 3D convolutional networks

Chi-Hieu Pham[a,*], Ronan Fablet[b], François Rousseau[a]

[a] *IMT Atlantique, LaTIM U1101 INSERM, UBL, Brest, France*
[b] *IMT Atlantique, LabSTICC UMR CNRS 6285, UBL, Brest France*

Abstract

The purpose of super-resolution (SR) approaches is to overcome the hardware limitations and the clinical requirements of imaging procedures by reconstructing high-resolution images from low-resolution acquisitions using post-processing methods. SR techniques could strong impacts on structural MRI when focusing on cortical surface or fine-scale structure analysis for instance. In this paper, we study deep three-dimensional (3D) convolutional neural networks (CNNs) for SR of brain MRI data. First, our work delves the relevance of several factors in the performance of the purely CNN-based technique for monomodal SR: optimization method, weight initialization, network depth, residual learning, filter size in convolution layers, number of the filters, training patch size and number of training subjects. Second, our study also highlights that one single network can efficiently handle multiple arbitrary scaling factors based on a multi-scale training approach. Third, we further extend our SR networks to multimodal SR using intermodality priors. Lastly, comparison to state-of-the-art methods in terms of transfer learning demonstrates the potential of CNNs for enhancing medical imaging.

Keywords: super-resolution, 3D convolutional neural network, brain MRI

1. Introduction

Magnetic Resonance Imaging (MRI) is a powerful imaging modality for in vivo brain visualization with a typical image resolution of $1mm$. Acquisition time of MRI data and signal-to-noise ratio are two parameters that drive the choice of an appropriate image resolution for a given study. The accuracy of further analysis such as brain morphometry can be highly dependent on image resolution. Super-Resolution (SR) aims to enhance the resolution of an imaging system using single or multiple data acquisitions [1]. The use of SR

*Corresponding author
Email addresses: ch.pham@imt-atlantique.fr (Chi-Hieu Pham), ronan.fablet@imt-atlantique.fr (Ronan Fablet), francois.rousseau@imt-atlantique.fr (François Rousseau)

techniques has been studied in many works in the context of brain MRI analysis: structural MRI [2, 3, 4, 5, 6], diffusion MRI [7, 8, 9, 10], spectroscopy MRI [11], quantitative T_1 mapping [12, 13], fusion of orthogonal scans of moving subjects [14, 15, 16, 17], *etc.* The development of efficient and accurate SR techniques for 3D MRI data could be a major step forward for brain studies.

Most SR methods rely on the minimization of a cost function consisting of a fidelity term related to an image acquisition model and a regularization term that constrains the space of solutions. The observation model is usually a linear model including blurring, motion, the effect of the point spread function and downsampling. The regularizer, which guides the optimization process while avoiding unwanted image solutions, can be defined using pixel-based l_2-norm term [14], total variation [6], local patch-based similarities [2, 4, 3], sparse coding [5], low rank property [6], *etc.*

However, the choice of the regularization term remains difficult as it modifies implicitly the space of acceptable solutions without any guaranty on the reconstruction of realistic high-resolution images. Conversely, in a supervised context (in which one can exploit a learning database with low-resolution (LR) and high-resolution (HR) images), the SR of MRI data can be fully driven by examples. The key challenge of supervised techniques is to accurately estimate the mapping operator from the LR image space to the HR one. Recently, significant advances have been reported in SR for computer vision using convolutional neural networks (CNN). This trend follows the tremendous outcome of CNN-based schemes for a wide range of computer vision applications, including for instance image classification [18, 19, 20], medical segmentation [21] or medical image analysis [22].

CNN architectures have become the state-of-the-art for image SR. Initially, Dong *et al.* [23], proposed a three-layer CNN architecture. The first convolutional layer implicitly extracts a set of feature maps for the input LR image, the second layer maps these feature maps nonlinearly to HR patch representations and the third layer reconstruct the HR image from these patch representations. Several studies have further investigated CNN-based architectures for image SR. Among other, the following features have been reported to improve SR performance: an increased depth of the network [24], residual block (with batch normalization and skip connection) [25], sub-pixel layer [26], perceptual loss function (instead of Mean Square Error-based cost functions) [27, 25, 28], recurrent networks [29], generative adversarial networks [25].

Very recently, very deep architectures obtained the best performance in a challenge focusing on natural image SR (NTIRE 2017 challenge [30]). However, due to the variety of the proposed methods and the high number of parameters for the networks architecture design, it is currently difficult to identify the key elements of CNN architecture to achieve good performance for image SR and assess their applicability in the context of 3D brain MRI. In addition the extension of CNN architectures to 3D images, taking into account floating and possibly anisotropic scaling factors may be of interest to address the wide range of possible clinical acquisition settings, whereas classical CNN architectures only address a predefined (integer) scaling factor. The availability of multimodal imaging setting also questions the ability of CNN architectures to benefit from such multimodal data to improve the SR of a given modality.

Contributions: This work presents a comprehensive review of deep convolutional neural networks, and associated key elements, for brain MRI SR. Following Timofte *et al.* [31], who have experimentally showed several ways to improve SR techniques from a baseline architecture, we study the impact of eight key elements on the performance of convolutional neural networks for 3D brain MRI SR. We demonstrate empirically that residual learning associated with appropriate optimization methods can significantly reduce the time of the training step and fast convergence can be achieved in 3D SR context. Overall, we report better performance when learning deeper fully 3D convolution neural networks and using larger filters. Interestingly, we demonstrate that a single network can handle multiple arbitrary scale factors efficiently, for example, from $2 \times 2 \times 2$ mm to $2 \times 2 \times 1$ mm or $1 \times 1 \times 1$ mm or $0.67 \times 0.67 \times 0.67$ mm, by learning multiscale residuals from spline-interpolated image. We also report significant improvement using a multimodal architecture, where a HR reference image can guide the CNN-based SR of a given MRI volume.

2. Super-Resolution using Deep Convolutional Neural Networks

2.1. Related work and learning-based SR

Single image SR is a typically ill-posed inverse problem that can be stated according to the following linear formulation:

$$\mathbf{Y} = \mathbf{HX} + \mathbf{N} = D_{\downarrow} B\mathbf{X} + \mathbf{N} \tag{1}$$

where one LR image $\mathbf{Y} \in \mathbb{R}^n$, $\mathbf{X} \in \mathbb{R}^m$ is the HR image, $\mathbf{H} \in \mathbb{R}^{m \times n}$ is the observation matrix ($m > n$) and $\mathbf{N}$ denotes an additive noise. $D_{\downarrow}$ represents the downsampling operator and B is the blur matrix. The purpose of SR methods is to estimate $\mathbf{X}$ from the observations $\mathbf{Y}$. The SR image can be estimated by minimizing a least-square cost function:

$$\widehat{\mathbf{X}} = \arg\min_{\mathbf{X}} \|\mathbf{Y} - \mathbf{HX}\|^2. \tag{2}$$

In an unsupervised context, the optimization process usually leads to unstable solutions and requires appropriate regularization techniques. However, in a supervised context, regularization can be implicit and the HR image can be estimated with the following formulation:

$$\widehat{\mathbf{X}} = \arg\min_{\mathbf{X}} \|\mathbf{X} - \mathbf{H}^{-1}\mathbf{Y}\|^2. \tag{3}$$

In this setting, the matrix $\mathbf{H}^{-1}$ can be modeled as a combination of a restoration matrix $F \in \mathbb{R}^{m \times m}$ and an upscaling interpolation operator $S^{\uparrow} : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Given a set of HR images $\mathbf{X}_i$ and their corresponding LR images $\mathbf{Y}_i$ with K samples, the restoration operator F can be estimated as follows:

$$\widehat{F} = \arg\min_{F} \sum_{i=1}^{K} \|\mathbf{X}_i - F(S^{\uparrow}\mathbf{Y}_i)\|^2 = \arg\min_{F} \sum_{i=1}^{K} \|\mathbf{X}_i - F(\mathbf{Z}_i)\|^2 \tag{4}$$

where $\mathbf{Z} \in \mathbb{R}^m$ is the interpolated LR (ILR) version of $\mathbf{Y}$ (*i.e.* $\mathbf{Z} = S^{\uparrow}\mathbf{Y}$). F is then a mapping from the ILR image space to the HR image space.

SR is the process of estimating HR data from LR data. The main goal is then to estimate high-frequency components from LR observations. Instead of learning the mapping directly from the LR space to the HR one, it might be easier to estimate a mapping from the LR space to the missing high-frequency components, also called the residual between HR and LR data: $\mathbf{R} = \mathbf{X} - \mathbf{Z}$ or equivalently $\mathbf{X} = \mathbf{Z} + \mathbf{R}$. This approach can be modeled by a skip connection in the network. In such a residual-based modeling, one typically assumes that $\mathbf{R}$ is a function of $\mathbf{Z}$. The computation of HR data is then expressed as follows: $\mathbf{X} = \mathbf{Z} + F(\mathbf{Z})$ where F can be learned using the following equation:

$$\widehat{F} = \arg\min_{F} \sum_{i=1}^{K} \|(\mathbf{X}_i - \mathbf{Z}_i) - F(\mathbf{Z}_i)\|^2. \tag{5}$$

2.2. CNN-based baseline architecture

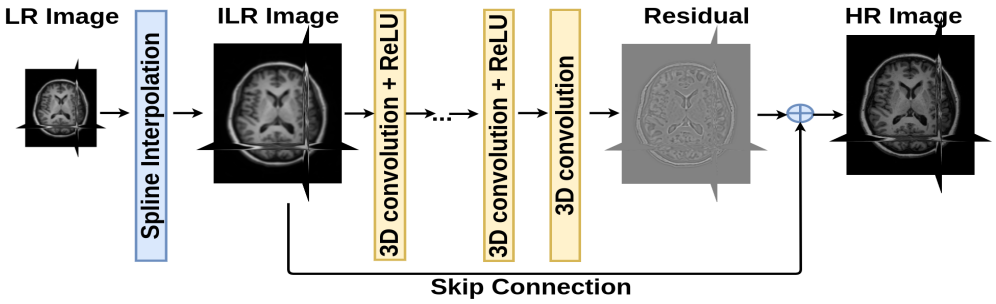


Figure 1: 3D deep neural network for single brain MRI super-resolution.

In this paper, we focus on the learning of mapping F with convolutional neural networks. Following Dong *et al.* [23] and Kim *et al.* [24], mapping F from $\mathbf{Z}$ to $(\mathbf{X} - \mathbf{Z})$ is decomposed into nonlinear operations corresponding to the combination of convolution-based and rectified linear unit (ReLU) layers.

The baseline architecture used in this work can be described as follows:

$$\begin{cases} F_1(\mathbf{Z}) = \max(0, W_1 * \mathbf{Z} + B_1) \\ F_i(\mathbf{Z}) = \max(0, W_i * F_{i-1}(\mathbf{Z}) + B_i) \quad for \quad 1 < i < L \\ F_L(\mathbf{Z}) = W_L * F_{L-1}(\mathbf{Z}) + B_L \end{cases} \tag{6}$$

where:

- L is the number of layers,
- W_i and B_i are the convolution parameters to learn. W_i corresponds to n_i convolution filters of support $c \times f_i \times f_i \times f_i$, where c is the number of channels in the input of layer i, f_i and n_i are respectively the spatial size of the filters and the number of filters of layer i,
- $\max(0, \cdot)$ refers to a ReLU applied to the filter responses.

This network architecture is depicted in Figure 1(a). Please note that, for instance, the SRCNN model [23] corresponds to a specific parameterization of this baseline architecture ($f_1 = 9$, $f_2 = 1$, $f_3 = 5$, $n_1 = 64$, $n_2 = 32$ and with no skip connection).

The performance of a given architecture depends on several parameters such as the filter size f_i, the number of filters n_i, the number of layers L, etc. Understanding how these parameters affect the reconstruction of the HR image with respect to the considered application setting (e.g., number of training samples, image size, scaling factor) is a key issue, which remains poorly explored. For instance, regarding the number of layers, it is commonly believed that the deeper the better [19, 24]. However, adding layers increases the number of parameters and can lead to overfitting. Previous works [23, 32], have shown that "a deeper structure does not always lead to better results" [23].

Specifically focusing on MRI data, the specific objectives of this study are: i) the evaluation and understanding of the effect of key elements of CNN for brain MRI SR, ii) the experimental study of arbitrary multi-scale SR using CNN, iii) investigating multimodality-guided SR using CNN.

3. Sensitivity Analysis of the considered Architecture

In this section, we present the MRI datasets used for evaluation and the key elements of CNN architecture to achieve good performance for single image SR.

3.1. MRI Datasets and LR simulation

To evaluate SR performances of CNN-based architectures, we have used two MRI datasets: the Kirby 21 dataset and the NAMIC Brain Multimodality dataset.

The Kirby 21 dataset [33] consists of MRI scans of twenty-one healthy volunteers with no history of neurological conditions. Magnetization prepared gradient echo (MPRAGE, T1-weighted) scans were acquired using a 3-T MR scanner (Achieva, Philips Healthcare, The Netherlands) with a $1.0 \times 1.0 \times 1.2$ mm^3 resolution over an FOV of $240 \times 204 \times 256$ mm acquired in the sagittal plane. Flair data were acquired using $1.1 \times 1.1 \times 1.1$ mm^3 resolution over an FOV of $242 \times 180 \times 200$ mm acquired in the sagittal plane. The T2-weighted volumes were acquired using a 3D multi-shot turbo-spin echo (TSE) with a TSE factor of 100 with over an FOV of $200 \times 242 \times 180$ mm including a sagittal slice thickness of 1 mm. Although, dataset Kirby 21 has repeated scans, we ensure that we do not have the same subjects in both train and test set.

The NAMIC Brain Multimodality [34] dataset consist of T1-weighted and T2-weighted data of ten normal controls and ten schizophrenic subjects with isotropic resolution of $1 \times 1 \times 1mm^3$. These scans have been acquired using a 3T GE at BWH in Boston with the following parameters : TR=$7.4ms$, TE=$3ms$, TI=600, 10 degree flip angle, 25.6 cm^2 FOV, matrix=256×256.

As in [6], LR images have been generated from a Gaussian blur with standard deviation $\sigma = 1$ and a down-sampling by isotropic scaling factors. In the training phase, a set of patches of training images is randomly extracted. In the baseline setting, the training dataset comprises 10 subjects (KKI33 to KKI42, 3200 patches $25 \times 25 \times 25$ per subject

randomly sampled) and the testing dataset is composed of 5 subjects (KKI01 to KKI05). During the testing step, the network is applied on the whole images. The peak signal-to-noise ratio (PSNR) in decibels (dB) is used to evaluate the SR results with respect to the original HR images.

3.2. Baseline and benchmarked architectures

The network architecture that is used as a baseline approach in this study is illustrated in Figure 1 (a). The baseline network is a 10 blocks (convolution+ReLU) network with the following parameters: 64 convolution filters of size ($3 \times 3 \times 3$) at each layer, mean square error (MSE) as loss function, weight initialization by [35] (MSRA filler), Adam (adaptive moment estimation) method for optimization [36], 20 epochs on Nvidia GPU and using Caffe package [37], batch size of 64, learning rate set to 0.001, no regularization or drop out has been used. The learning rate multipliers of weights and biases are respectively 1 and 0.1. For benchmarking purposes, we consider two other state-of-the-art SR models: low-rank total variation (LRTV) [6] and SRCNN3D [38].

The next sections present the impact of the key parameters studied in this work: optimization method, weight initialization, residual-based model, network depth, filter size, filter number, training patch size and size of training dataset.

3.3. Optimization Method

In the context of supervised learning, given a training dataset which consists of pairs of LR and HR images, network parameters are estimated by minimizing the objective function using optimization algorithms. These algorithms play a very important role in training neural networks. The more efficient and effective optimization strategies lead to faster convergence and better performance. More precisely, during the training step, the estimation of the restoration operator F corresponds to the minimization of the objective function $\mathcal{L}$ in Equation 5 over network parameters $\boldsymbol{\theta} = \{W_i, B_i\}_{i=1,\ldots,L}$.

Most optimization methods for CNNs are based on gradient descent. A classic method applies a mini-batch stochastic gradient descent with momentum (SGD [39]) as used in [23, 38]. The following equations are used to update the network parameters $\boldsymbol{\theta}$ at iteration $t+1$:

$$\begin{aligned} V_{t+1} &= \mu V_t - \alpha \nabla \mathcal{L}(\boldsymbol{\theta}_t) \\ \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t + (V_j)_{t+1} \end{aligned} \tag{7}$$

where $\nabla \mathcal{L}(\boldsymbol{\theta}_t)$ is the negative gradient of the objective function of current network parameters $\boldsymbol{\theta}_t$ at iteration t, V_t is called the weight update, μ and α denote respectively the momentum and learning rate. However, the fixed momentum causes numerical instabilities around the minimum. Nesterov's accelerated gradient (NAG) [40] was proposed to cope with this issued using the following update:

$$\begin{aligned} V_{t+1} &= \mu V_t - \alpha \nabla \mathcal{L}(\boldsymbol{\theta}_t + \mu V_t) \\ \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t + V_{t+1} \end{aligned} \tag{8}$$

However, when using small learning rates α with SGD or NAG (e.g. $\alpha \leq 10^{-4}$), deeper networks converge very slowly. By contrast, high learning rates may lead to exploding gradients [41, 42]. In order to address this issue, Kim *et al.*. [24] proposed a stochastic gradient descent with an adjustable gradient clipping (SGD-GC) [43] to achieve an optimization with high learning rates (e.g. $\alpha = 0.1$). When the norm of gradient goes over a threshold γ, this gradient is rescaled as follows:

$$\nabla\mathcal{L}(\boldsymbol{\theta}) = \begin{cases} \nabla\mathcal{L}(\boldsymbol{\theta})/\gamma & \| \nabla\mathcal{L}((\boldsymbol{\theta}) \| > \gamma \\ \nabla\mathcal{L}(\boldsymbol{\theta}) & otherwise \end{cases} \tag{9}$$

The predefined range over which gradient clipping is applied may still cause SGD-GC not to converge quickly or make difficult the tuning of a global learning rate. Recently, methods have been proposed to address this issue through an automatic adaption of the learning rate for each parameter to be learned. RMSProp and Adam [44, 36] are the two most popular models in this category. RMSProp (root mean square propagation) [44] method updates trainable weights by rescaling the gradients by the root mean square of its second moments u as:

$$\begin{aligned} u_t &= \delta u_{t-1} + (1-\delta)\nabla\mathcal{L}(\boldsymbol{\theta}_t)^2 \\ \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t - \alpha\frac{\nabla\mathcal{L}(\boldsymbol{\theta}_t)}{\sqrt{u_t}} \end{aligned} \tag{10}$$

where δ is called RMSProp decay. But RMSProp ignores the first moment of gradients and bias corrections, which may lead to very large step sizes and divergence [36]. Adam (Adaptive moment estimation) method [36] uses a first-order stochastic gradient-based optimization, which relies on adaptive estimates of both the first and second moments of the gradients (m, u). The Adam method applies the following update:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1)\nabla\mathcal{L}(\boldsymbol{\theta}_t) \\ u_t &= \beta_2 u_{t-1} + (1-\beta_2)\nabla\mathcal{L}(\boldsymbol{\theta}_t)^2 \\ \hat{m}_t &= m_t/(1-(\beta_1)^t) \\ \hat{u}_t &= u_t/(1-(\beta_2)^t) \\ \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t - \alpha\frac{\hat{m}_t}{\sqrt{\hat{u}_t+\epsilon}} \end{aligned} \tag{11}$$

where β_1 and β_2 are the first and second moment decay rates, and α is a predefined parameter. $(\hat{m}_j)_t$ and $(\hat{v}_j)_t$ are called respectively the moment bias corrections of the first and second moment estimates.

The results of four optimization methods (NAG, SGD-GC, RMSProp and Adam) for the baseline network are illustrated in Figure 2. Firstly, regardless the method used, the baseline network shows better performance than LRTV [6] and SRCNN3D [38]. Secondly, it can be observed that the baseline network can converge very fast and stably (only 20 epochs with small learning rate of 0.001). Finally, in these experiments, the most efficient and effective optimization method is Adam as regards both PSNR metric and convergence speed. Hence, in the next sections, we use Adam method with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ to train our networks.

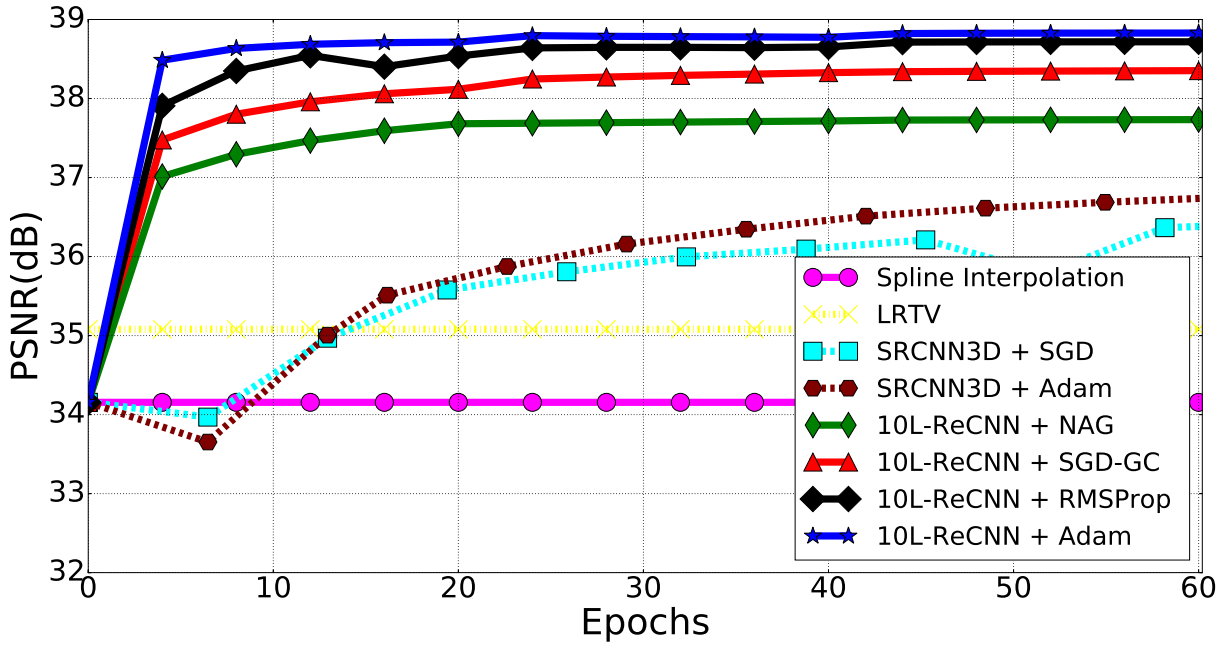
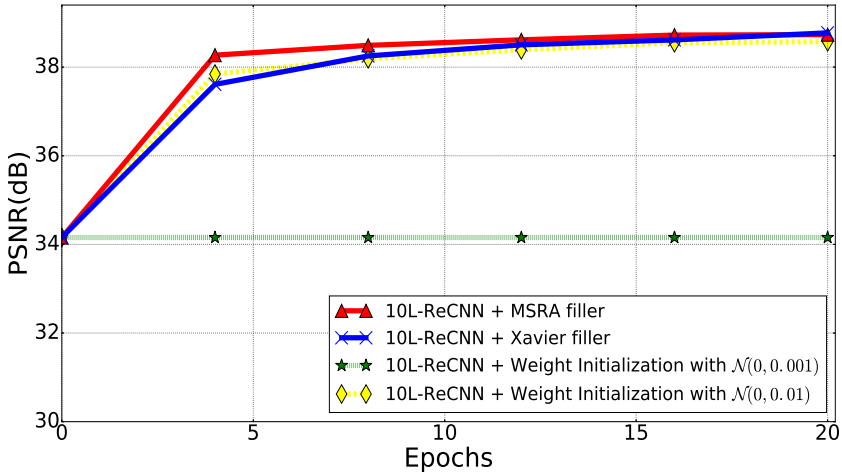


Figure 2: Impact of the optimization methods onto SR performance: SGD-GC, NAG, RMSProp and Adam optimisation of a 10L-ReCNN (10-layer residual-learning network with $f = 3$ and $n = 64$). We used Kirby 21 [33] for training and testing with isotropic scaling factor $\times 2$. The initial learning rates of SGC-GC, NAG, RMSProp and Adam are set respectively to 0.1, 0.0001, 0.001 and 0.001. These learning rates are decreased by a factor of 10 every 20 epochs. The momentum of these methods, except RMSProp, is set to 0.9. All optimization methods use the same weight initialization described in [35].

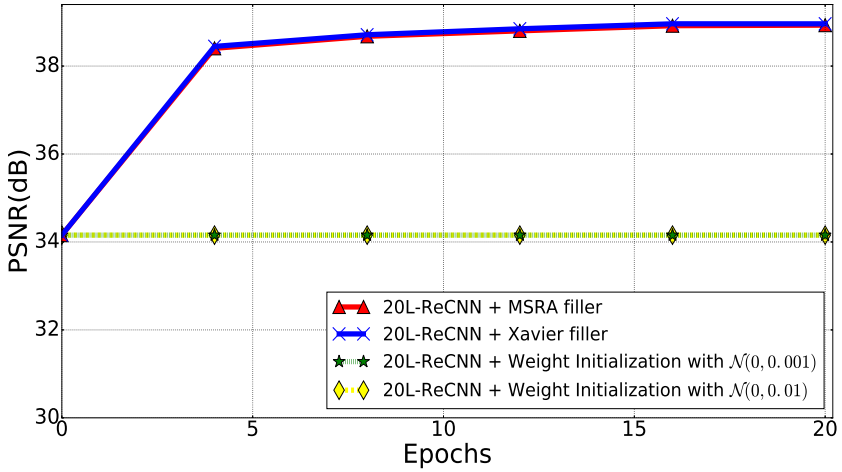
3.4. Weight Initialization

The optimization algorithms for training a CNN are typically initialized randomly. Inappropriate initialization can lead to long time convergence or even divergence. Several studies [23, 32, 38] used a normal distribution $\mathcal{N}(0, 0.001)$ to initialize the weights of convolutional filters. However, because of too small initial weights, the optimizer can be stuck into a local minimum especially when building deeper networks. Both SRCNN [23] concluded that deeper networks do not lead to better performance, and [32] confirmed that the addition of extra convolutional layers to the 7-layer model is found to be ineffective. Deeper neural networks are more difficult to train because of exploding gradients with 2D CNNs [41] due to the multiplication of the variance at every layer by the number of weights $n \times f \times f \times f$ (i.e. nf^3) [42]. Uniform distribution $\mathcal{U}(-\sqrt{3/(nf^3)}, \sqrt{3/(nf^3)})$ (called Xavier filler [42]) was also proposed to initialize the weights of deeper networks. In order to add more layers to networks, He *et al.* [35] suggested an initial training stage by sampling from the normal distribution $\mathcal{N}(0, \sqrt{2/(nf^3)})$ (called here Microsoft Research Asia - MSRA filler). Several deep networks adopted effectively this scheme such as very deep CNNs for SR [24], ResNet [20] and the brain lesion segmentation using 3D CNN [21].

Overall, we evaluate here the weight initialization schemes described in [42] and [35], a normal distribution $\mathcal{N}(0, 0.001)$ as in [23, 32] and a normal distribution $\mathcal{N}(0, 0.01)$ for the considered SR architecture. Experiments with a deeper architecture were also performed,



(a) : 10-layer residual-learning networks (10L-ReCNN)



(b) : 20-layer residual-learning networks (20L-ReCNN)

Figure 3: Weight Initialization Scheme vs Performance (residual-learning networks with the same filter numbers $n = 64$ and filter size $f = 3$ using Adam optimization and tested with isotropic scaling factor $\times 2$ using Kirby 21 for training and testing, 32000 patches with size 25^3 for training).

more precisely for a 20-layer architecture, which is the deepest architecture that could be implemented for the considered experimental setup due to GPU memory setting. As shown in Figure 3, the initialization with normal distributions $\mathcal{N}(0, 0.001)$ failed to make the training of both 10-layer and 20-layer residual-learning networks converge. In addition, our 20-layer network also does not converge when initialized with normal distributions $\mathcal{N}(0, 0.01)$. By contrast, MSRA and Xavier filler schemes make the networks converge and reach similar reconstruction performance. For the rest of this paper, we use MSRA weight filler as initialization scheme [35].

3.5. Residual Learning

The CNN methods in [23, 26, 45] use the LR image as input and outputs the HR one. We refer to such approach as a non-residual learning. Within these approaches, low-frequency

features are propagated through the all network, which may decrease the sparsity overall the network structure and in turn the computational efficiency of the training stage. By contrast, one may consider residual learning or normalized HR patch prediction as pointed out by numerous learning-based SR model [46, 47, 48, 24]. When considering CNN methods, one may design a network which predicts the residual between the HR image and the output of the first transposed convolutional layer [32]. Using residual blocks, a CNN architecture may implicitly embed residual learning while still predicting the HR image [25].

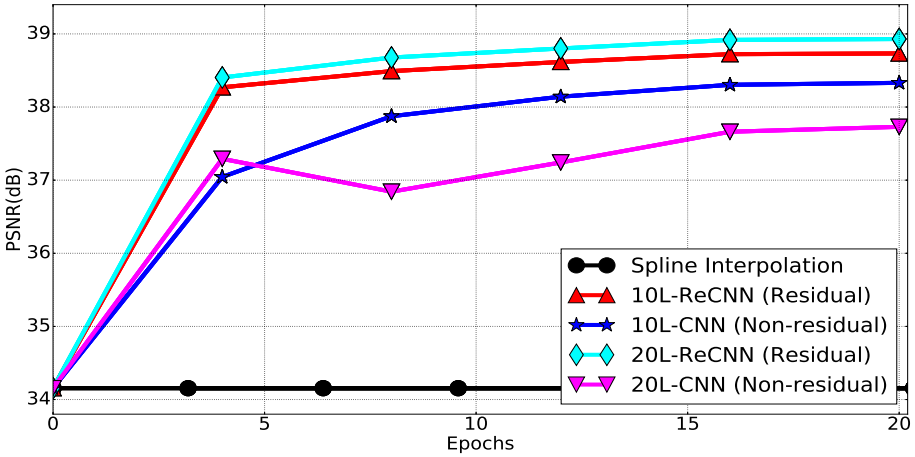


Figure 4: Non-residual-learning vs Residual-learning networks with the same $n = 64$ and $f^3 = 3^3$ and the depths of 10 and 20 (called here 10L-CNN vs 10L-ReCNN and 20L-CNN vs 20L-ReCNN) over 20 training epochs using Adam optimization with the same training strategy and tested with isotropic scale factor $\times 2$ using Kirby 21 for training and testing.

Here, we perform a comparative evaluation of non-residual learning vs. residual learning strategies. Figure 4 depicts PSNR values and convergence speed of residual vs non-residual network structures with 10 and 20 convolutional layers. The residual-learning networks converge faster than the non-residual-learning ones. In addition, residual learning leads to significant improvements in PSNR (+0.4dB for 10 layers and +1.2dB for 20 layers). It might be noted that these experiments do not support the common statement that the deeper, the better for CNNs. Here, the use of additional layers is only beneficial when using residual modeling. Deeper architectures even lower the reconstruction performance with non-residual learning.

3.6. Depth, Filter Size and Number of Filters

As shown by the previous experiment, the link between network depth and performance remains unclear. Besides, it is hard to train deeper networks because gradient computation can be unstable when adding layers [42]. For instance, Oktay *et al.* tested extra convolutional layers to a 7-layer model but achieved negligible performance improvement [32]. As mentioned in the Section 2.2, SRCNN [23] was also tested with deeper architectures but no improvement was reported. However in [24] Kim *et al.* argue that the performance of CNNs

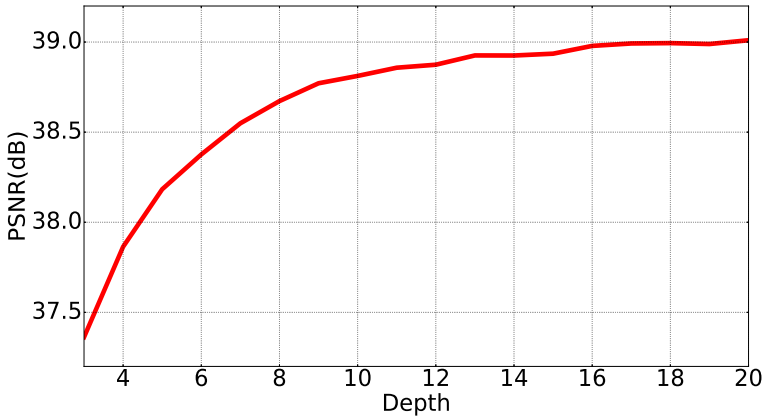


Figure 5: Depth vs Performance (residual-learning networks with the same filter numbers $n = 64$ and filter size $f = 3$ over 20 training epochs using Adam optimization and tested with isotropic scale factor $\times 2$ using Kirby 21 for training and testing, 32000 patches with size 25^3 for training).

for SR could be improved by increasing the depth of network compared to neural network architectures in [23, 32].

The previous section supports that deeper architectures may be beneficial when considering a residual learning. We further evaluate here the reconstruction performance as a function of the number of layers. Results are reported in Figure 5. They stress that increasing network depth with residual learning significantly improves the quality of the estimated HR image (e.g. +1.6dB increasing of the depth from 3 to 20 or +0.5dB increasing of the depth from 7 to 20).

The parameterization of the convolutional filters is also of key interest. Inspired by the VGG network designed for classification [19], previous CNN methods for SR mostly focused on small convolutional filters of size $(3 \times 3 \times 3)$ in [24, 32, 21]. Oktay *et al.*. [32] even argued that such architecture can lead to better non-linear estimations. Regarding the number of filters for each layer, [23] reported greater reconstruction performance when increasing the number of filters. No such experimental evaluation was reported in the studies we further explored CNN-based SR [24, 32] . Here, we both evaluate the effect of the filter size and of the number of filters.

Figure 6 shows that a 10-layer network with a filter size of 5^3 leads to better results than a 20-layer network with 3^3 filters. Besides reconstruction performance, the use of a larger filter size decreases the training speed and significantly increases the complexity and memory cost for training. For example, it took us 50 hours to train a 10-layer network with a filter size of 5^3. By contrast, a deeper network with smaller filler (i.e. 20-layer network with 3^3 filters) involves a smaller number of parameters, such that it took us only 24 hours to train. These experiments suggest that deeper architectures with small filters can relevantly replace shallower networks with larger filters both in terms of computational complexity and of reconstruction performance.

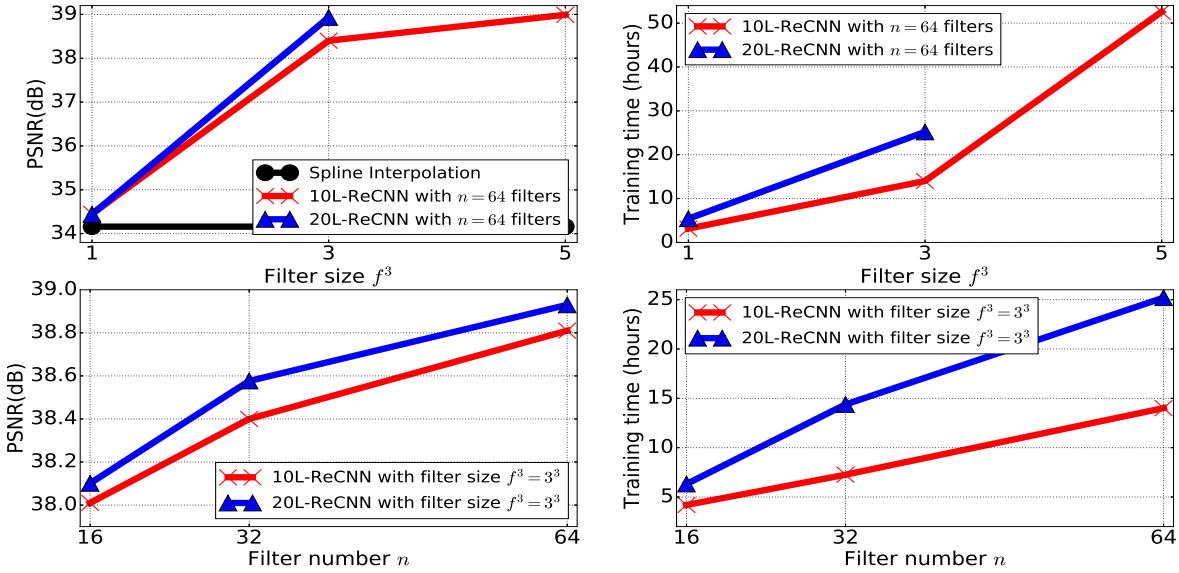


Figure 6: Impact of convolution filter parameters (sizes $f \times f \times f = f^3$ with n filters) on PSNR and computation time. These 10-layers residual-learning networks are trained from scratch with Adam optimization over 20 epochs and tested with Kirby 21 for isotropic scale factor $\times 2$.

3.7. Training Patch Size and Subject Number

In the context of brain MRI SR, the acquisition and collection of large datasets with homogeneous acquisition settings is a critical issue. We here evaluate the extent to which the number of training subjects affects SR reconstruction performance. It may be noted that data augmentation [49, 50, 24, 31], which is widely used for application to natural images, does appear relevant in our context. Brain MRI images have the same direction and orientation and share the same anatomical form. As the training exemplars are extracted as patches of brain MRI images, we also evaluate the impact of the training patch size onto learning and reconstruction performance.

The size of training patches should be larger or equal to the size of the receptive field of the considered network [19, 24], which is given by $((f-1)D+1)^3$ for a D-layer network with filter size f^3. Figure 7 confirms that better performance can be achieved using larger training patches (from 11^3 to 31^3 with the 10-layer network and from 11^3 to 29^3 with the 12-layer network). However, if the patch size is larger than the receptive field (e.g. 21^3 within the 10-layers network and 25^3 within the 12-layers network), the improvement is negligible.

We stressed previously that the selection of the network depth involves a trade-off between reconstruction performance and GPU memory requirement and training time increase. A similar pattern can be drawn with respect to the patch size. Figure 7 illustrates that larger training patch sizes also require more memory for training. Moreover, this figure shows that better performance can be achieved using larger training patch sizes. It may be noted that the relatively lower performance of the 10-layer networks may reach a performance similar to 12-layer and 20-layer networks when using larger training patches.

Regarding the number of training subjects, Figure 8 points out that a single subject

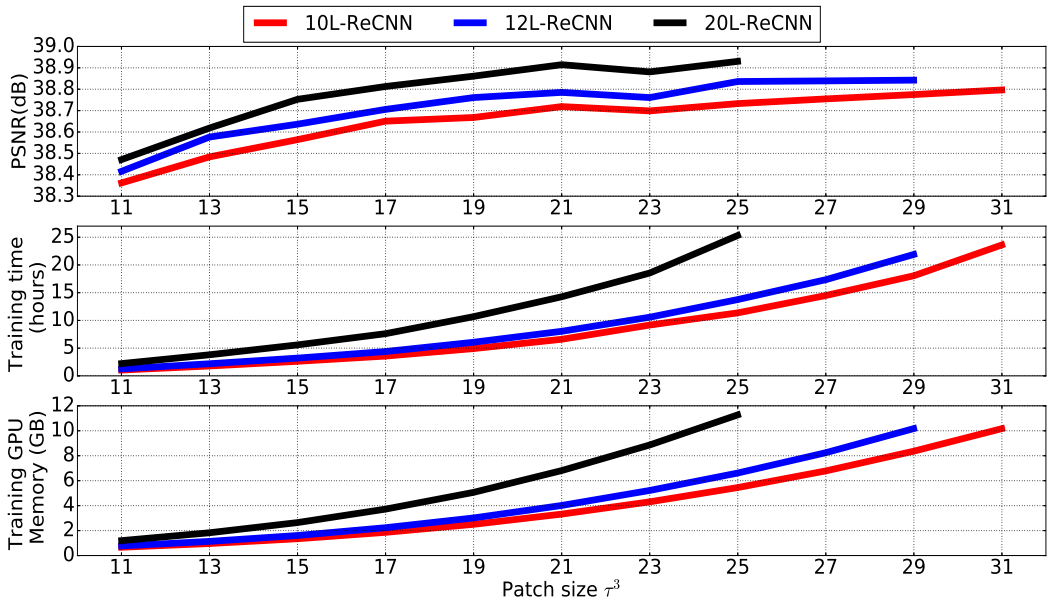


Figure 7: First row: Training patch size vs Performance. Second row: Patch size vs Training Time. Third row: Patch size vs Training GPU Memory Requirement. These networks with the same $n = 64$ and $f^3 = 3^3$ are trained from scratch with batch of 64 and tested with Kirby 21 for isotropic scale factor $\times 2$.

is enough to reach better performance than spline interpolation. Interestingly, reconstruction performance reaches a plateau when about 10 subjects are considered, which appears appropriate for real-world applications.

4. Handling Arbitrary Scales

In some CNN-based SR approaches, the networks are learned for a fixed and specified scaling factor. Thus, a network built for one scaling factor cannot deal with any other scale. In medical imaging, Oktay *et al.*. [32] have applied CNNs for upscaling cardiac image slices with the scale of 5 (e.g. upscaling the voxel size from $1.25 \times 1.25 \times 10.00mm$ to $1.25 \times 1.25 \times 2.00mm$). Typically, their network is not capable of handling other scales due to the use of fixed deconvolutional layers. In brain MRI imaging, the variety of the possible acquisition settings motivates us to explore multi-scale settings.

Following Kim *et al.* [24], we investigate how we may embed multiple scales in a single network. It consists in creating a training dataset within which we consider LR and HR image pairs corresponding to different scaling factors. To avoid a converge towards a local minimum of one of the scale factor, we learn network parameters on randomly swapped dataset.

Table 1 summarizes experimental results. First, when the training is achieved for the scaling from $(2 \times 2 \times 2)$ on a dataset of $(2 \times 2 \times 2)$ scale, it can be noticed that reconstruction performances decrease very fast when applied to other scaling factors (there is a drop from $39.01dB$ to $33.43dB$ when testing with $(3 \times 3 \times 3)$). Second, it can be noticed that when the training is performed on multi-scale data, there is no significant performance change

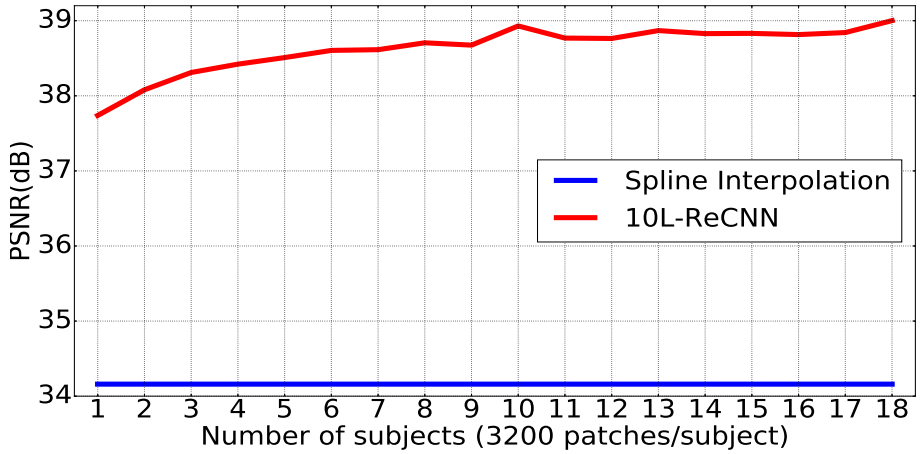


Figure 8: Number of Subjects vs Performance (10-layer residual-learning networks with the same filter numbers $n = 64$ and filter size $f = 3$ over 20 training epochs using Adam optimization and tested with isotropic scale factor $\times 2$ using Kirby 21 for training and testing, 3200 patches per subject with size 25^3 for training).

compared to training from a single-scale dataset. Training from multiple scaling factors leads to the estimation of a more versatile network. Overall, these result shows that one single network can handle multiple arbitrary scaling factors efficiently.

Test / Train	Full-training		
	$\times$(2,2,2)	$\times$(3,3,3)	$\times$(2,2,2),(3,3,3)
$\times$(2,2,2)	**39.01**	35.25	**38.80**
$\times$(2,2,3)	36.80	35.11	37.24
$\times$(2,2.5,2)	37.71	35.41	37.93
$\times$(2,3,3)	35.23	35.13	36.20
$\times$(2.5,2.5,2.5)	35.47	35.52	36.63
$\times$(3,3,3)	33.43	**35.01**	**35.20**

Table 1: Experiments with multiple isotropic scaling factors with the 20-layers network. **Bold numbers** indicate that the tested scaling factor is present in the training dataset.

5. Multimodality-guided SR

In a typical clinical setting, it is common to acquire one isotropic HR image and other LR images in order to limit the acquisition time. Hence, a coplanar isotropic HR image might be considered as a complementary information source to reconstruct HR MRI images from LR ones [3]. To address this multimodality-guided SR problem, we add a concatenation layer as the first layer of the network as illustrated in Figure 9. This layer concatenates the ILR image and a registered HR reference along the channel axis. The registration step of HR reference ensures that the two input images share the same geometrical space.

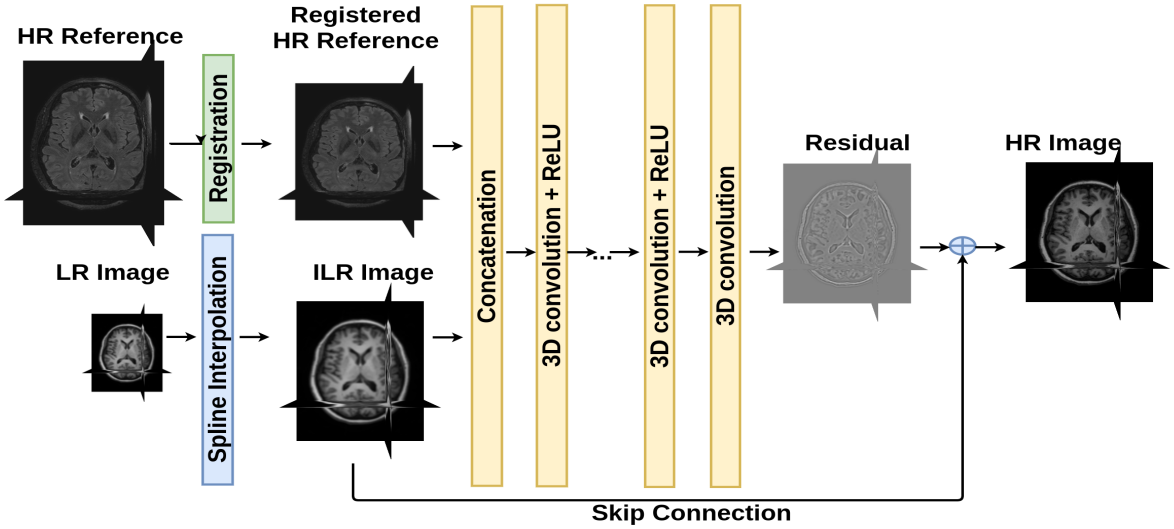
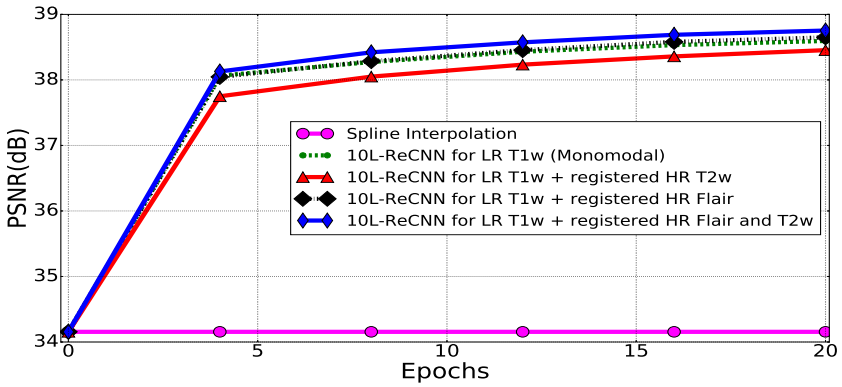


Figure 9: 3D deep neural network for multimodal brain MRI super-resolution using intermodality priors. Skip connection computes the residual between ILR image and HR image.

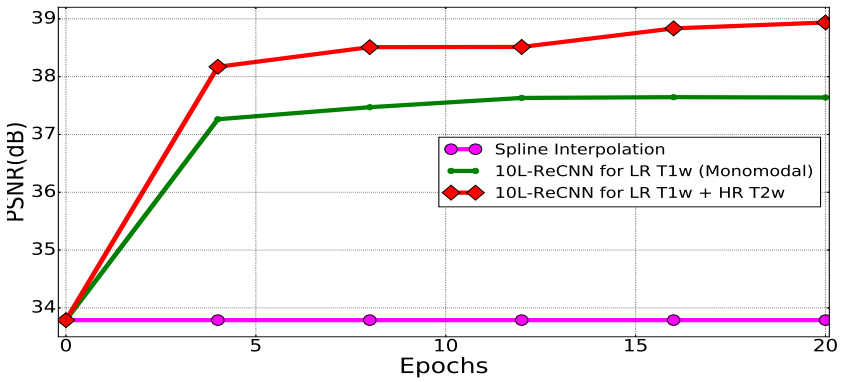
We experimentally evaluate the relevance of the proposed multimodality-guided SR model according to the following setting. We investigate whether the complementary use of a Flair or T2-weighted MRI image might be beneficial to improve the reconstruction HR T1-weighted MRI images from LR ones. In our experiments, we use T1-weighted, T2-weighted, Flair MRIs of the Kirby 21 dataset and T1-weighted, T2-weighted MRIs of NAMIC dataset. We apply FSL software [51] for the registration of Flair and T2-weighted data onto T1-weighted scans and we adopt the observation model and the baseline network as in the Section 3.2 (10L-ReCNN). Due to the same resolution of T1-weighted and T2-weighted scans of NAMIC, no registration is required. The scans of 10 subjects of NAMIC are used for training and 10 other subjects for testing.

Figure 10 shows the results of the multimodailty-guided SR compared to the monodal SR for both Kirby dataset (a) and NAMIC datasets (b). It can be seen that multimodality driven approach can lead to improved reconstruction results. It however depends on the quality of the HR image used to drive the reconstruction process. For instance, when considering T2w images, no improvement is observed for Kirby dataset and a significant improvement greater than $1dB$ is reported for NAMIC dataset. As T2w image resolution is lower than T1w modality in Kirby dataset, these results may emphasize the requirement for actual HR information source to expect significant gain w.r.t. the monomodal model.

Figure 12 highlights the fact that the proposed multimodal method provides a more favorable result compared to interpolation and monomodal methods. In addition, we explore the impact of the network depth augmentation with regard to the performance of multimodal SR approach. The experiments shown in Figure 11 indicate that the deeper structures do not lead to better results within the multimodal method.



(a) Multimodal experiments using Kirby dataset for training and testing. Images are registered using FSL [51].



(b) Multimodal experiments using NAMIC dataset for training and testing. No image registration is performed.

Figure 10: Multimodality-guided SR experiments. The LR T1-weighted images are upscaled with isotropic scale factor $\times 2$ using respectively monomodal network (10L-ReCNN for LR T1w), HR T2w multimodal network, HR Flair multimodal network and both HR Flair and T2w multimodal images.

6. Transfer Learning

Training a CNN from scratch requires a substantial amount of training data and may take a long time. Moreover, to avoid overfitting, the training dataset has to reflect the appearance variability of the images to reconstruct. In the context of brain MRI, part of image variability comes from acquisition systems. Hence, we investigate the impact of such image variability onto SR performance by evaluating transfer learning skills among different datasets corresponding to the same imaging modality.

In order to characterize such transfer learning skills, we evaluate the extent to which the selection of a given training dataset affects the reconstruction performance of the network. We proceed as follows. We train from scratch two 20L-ReCNN networks respectively for a 10-image NAMIC T1-weighted dataset and a 10-image Kirby T1-weighted dataset, and we test the trained models for the remaining 10-image NAMIC and Kirby T1-weighted datasets. The considered case-study involves a scaling factor of $(2 \times 2 \times 2)$. For quantitative comparison, the PSNR and SSIM [52] are used to evaluate the performance of each model

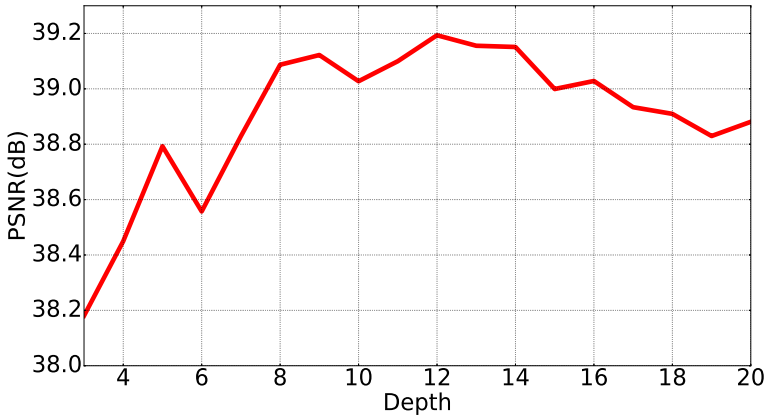


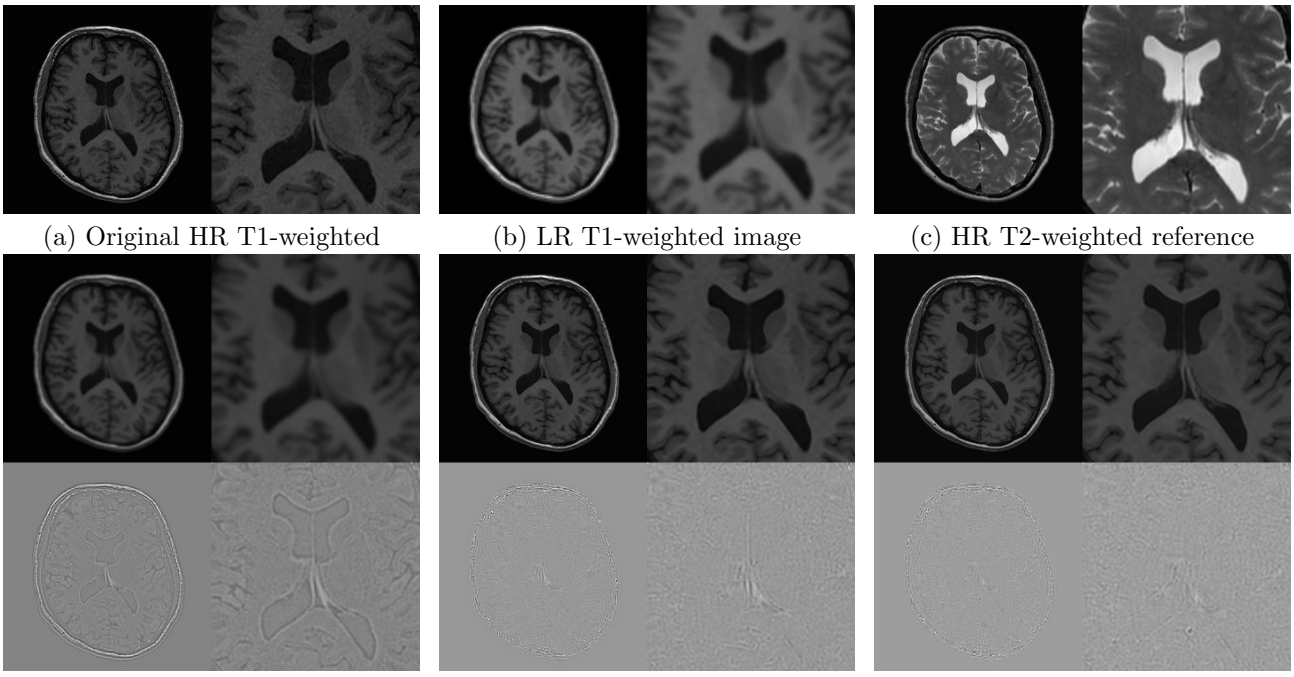
Figure 11: Depth vs Performance (multimodal SR using residual-learning networks with the same filter numbers $n = 64$ and filter size $f = 3$ over 20 training epochs using Adam optimization and tested with isotropic scale factor $\times 2$ using NAMIC for training and testing).

in Table 2. For benchmarking purposes, we also include a comparison with the following methods: cubic spline interpolation, low-rank total variation (LRTV) [6], SRCNN3D [38]. The use of 20-layer CNN-based approaches for each training dataset can lead to significant improvement over spline interpolation, LRTV method and SRCNN3D (with respect to both PSNR and SSIM). Although, we lose a little gain (e.g. PSNR: 0.55dB for testing Kirby and 0.74dB for NAMIC, SSIM: 0.003 for Kirby and 0.0019 for NAMIC) when using different training and testing dataset (i.e. different resolution), our proposed networks have better results than compared methods.

For qualitative comparison, Figures 13 and14 show the results of reconstructed 3D images obtained from all the compared techniques. The HR reconstruction of the proposed 20L-ReCNN model best preserves contours and geometrical structures. It also visually better recovers the image contrast compared with the other methods.

Testing dataset	Spline Interpolation		LRTV [6]		SRCNN3D Kirby (10 images)		20L-ReCNN Kirby (10 images)		20L-ReCNN NAMIC(10 images)	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Kirby (5 images)	34.16	0.9402	35.08	0.9585	37.51	0.9735	38.80	0.9797	38.06	0.9767
Standard deviation	1.90	0.0111	2.09	0.0083	1.97	0.0053	1.84	0.0044	1.83	0.0045
Gain	-	-	0.92	0.0183	3.36	0.0333	4.64	0.0395	3.9	0.0365
NAMIC (10 images)	33.78	0.9388	34.34	0.9549	36.72	0.9694	37.73	0.9762	38.28	0.9781
Standard deviation	1.82	0.0071	1.79	0.0044	1.76	0.0035	1.81	0.0031	1.78	0.0029
Gain	-	-	0.56	0.0161	2.94	0.0306	3.95	0.0374	4.5	0.0393

Table 2: The results of PSNR/SSIM for isotropic scale factor $\times 2$ with the gain between compared methods and the method of spline interpolation. One network 20L-ReCNN trained with 10 images of Kirby and one trained with NAMIC



(a) Original HR T1-weighted (b) LR T1-weighted image (c) HR T2-weighted reference

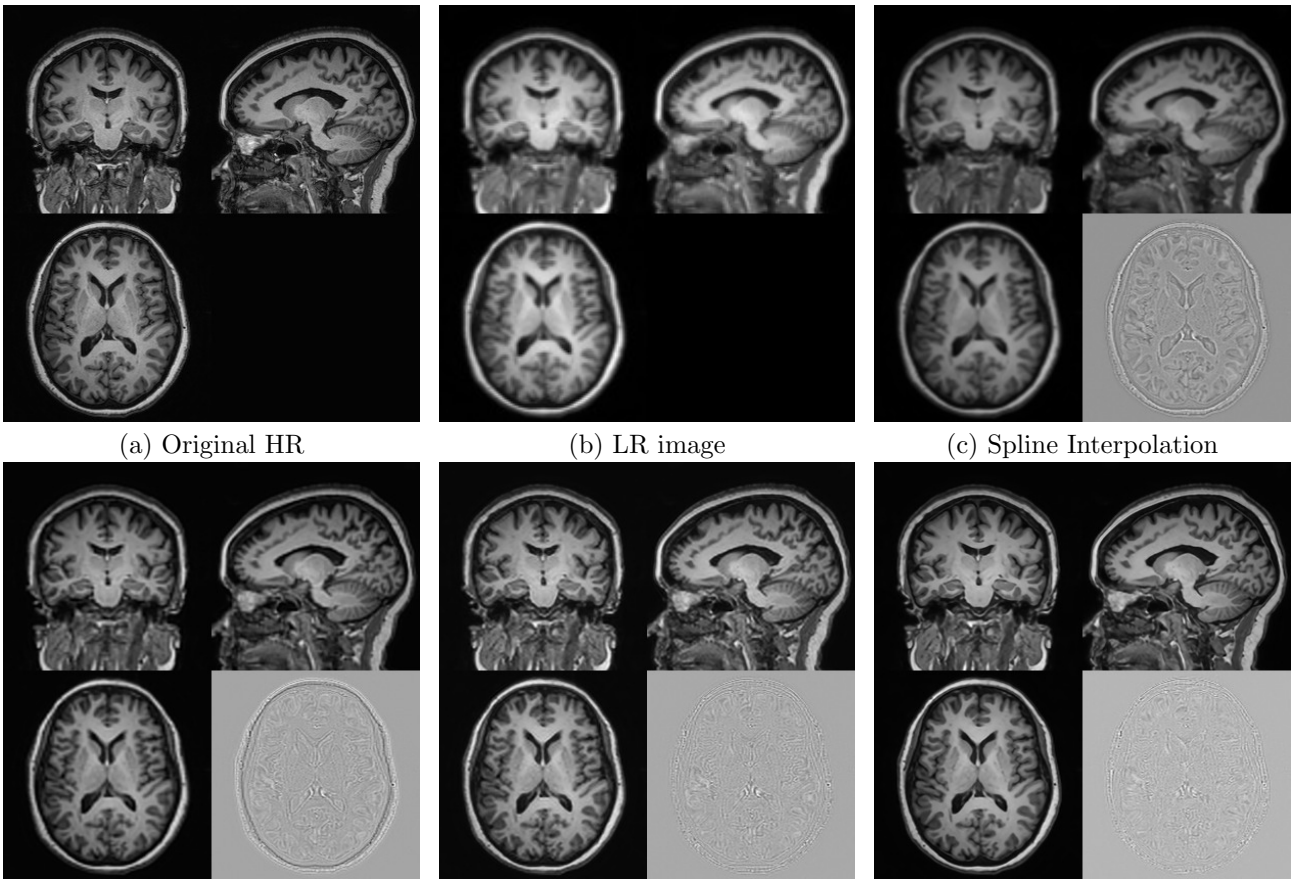
(d) Spline Interpolation (e) Monomodal 10L-ReCNN (g) Multimodal 10L-ReCNN

Figure 12: Illustration of the axial slices of monomodal and multimodal SR results (01018, pathological case) with isotropic voxel upsampling. LR T1-weighted image (b) with voxel size $2 \times 2 \times 2mm^3$ is upsampled to size $1 \times 1 \times 1mm^3$. Multimodal network 10L-ReCNN uses the HR T2-weighted reference (c) to upscale LR image. The different between ground truth image and reconstruction results are at the bottom. Their zoom version are at the right.

7. Discussion

This study investigates CNN-based models for 3D brain MR image SR. Based on a comprehensive experimental evaluation, we may first draw the following conclusions and recommendations regarding the setup to be considered. We highlight that eight complementary factors may drive the reconstruction performance of CNN-based models. The combination of 1) appropriate optimization with 2) weight initialization and 3) residual learning is a key to exploit deeper networks with a faster and effective convergence. The choice of an appropriate optimization method can lead to a improvement of (at least) $1dB$. In this study, it has appeared that Adam method [36] provides significantly better reconstruction results than other classic techniques such as SGD, and a faster convergence. Moreover, weights initialization is a very important step. Indeed, some approaches simply do not achieve convergence in the learning phase. This study has also shown that residual modeling for single image SR is a straightforward technique to improve the reconstruction performances ($+0.4dB$) without requiring major changes in the network architecture. Appropriate weight initialization methods described in [35, 42] allow us to build deeper residual-learning networks. From our point of view, these three aspects of SR algorithm are the first to require special attention for the implementation of a SR technique based on CNN.

Overall, we show that better performance can be achieved by learning a 4) deeper fully 3D



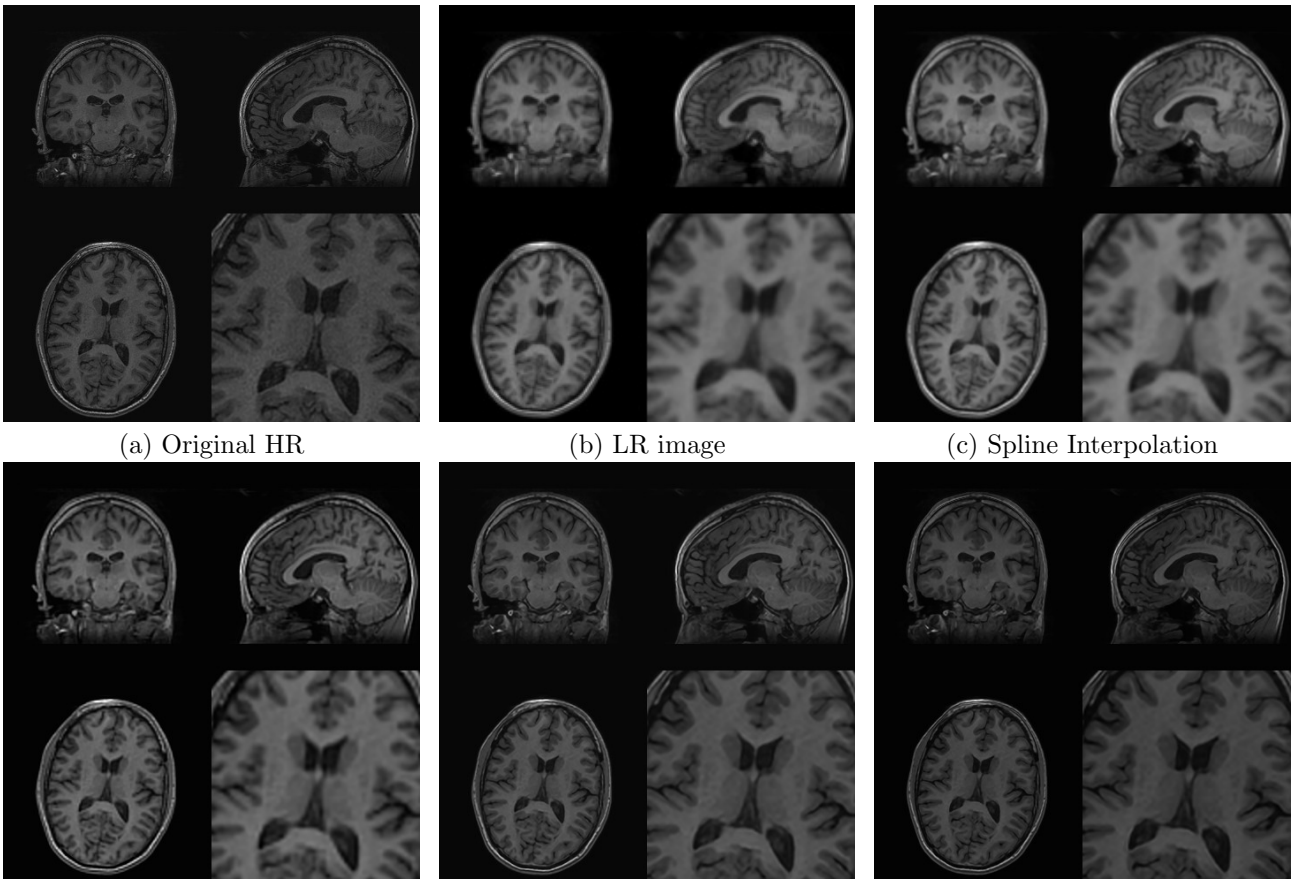
(a) Original HR (b) LR image (c) Spline Interpolation

(d) LRTV [6] (e) SRCNN3D [38] (g) 20L-ReCNN

Figure 13: Illustration of SR results (KKI2009-02-MPRAGE, non-pathological case) with isotropic voxel upsampling. LR data (b) with voxel size $2 \times 2 \times 2.4mm^3$ is upsampled to size $1 \times 1 \times 1.2mm^3$. The different between ground truth image and reconstruction results are in the right bottom corners. Both network SRCNN3D [38] and network 20L-ReCNN are trained with the 10 last images of Kirby.

convolution neural network, 5) exploring more filters and 6) increasing filter size. In addition, using 7) larger training patch size and 8) augmentation of training subject lead to significant increase in PSNR performance. The adjustment of these 5 elements provides a similar improvement (about $0.5dB$). Although it seems natural to implement the deepest possible network, this parameter is not always the key to obtaining a better estimate of a high-resolution image. Our study shows that, depending on the type of input data (monomodal or multimodal), network depth is not necessarily the main parameter leading to better image reconstruction. In addition, it is necessary to take into account the time of the learning phase as well as the maximum memory available in the GPU in order to choose the best architecture of the network. For instance, for the monomodal SR case, we suggest using 20-layer networks with 64 small filters with size of 3^3 regarding 10 training subjects of size 25^3 to achieve practicable results.

In CNN-based approaches, the up-scaling operation can be performed by using transposed convolution (so-called fractionally strided convolutional) layers in [32, 45] or sub-pixel



(a) Original HR (b) LR image (c) Spline Interpolation

(d) Low-Rank Total Variation (LRTV) [6] (e) 20L-ReCNN (trained with Kirby) (g) 20L-ReCNN (trained with NAMIC)

Figure 14: Illustration of SR results (01011-t1w, pathological case) with isotropic voxel upsampling. LR data (b) with voxel size $2 \times 2 \times 2mm^3$ is upsampled to size $1 \times 1 \times 1mm^3$. The zoom versions of the axial slices are in the right bottom corners.

layers [26]. However, the trained weights of these networks are appropriately applied for a specified scale factor. This is a limiting aspect of CNN-based SR for MR data since a fixed upscaling factor is not appropriate in this context. In this study, we have presented a multi-scale CNN-based SR method for single 3D brain MRI that is capable of learning multiple scale by training fully all isotropic scale factors due to an independent upsampling technique such as spline interpolation. Handling multiple scales is related to multi-task learning. The lack of flexibility of learned network architecture raises an open issue requiring further studies: how can we build a network that can deal with a set of observation models (*i.e.* multiple scales, arbitrary point spread functions, non uniform sampling, etc.)?

In this paper, we have proposed a multimodal method for brain MRI SR using CNNs where a HR reference image of the same subject can drive the reconstruction process of the LR image. By concatenating these HR and LR images, the reconstruction of the LR one can be enhanced by exploiting the multimodality feature of MR data. As shown in previous works [3, 4], the use of HR reference can lead to significant improvements of the

reconstruction process. However, unlike the monomodal setup, a deeper network does not necessarily lead to better performance. Experiments from our study show that future work is needed to understand the relationship between network depth and the quality of HR image estimation.

Moreover, we have experimentally investigated the performances of CNN for transfer learning (*"i.e. how a learned network can be used in another context"*). More specifically, our study illustrates how knowledge learned from one MR dataset is transferred to another one (different acquisition protocol and different scales). We have used KKI and NAMIC datasets for this purpose. Although a slight decrease in performance can be observed, CNN-based approach can still achieve better performance than existing methods. These results tend to demonstrate the potential applications of CNN-based techniques for MRI SR. Further investigations are required to fully assess the possibilities of transfer learning in medical imaging context, and the contributions of fine-tuning technique [22].

Finally, future research directions for CNN-based SR techniques could focus on other elements of network architecture. For instance, batch normalization (BN) step has been proposed by Ioffe *et al.* in [53]. The purpose of a BN layer is to normalize the data through the entire network, rather than just performing normalization once in the beginning. Although BN has been shown to improve classification accuracy and decrease training time [53], our first attempts to include BN layers into CNN for image SR have not lead to performance increase. Similarly, following previous work described in [54], we have examined the contribution of residual blocks and experiments have not shown performance improvement. Moreover, while the classical MSE-based loss attempts to recover the smooth component, perceptual losses [25, 27, 28] are proposed for natural image SR to better reconstruct fine details and edges. Thus, adding this type of layer (BN or residual block) or defining new loss functions may be beneficial for MRI SR and may provide new directions for research.

8. Acknowledgement

The research leading to these results has received funding from the ANR MAIA project, grant ANR-15-CE23-0009 of the French National Research Agency, INSERM and Institut Mines Télécom Atlantique (Chaire "Imagerie médicale en thérapie interventionnelle"). We also gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

[1] P. Milanfar, Super-resolution imaging, CRC press, 2010.
[2] J. V. Manjón, P. Coupé, A. Buades, V. Fonov, D. L. Collins, M. Robles, Non-local mri upsampling, Medical image analysis 14 (6) (2010) 784–792.
[3] F. Rousseau, A. D. N. Initiative, et al., A non-local approach for image super-resolution using intermodality priors, Medical image analysis 14 (4) (2010) 594–605.
[4] J. V. Manjón, P. Coupé, A. Buades, D. L. Collins, M. Robles, Mri superresolution using self-similarity and image priors, Journal of Biomedical Imaging 2010 (2010) 17.
[5] A. Rueda, N. Malpica, E. Romero, Single-image super-resolution of brain mr images using overcomplete dictionaries, Medical image analysis 17 (1) (2013) 113–132.

[6] F. Shi, J. Cheng, L. Wang, P. Yap, D. Shen, Lrtv: Mr image super-resolution with low-rank and total variation regularizations, IEEE transactions on medical imaging 34 (12) (2015) 2459–2466.
[7] B. Scherrer, A. Gholipour, S. K. Warfield, Super-resolution reconstruction to increase the spatial resolution of diffusion weighted images from orthogonal anisotropic acquisitions, Medical image analysis 16 (7) (2012) 1465–1476.
[8] D. H. Poot, B. Jeurissen, Y. Bastiaensen, J. Veraart, W. Van Hecke, P. M. Parizel, J. Sijbers, Super-resolution for multislice diffusion tensor imaging, Magnetic resonance in medicine 69 (1) (2013) 103–113.
[9] M. Fogtmann, S. Seshamani, C. Kroenke, X. Cheng, T. Chapman, J. Wilm, F. Rousseau, C. Studholme, A unified approach to diffusion direction sensitive slice registration and 3-d dti reconstruction from moving fetal brain anatomy, IEEE transactions on medical imaging 33 (2) (2014) 272–289.
[10] G. Steenkiste, B. Jeurissen, J. Veraart, A. J. Den Dekker, P. M. Parizel, D. H. Poot, J. Sijbers, Super-resolution reconstruction of diffusion parameters from diffusion-weighted images with different slice orientations, Magnetic resonance in medicine 75 (1) (2016) 181–195.
[11] S. Jain, D. M. Sima, F. S. Nezhad, G. Hangel, W. Bogner, S. Williams, S. Van Huffel, F. Maes, D. Smeets, Patch-based super-resolution of mr spectroscopic images: Application to multiple sclerosis, Frontiers in neuroscience 11.
[12] G. Ramos-Llordén, J. Arnold, G. Van Steenkiste, B. Jeurissen, F. Vanhevel, J. Van Audekerke, M. Verhoye, J. Sijbers, A unified maximum likelihood framework for simultaneous motion and $t\_\{1\}$ estimation in quantitative mr $t\_\{1\}$ mapping, IEEE transactions on medical imaging 36 (2) (2017) 433–446.
[13] G. Van Steenkiste, D. H. Poot, B. Jeurissen, A. J. Den Dekker, F. Vanhevel, P. M. Parizel, J. Sijbers, Super-resolution t1 estimation: Quantitative high resolution t1 mapping from a set of low resolution t1-weighted images with different slice orientations, Magnetic resonance in medicine 77 (5) (2017) 1818–1830.
[14] A. Gholipour, J. A. Estroff, S. K. Warfield, Robust super-resolution volume reconstruction from slice acquisitions: application to fetal brain mri, IEEE transactions on medical imaging 29 (10) (2010) 1739–1758.
[15] F. Rousseau, K. Kim, C. Studholme, M. Koob, J.-L. Dietemann, On super-resolution for fetal brain mri, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2010, pp. 355–362.
[16] B. Kainz, M. Steinberger, W. Wein, M. Kuklisova-Murgasova, C. Malamateniou, K. Keraudren, T. Torsney-Weir, M. Rutherford, P. Aljabar, J. V. Hajnal, et al., Fast volume reconstruction from motion corrupted stacks of 2d slices, IEEE transactions on medical imaging 34 (9) (2015) 1901–1913.
[17] Y. Jia, A. Gholipour, Z. He, S. K. Warfield, A new sparse representation framework for reconstruction of an isotropic high spatial resolution mr volume from orthogonal anisotropic resolution scans, IEEE Transactions on Medical Imaging 36 (5) (2017) 1182–1193.
[18] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.
[19] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2014.
[20] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[21] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, B. Glocker, Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation, Medical Image Analysis 36 (2017) 61–78.
[22] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, J. Liang, Convolutional neural networks for medical image analysis: full training or fine tuning?, IEEE transactions on medical imaging 35 (5) (2016) 1299–1312.
[23] C. Dong, C. C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE transactions on pattern analysis and machine intelligence 38 (2) (2016) 295–307.
[24] J. Kim, J. K. Lee, K. M. Lee, Accurate image super-resolution using very deep convolutional networks, in: in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[25] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, in: in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[26] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, Z. Wang, 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, pp. 1874–1883.
[27] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: European Conference on Computer Vision, Springer, 2016, pp. 694–711.
[28] H. Zhao, O. Gallo, I. Frosio, J. Kautz, Loss functions for neural networks for image processing, IEEE Transactions on Computational Imaging 2017 (TCI).
[29] J. Kim, J. K. Lee, K. M. Lee, Deeply-recursive convolutional network for image super-resolution, in: in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016.
[30] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, et al., Ntire 2017 challenge on single image super-resolution: Methods and results, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2017.
[31] R. Timofte, R. Rothe, L. Van Gool, Seven ways to improve example-based single image super resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1865–1873.
[32] O. Oktay, W. Bai, M. Lee, R. Guerrero, K. Kamnitsas, J. Caballero, A. de Marvao, S. Cook, D. O'Regan, D. Rueckert, Multi-input cardiac image super-resolution using convolutional neural networks, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 246–254.
[33] B. A. Landman, A. J. Huang, A. Gifford, D. S. Vikram, I. A. L. Lim, J. A. Farrell, J. A. Bogovic, J. Hua, M. Chen, S. Jarso, et al., Multi-parametric neuroimaging reproducibility: a 3-t resource study, Neuroimage 54 (4) (2011) 2854–2866.
[34] NAMIC: Brain Multimodality, http://hdl.handle.net/1926/1687.
[35] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1026–1034.
[36] D. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations, 2015.
[37] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, in: Proceedings of the 22nd ACM international conference on Multimedia, ACM, 2014, pp. 675–678.
[38] C.-H. Pham, A. Ducournau, R. Fablet, F. Rousseau, Brain mri super-resolution using deep 3d convolutional networks, in: Biomedical Imaging (ISBI 2017), 2017 IEEE 14th International Symposium on, IEEE, 2017, pp. 197–200.
[39] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.
[40] Y. Nesterov, A method of solving a convex programming problem with convergence rate o (1/k2), in: Soviet Mathematics Doklady, Vol. 27, 1983, pp. 372–376.
[41] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE transactions on neural networks 5 (2) (1994) 157–166.
[42] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks., in: Aistats, Vol. 9, 2010, pp. 249–256.
[43] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks., ICML (3) 28 (2013) 1310–1318.
[44] T. Tieleman, G. Hinton, Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude, COURSERA: Neural networks for machine learning 4 (2) (2012) 26–31.
[45] C. Dong, C. C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in:

European Conference on Computer Vision, Springer, 2016, pp. 391–407.
[46] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in: Curves and Surfaces, Springer, 2012, pp. 711–730.
[47] R. Timofte, V. De, L. Van Gool, Anchored neighborhood regression for fast example-based super-resolution, in: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE, 2013, pp. 1920–1927.
[48] R. Timofte, V. De Smet, L. Van Gool, A+: Adjusted anchored neighborhood regression for fast super-resolution, in: Computer Vision–ACCV 2014, Springer, 2014, pp. 111–126.
[49] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil in the details: Delving deep into convolutional nets, in: British Machine Vision Conference, 2014. arXiv:1405.3531.
[50] S. Schulter, C. Leistner, H. Bischof, Fast and accurate image upscaling with super-resolution forests, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3791–3799.
[51] M. Jenkinson, C. F. Beckmann, T. E. Behrens, M. W. Woolrich, S. M. Smith, Fsl, Neuroimage 62 (2) (2012) 782–790.
[52] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing 13 (4) (2004) 600–612.
[53] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, 2015, pp. 448–456.
[54] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: European Conference on Computer Vision, Springer, 2016, pp. 630–645.