Deep learning-based super-resolution of 3D Magnetic Resonance Images by Regularly Spaced Shifting

Karl Thurnhofer-Hemsi*, Ezequiel López-Rubio*, Enrique Domínguez*, Rafael Marcos Luque-Baena*

Department of Computer Languages and Computer Science
University of Málaga, Bulevar Louis Pasteur, 35, 29071 Málaga, Spain

Núria Roé-Vellvé*

Molecular Imaging Unit, Centro de Investigaciones Médico-Sanitarias
General Foundation of the University of Málaga, C/ Marqués de Beccaria, 3, 29010
Málaga, Spain

Abstract

The image acquisition process in the field of magnetic resonance imaging (MRI) does not always provide high resolution results that may be useful for a clinical analysis. Super-resolution (SR) techniques manage to increase the image resolution, being especially effective those based on examples that determine a correspondence between patterns of low resolution and high resolution. Deep learning neural networks have been applied in recent years to estimate this association with very competitive results. In this work, the starting point is a convolutional neuronal network to which a regularly spaced shifting mechanism over the input image is applied, with the aim of substantially improving the quality of the resulting image. This hybrid proposal has been compared with several SR techniques using the peak signal-to-noise ratio, structural similarity index and Bhattacharyya coefficient metrics. The results obtained on different MR images show a considerable improvement both in the restored image and in the residual image without an excessive increase in computing time.

Keywords: Magnetic Resonance Imaging, super resolution, convolutional

---

*Corresponding author
Email address: karlkhader@lcc.uma.es (Karl Thurnhofer-Hemsi)

neural networks, supervised learning

1. Introduction

Super-resolution (SR) is the process whose aim is to generate a high-resolution image (HR) from a single low-resolution image (LR) by means of post-processing techniques. The objective is to improve the image quality taking into account that the solution is not unique [1]. Besides its a classic applications in the computer vision field, SR is also of great interest for medical imaging. Improvements in SR can be of critical importance, due to the need of obtaining more detailed and realistic images when they are used for diagnostic purposes.

Despite advances in acquisition technology such as Computerized Tomography (CT), Positron Emission Tomography (PET), Magnetic Resonance Imaging (MRI) or combined modalities (e.g. SPECT/CT), factors like the inherent noise in the devices or blurring, are always present in each of these techniques, which sets limits to the resolution and quality of the obtained images. MRI is no exception to that, with a resolution of the order of millimeters, due to the magnetic properties of the tissues and varying as a function of the way the signal and noise are sampled and filtered. Hardware limitations, high signal-to-noise ratios (SNR), practical limits to the acquisition time and patient movement also contribute to limiting image resolution. All of this calls for the application of SR techniques to the resulting images.

According to this need, in recent years there has been a growing interest in improving SR methods for MRI, linked to the development of machine learning algorithms. Interpolation and spline-based methods are the traditional ways to increase image resolution [2, 3]. However, these approaches estimate new intermediate points assuming the homogeneity of a region, causing a blurring perception in the image, especially in areas with edges or contrast changes. More successful proposals, such as the example-based methods, have become popular as super-resolution techniques [4, 5]. Among them, some exploit the internal similarities of the image [6, 7] and others learn mapping patterns between LR

and HR images from external datasets [8, 9, 10, 11].

Recently, a super-resolution convolutional neural network (SRCNN) [12], which can be considered as an example-based method, has obtained great attention because of its ability to learn an end-to-end mapping between LR and HR images. In this way, it is not required to learn dictionaries or manifolds to model the high-resolution space. Other recent works also incorporate neural architectures based on deep learning to perform super-resolution tasks [13, 14, 15, 16, 14].

Convolutional neuronal networks (CNNs), inspired by the animal visual cortex, have been one of the first deep learning architectures that have demonstrated excellent performance in any field associated with the image and video processing. Thanks to the power of the new graphical acceleration devices (GPU), CNNs have been successfully applied both detection and recognition of objects[17, 18], classification of images or within recommender systems. Hundreds of articles have been published in several areas [19, 20], including the field of medical image analysis [21, 22], where the CNNs popularity is growing and its use is progressively expanding.

In the image medical analysis field, a 3D densely connected super-resolution network for brain MRI data improvement was recently presented [23]. Its particularity is the use of dense layers instead of convolutional layers on its architecture. The expected high complexity of the method is mitigated because it is a light-weight model with weight sharing and a very reduced number of parameters. Another recent work presents a combination of a densely connected network with a generative adversarial network (GAN) which seems to achieve promising results [24]. Additionally, other interesting works which relate brain MRI and deep learning architectures can be found in the literature [25, 26, 27].

In this work, a 3D convolutional neural network previously defined [28] (SR-CNN3D), is combined with a regularly space shifting mechanism to enhance the quality of the high-resolution image. Thus, the combination of the network output after testing several regularly shifted input images, will be significantly improved the HR image quality. This technique has been satisfactorily applied

3

to diverse brain MRI image datasets and has been compared with several state-of-the-art SR algorithms. Additionally, another previously published proposal from our research group [29] has been included in the comparison.

The remaining of the paper is structured as follows. Section 2 describe the proposed model, where in subsection 2.1 the convolutional neural network is commented and in subsection 2.2 the regularly spaced shifting mechanism is explained. The explanation about how the results are obtained, the different competitive methods and the image datasets analyzed, are outlined in Section 3 whereas the quantitative and qualitative results are displayed in Section 4. To sum up, Section 5 discusses the results previously reported and Section 6 exposes the conclusions and future works of our approach.

2. Theory

In this section deep learning super resolution is carried out on regularly spaced shifted versions of the input image. This technique is proposed to enhance the quality of MR images. In Subsection 2.1 the base SRCNN method is outlined. Then in Subsection 2.2 the regular shifting algorithm to produce the final estimation of the HR image from the LR one is detailed.

2.1. Convolutional neural network

The standard SR reconstruction is carried out through a CNN following these two steps:

1. Given a LR image $\mathbf{X}$, a spline interpolation I is performed in order to obtain a HR image $\mathbf{Z} = I(\mathbf{X})$.

2. A convolutional neural network is applied to restore the image.

The CNN is composed by three blocks of layers. The first two consist in a convolutional layer followed by a Rectified Linear Unit (ReLU). The last layer only performs a convolution, without any ReLU after the filter responses. If we call g_1, g_2, g_3 for each of these operations, the net computes a HR image

$$g = g_3 \circ g_2 \circ g_1 \tag{1}$$

4

and the restoration based on a residual learning technique and computed minimizing the Euclidean loss

$$\tilde{f} = arg\,min_g \sum ||\mathbf{Y} - g(\mathbf{Z})||^2 \qquad (2)$$

where $\mathbf{Y}$ is the HR image corresponding to $\mathbf{X}$.

The first convolutional layer applies 64 filters of size $9 \times 9 \times 9$, 32 of size
$1 \times 1 \times 1$ in the second, and one filter of size $5 \times 5 \times 5$ in the last layer. Several overlapping patches per image are extracted from a set of HR reference images in order to have enough samples to train the network, and feature maps are computed. For each patch, a down-sampling and up-sampling is applied and a relationship is established to learn an end-to-end mapping between LR and HR images.

This neural network was used as a base for the proposed model described in the next section. It is called SRCNN3D and specific details about its implementation can be found in [28].

2.2. Regularly spaced shifting model

Our proposed regular shifting method is presented in this subsection. It consists in combining the outputs produced by the SRCNN network for shifted versions of the original input image, where the shift vectors are varied according to a regular rectangular pattern. The motivation for this approach is that each shifted input image produces a slightly different output when processed by the SRCNN network, so that a number of variants of the output are obtained as different shift vectors are considered. After that, the obtained output images can be averaged in order to produce an ensemble output. This averaging operation smooths out a certain amount of noise, thereby increasing the quality of the final combined output image. As compared to our previous random shifting approach [29], the regular shifting approach proposed here ensures that the set of considered shift vectors is balanced within the set of all possible shift vectors. The regular shifting approach is based on numerical integration theory, while our previous approach relies on statistical approximation theory.
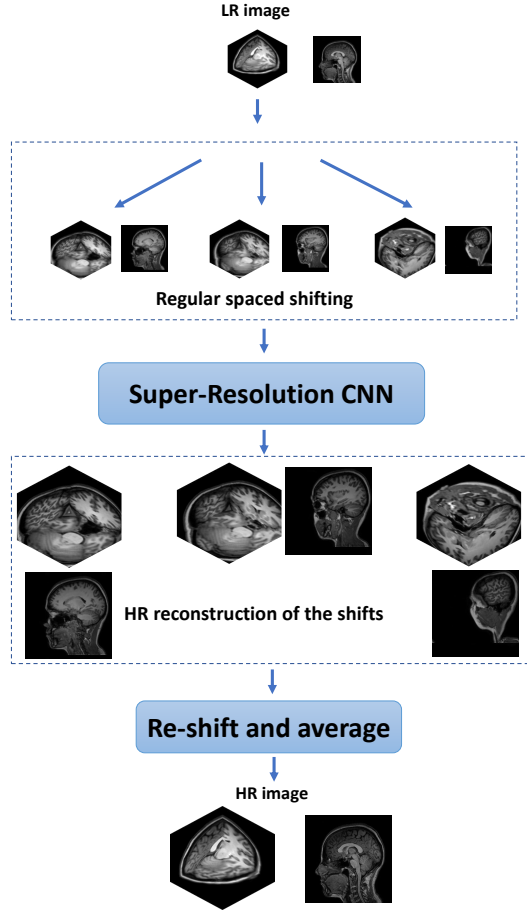
5

Figure 1: Scheme of the proposed algorithm.

Given an LR image $\mathbf{X}$, the SRCNN model learns an approximation $\tilde{f}(\mathbf{X})$ of the underlying HR image $f(\mathbf{X})$:

$$f(\mathbf{X}) = \tilde{f}(\mathbf{X}) + \tilde{\epsilon}(\mathbf{X}) \tag{3}$$

where $\tilde{\epsilon}(\mathbf{X})$ is the approximation error. Let us note:

$$\mathcal{V} = [0, Z) \times [0, Z) \times [0, Z) \tag{4}$$

where Z is a window size (in voxels). Shifted versions LR image can be con-

sidered as inputs to the SRCNN, where $\mathbf{a} \in \mathcal{V}$ is a shift vector. Therefore a different approximation $\bar{f}_{\mathbf{a}}(\mathbf{X})$ of the underlying HR image f is obtained for each shift vector $\mathbf{a}$:

$$\bar{f}_{\mathbf{a}}(\mathbf{X}) = \tilde{f}(\mathbf{X} \ll \mathbf{a}) \gg \lambda \mathbf{a} \tag{5}$$

$$\bar{\epsilon}_{\mathbf{a}}(\mathbf{X}) = \tilde{\epsilon}(\mathbf{X} \ll \mathbf{a}) \gg \lambda \mathbf{a} \tag{6}$$

$$f(\mathbf{X}) = \left(\tilde{f}(\mathbf{X} \ll \mathbf{a}) + \tilde{\epsilon}(\mathbf{X} \ll \mathbf{a}) \right) \gg \lambda \mathbf{a} =$$

$$\left(\tilde{f}(\mathbf{X} \ll \mathbf{a}) \gg \lambda \mathbf{a} \right) + \left(\tilde{\epsilon}(\mathbf{X} \ll \mathbf{a}) \gg \lambda \mathbf{a} \right) =$$

$$\bar{f}_{\mathbf{a}}(\mathbf{X}) + \bar{\epsilon}_{\mathbf{a}}(\mathbf{X}) \tag{7}$$

where $\ll$ and $\gg$ stand for the image left shift and right shift operators, respectively, and λ is the super-resolution zoom factor.

Given an LR image $\mathbf{X}$, let $\varphi_{\mathbf{X}}$ be a constant function which takes the shift vector $\mathbf{a}$ as argument:

$$\varphi_{\mathbf{X}}(\mathbf{a}) = f(\mathbf{X}) \tag{8}$$

Here it must be highlighted that the domain of $\varphi_{\mathbf{X}}$ is the set of valid values for $\mathbf{a}$, i.e. $\mathbf{a} \in \mathcal{V}$. Moreover, from (8) we can write:

$$f(\mathbf{X}) = \frac{1}{Z^3} \int_{\mathcal{V}} \varphi_{\mathbf{X}}(\mathbf{a}) \, d\mathbf{a} \tag{9}$$

For each $\mathbf{a}$, from (7) and (8) we get an approximation for $\varphi_{\mathbf{X}}(\mathbf{a})$:

$$\varphi_{\mathbf{X}}(\mathbf{a}) = \bar{f}_{\mathbf{a}}(\mathbf{X}) + \bar{\epsilon}_{\mathbf{a}}(\mathbf{X}) \tag{10}$$

Now the standard approximation (3) of the underlying function f by the SRCNN model can be rewritten as follows:

$$f\left(\mathbf{X}\right) = \tilde{f}\left(\mathbf{X}\right) + \tilde{\epsilon}\left(\mathbf{X}\right) = \varphi_{\mathbf{X}}\left(\mathbf{0}\right) = \bar{f}_{\mathbf{0}}\left(\mathbf{X}\right) + \bar{\epsilon}_{\mathbf{0}}\left(\mathbf{X}\right) \tag{11}$$

This means that approximating $f\left(\mathbf{X}\right)$ by $\tilde{f}\left(\mathbf{X}\right)$, which is what the original SRCNN does, amounts to a zeroth order approximation of $\varphi_{\mathbf{X}}\left(\mathbf{0}\right)$ obtained by taking a single noisy value $\bar{f}_{\mathbf{0}}\left(\mathbf{X}\right)$. We propose to estimate $f\left(\mathbf{X}\right)$ by approximating the integral in (9). In turn, the integral is estimated by averaging approximations of $\varphi_{\mathbf{X}}\left(\mathbf{a}\right)$ given by (10) and taken at regularly spaced values of $\mathbf{a}$:

$$\hat{f}\left(\mathbf{X}\right) = \frac{1}{M}\sum_{j=1}^{M} \bar{f}_{\mathbf{a}_j}\left(\mathbf{X}\right) \tag{12}$$

where the $\mathbf{a}_j$ are taken at regularly spaced values from $\mathcal{V}$.

Now, it is reasonable to think that $\hat{f}\left(\mathbf{X}\right)$ is a better approximation to $f\left(\mathbf{X}\right)$ than the original approximation $\bar{f}_{\mathbf{0}}\left(\mathbf{X}\right)$. On one hand, $\hat{f}\left(\mathbf{X}\right)$ is built as an approximation of an integral from many regularly spaced noisy observations, where the noise function $\bar{\epsilon}_{\mathbf{a}}\left(\mathbf{X}\right)$ in (10) is expected to be smooth and have zero mean with respect to $\mathbf{a}$, i.e. $E_{\mathbf{a}}\left[\bar{\epsilon}_{\mathbf{a}}\left(\mathbf{X}\right)\right] = \mathbf{0}$, because $\bar{f}_{\mathbf{a}}\left(\mathbf{X}\right)$ is the output of the SRCNN as the input is shifted by $\mathbf{a}$. Therefore, the noise might be partially averaged out by the integral. On the other hand, $\bar{f}_{\mathbf{0}}\left(\mathbf{X}\right)$ is built as a zeroth order approximation from a single noisy observation. Hence we propose to use $\hat{f}\left(\mathbf{X}\right)$ as an approximator to $f\left(\mathbf{X}\right)$. In practice the $\mathbf{a}_j$ are constrained to be integer vectors, so that fractional shifts are not necessary, since fractional shifts would pose a difficult problem themselves.

The proposed algorithm reads as follows:

1. Given M, compute the following set of vectors:

$$\mathcal{S}_M = \left\{\mathbf{a}_m\right\}_{m \in \{1, \ldots, M\}} \tag{13}$$

$$\mathbf{a}_m = \left(a^{(m)}, a^{(m)}, a^{(m)}\right) \in \mathbb{N}^3 \tag{14}$$

8

where $a^{(m)}$ is obtained as:

$$a^{(m)} = (m-1)R \tag{15}$$

with R being a tunable parameter of the algorithm which indicates the spacing between consecutive shifts.

2. For an input LR image $\mathbf{X}$, compute M circularly image shifts:

$$\{\mathbf{X} \ll \mathbf{a}_m, \mathbf{a}_m \in \mathcal{S}_M\} \tag{16}$$

3. Apply the CNN to obtain a set of HR images:

$$\{\tilde{f}(\mathbf{X} \ll \mathbf{a}_m), \mathbf{a}_m \in \mathcal{S}_M\} \tag{17}$$

4. Recompose the images considering the super-resolution zoom factor to obtain $\{\bar{f}_{\mathbf{a}_m}(\mathbf{X}), \mathbf{a}_m \in \mathcal{S}_M\}$, and then compute the final restored HR image $\bar{f}(\mathbf{X})$ following Eq. (12).

A schematic depiction of the operation of our algorithm is shown in Figure 1. It must be noted that (15) corresponds to the following window size:

$$Z = MR \tag{18}$$

3. Material and Methods

Description of the experiments we have carried out, and also datasets and algorithms are reported in this section. The LR image generation procedure, the software and hardware that we have used, and the selected performance metrics for comparison between methods are specified in Subsection 3.1. Then, the set of MR images are described in 3.2. The set of tuned parameters of our proposal and the tuning experiments are explained in Subsection 3.3 and the descriptions of the competing algorithms in Subsection 3.4.

160   Firstly, we need to describe how the LR images are generated since we only have available the HR ones. Input data for each method were obtained following this procedure:

1. HR images were cropped in relation to the zoom factor to be applied, in order to avoid fractional values.

165   2. A three-dimensional Gaussian filter was applied, with standard deviation equal to 1.

3. Finally, the LR image is generated applying a cubic interpolation through the *imresize3* function of Matlab and its default parameters.

As the convolutional neural network used for the super-resolution step we

170 selected the SRCNN3D method. It has demonstrated a considerable effectiveness compared with other state-of-art deep learning methods. This is a CNN method where a huge set of reference patches are extracted from each training HR image and compared with the same set of downsampled and upsampled patches in order to update the network weights.

175   An online available pre-trained model was used for the experiments. This network was trained over 470000 iterations, with 10 images from Kirby dataset [30] for training the CNN (images 33-42), using momentum of 0.9, learning rate of 0.0001 and batch size of 256. Stochastic gradient descent was used for model optimization.

180   All the experiments described in this work were carried out on a 64-bit Personal Computer with an eight-core Intel i7 3.60GHz CPU, NVidia Titan X GPU, 32 GB RAM and standard hardware, using Matlab R2017b. The deep learning method network has been developed using the Caffe package [11] on a Python framework.

185   Three different quality measures were used to evaluate and compare the proposed method:

- *Peak Signal-to-Noise Ratio (PSNR)*, measured in (decibels) dB, which is commonly used in medical image processing (higher is better).
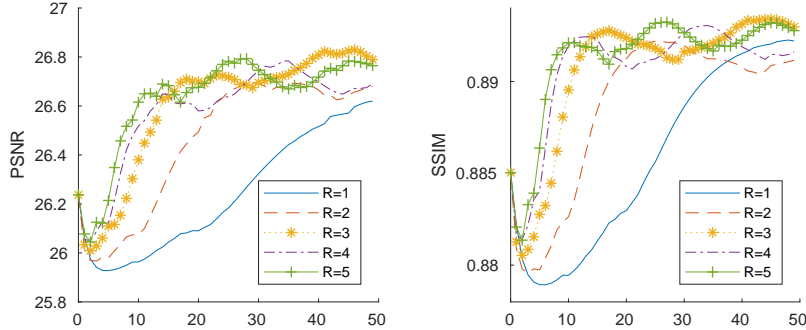
10

Figure 2: Evolution of the PSNR and SSIM metrics (higher is better) when varying the number of shifts employed for the reconstruction. Five different spacing values are plotted. Image 1 of the Kirby 21 dataset, image 80 of disc1 of the OASIS dataset and T1 noiseless image of Brainweb were used for the analysis.

- *Structural Similarity index (SSIM)* [31], which focuses on structural similarities between images (higher is better):

$$SSIM(x,y) = \frac{(2\mu_x\mu_y)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{19}$$

where μ_x and μ_y are the mean value of images x and y, σ_x and σ_y are the standard deviation of images x and y, σ_{xy} is the covariance of x and y, $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$ (default values were used: $L = 1$ is the dynamic range, $k_1 = 0.01$ and $k_2 = 0.03$).

- *Bhattacharyya coefficient (BC)* [32], which measures the closeness of the two discrete pixel probability distributions P and $\hat{P}$ corresponding to the ground truth (GT) and modeled images with values in the range $[0, 255]$:

$$BC = \sum_{j=0}^{255} P(j)\hat{P}(j) \tag{20}$$

where $BC \in [0, 1]$ and higher is better.

Also, CPU time was measured for each method. In the case of the methods based on the repetition of CNN testings as many times as shifted images were generated, we measured the processing time sequentially considering only one

11

Table 1: Considered parameter values for our method

Parameter	Value
Spacing R	3
Number of shifts M	47

GPU. Nevertheless, they could be parallelized in different GPUs to improve the total execution time.

From a qualitative point of view, we compared the competing methods using both restored and residual HR images. This residual image r was computed as the difference between the original HR image h and the super-resolved one s:

$$r = h - s \tag{21}$$

A darker residual image imply a better performance, as the differences between the original HR image and the restored must be close to zero. Its values were subtracted to the positive constant 0.5 and color maps were adjusted for a better visualization and discrimination between methods.

3.2. Datasets

Seven different images obtained from different datasets were considered in order to evaluate the results of the SR algorithms.

- Three T1-weighted MRI images of the Kirby 21 (images 5, 10 and 11) [30]. These data were acquired using a 3-T MR scanner with a $1.0 \times 1.0 \times 1.2 \, mm^3$ voxel resolution over an field-of-view (FOV) of $240 \times 204 \times 256 \, mm$ acquired in the sagittal plane. They are named as KKI2009XXMPRAGE.

- Two T1 images of the OASIS dataset (images 1 and 2 of the cross-sectional data) [33] . Data were acquired on a 1.5-T Vision scanner with a $1.0 \times 1.0 \times 1.25 \, mm^3$ voxel resolution over an FOV of $256 \times 256 \, mm$. They are named as OAS1_000X_MR1_mpr1_anon.

- One image of the IBSR public dataset [34]. It is named IBSR_07, it has image size $256 \times 256 \times 128$, with $1.5 \times 1.0 \times 1.0\,mm^3$ voxel resolution.

- A T1-weighted image was acquired at the Medical Research Center of the University of Málaga (CIMES)[1] using a 3-T MR scanner with a $0.93 \times 0.93 \times 1.0\,mm^3$ voxel resolution over an FOV of $256 \times 256\,mm$.

All these images are different from the CNN training dataset. Image 5 of Kirby 21 was only analyzed with zoom factor 3.

3.3. Parameter selection

Our model depends on the tuning of two parameters: the spacing of the shifting R and the number of shifts M. For this necessary prior task, we employed three images different from those selected for the final experiments:

- Image 1 of the Kirby 21 dataset: KKI2009-01-MPRAGE.

- Image 80 of the OASIS dataset: OAS1_0080_MR1_mpr-1_anon.

- A T1-weighted image from the Brainweb[2] simulated database (slice thickness $1\,mm$, $0\,\%$ noise level and $RF = 0$).

The default values of the SRCNN3D model were used, so no tuning was needed. We computed the PSNR and SSIM measures for all these images and we plotted the mean values for each spacing value $R \in \{1, 2, 3, 4, 5\}$ varying the number of shifts M between 0 and 50, where $M = 0$ corresponds to the base method SRCNN3D. The evolution of these metrics displayed in Figure 2. Despite the fact that the frequency of the oscillations in the performance measures raises as the number of shifts increases, a stabilization is observed above 30 shifts. The performance values for $R = 3$ seem to be higher than those for other values of R. According to these results, a parameter selection has been carried out. The chosen parameters are reported in Table 1.

---

[1]`www.cimes.es`

[2]`http://mouldy.bic.mni.mcgill.ca/brainweb/`

3.4. Competitors

Seven SR algorithms were used to compare our proposal[3], which was named as *SRCNN3D+RegSS*:

- *Spline*: bicubic spline interpolation as implemented in Matlab (Mathworks Inc.).

- *NLMU* (non-local means upsampling) [6]: recover some of high frequency information by using a data-adaptive patch-based reconstruction in combination with a subsampling coherence constraint.

- *LRTV* (low-rank total variation) [35]: low-rank regularization and total variation techniques were used to integrate both local and global informa-
tion for image reconstruction.

- *SRCNN3D* [28]: three-dimensional convolutional neural network trained with patches of HR brain images. Instead of learning the mapping directly from the LR space to the HR one, it estimates a mapping from the LR space to the missing high-frequency components.

- *SRReCNN3D-10L* [36]: A generalization of SRCNN3D where 10 blocks of Convolution+ReLU are used in the network architecture. A pre-trained model available online was used. Only works for zoom factor 2, so experiments with zoom 3 omit this competing method.

- *SRReCNN3D-20L* [36]: A generalization of SRCNN3D where 20 blocks of
Convolution+ReLU are used in the network architecture. A pre-trained model available online was used. Only works for zoom factors 2 and 3.

- *VDSR* [37]: very deep convolutional network with 20 layers based on residual learning with extremely high learning rates. Contextual information is exploited by cascading small filters.

---

[3]`https://bitbucket.org/icaiuma/srcnn3d_regss/`

₂₆₅ • *EDSR* [38]: compact deep network which removes unnecessary modules from conventional ResNet architecture, also based on residual scaling techniques.

• *SRCNN3D+RndS* [29]: A previous approach based on the use of a random shifting technique.

₂₇₀ The first three have been implemented in Matlab. The convolutional network called *SRCNN3D*, which is used by our proposal, and also *SRReCNN3D-10L* and *SRReCNN3D-20L* have been developed using Caffe package [11] on a Python framework. *VDSR* was created and integrated in Matlab. There are versions in Torch and Pytorch of the *EDSR* network, so we used the last one[4]. ₂₇₅ As both *VDSR* and *EDSR* networks were designed for 2D images, we applied a straightforward strategy combining results from coronal, axial, and sagittal views by computing the average of this three image reconstructions [23, 28].

4. Experimental Results

A quantitative evaluation of the performance of the ten different methods ₂₈₀ using zoom factor 2 is summarized in Figure 3. The mean and the standard deviation for each method are shown with horizontal error bars, for the six tested images (all except image 5 of Kirby21), using a super-resolution zoom factor of 2. Traditional methods, i.e. *Spline*, *NLMU* and *LRTV*, perform clearly worse than deep learning methods, both in mean and standard deviation values. We ₂₈₅ can see that the *LRTV* method outperforms the other two in BC, but PSNR and SSIM values still remain similar. A second block of methods can be distinguish from the others, composed by the *SRCNN3D* algorithm and its generalizations *SRReCNN3D-10L* and *SRReCNN3D-20L*. An improvement of all the metrics can be appreciated with respect to the conventional methods, reaching 29 dB ₂₉₀ of PSNR and a SSIM value of 0.94. On the other hand, *VDSR* and *EDSR* networks do not reach good results with respect *SRCNN3D*.
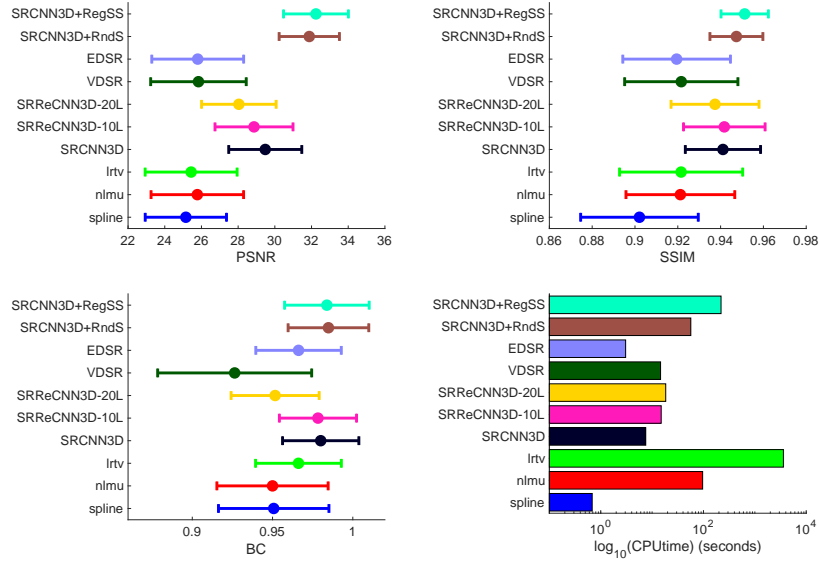
---

[4]https://github.com/thstkdgus35/EDSR-PyTorch

Figure 3: Comparison of the PSNR, SSIM, BC (higher is better) and CPU time for the ten methods. Mean and standard deviation of the results for all the test images except image 5 of Kirby21 are displayed, using $\lambda = 2$.

A second amelioration of all the metrics can be seen when comparing the shifting models with the isolated CNN-based methods. PSNR increases from 29 dB up close to 32 dB, and SSIM and BC are also improved, which indicates that the local brain structures are restored in a better way. With the proposed regular spaced shifting model *SRCNN3D+RegSS* PSNR values of 33 dB are achieved. Actually, better values of SSIM are reported reducing the standard errors. BC values remains flat.

Figure 3 also shows the mean processing time (in a logarithmic scale) required for the execution of each method. The *Spline* method is the fastest one, followed by *EDSR*. However, LR images are poorly restored as seen before. *NLMU* and *LRTV* need long times to process a single image and are also very dependent on its dimensions. The higher the number of voxels, the more time they use to generate the HR image. Our proposal and *SRCNN3D+RndS* also require time, which depends on the number of shifts. CNN methods processes
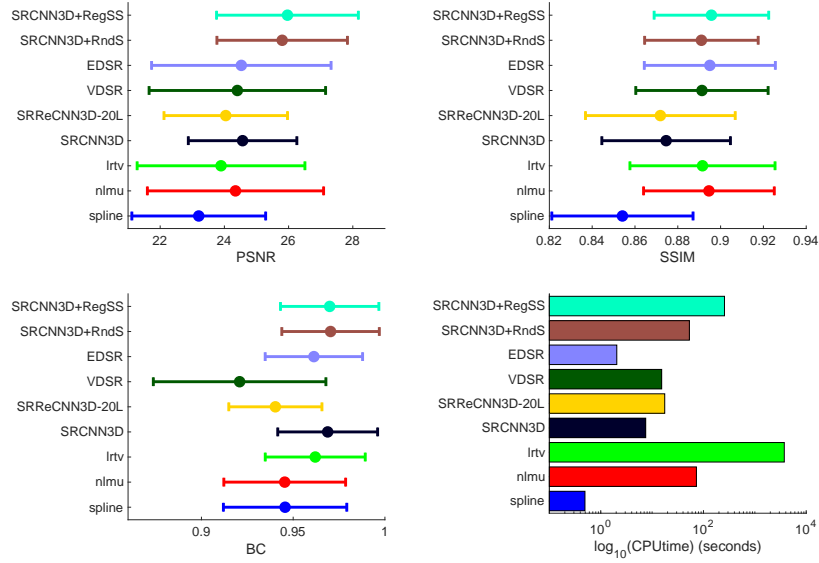
16

Figure 4: Comparison of the PSNR, SSIM, BC (higher is better) and CPU time for nine methods. Mean and standard deviation of the results for all the test images except image 5 of Kirby21 are displayed, using $\lambda = 3$.

an image between 8 and 12 seconds. *SRCNN3D+RegSS* takes around 3-4 minutes, which is less than *LRTV* but longer than the 53 seconds required by the random shifting method.

In Figure 4 are shown the outcomes using $\lambda = 3$ for the same set of images. *SRReCNN3D-10L* method is not displayed because the network was not trained for this zoom factor. In this case, *LRTV* and *NLMU* methods perform better, reaching good values of SSIM. Actually, they outperform the CNN methods *SRCNN3D* and *SRReCNN3D-20L* also for some BC values. Unlike the analysis of zoom 2, the very deep networks *VDSR* and *EDSR* performs better, improving sometimes our proposal for the SSIM measure, but with large variances in the outcomes for PSNR and BC. Analysing carefully the results, these discrepancies are originated due to their bad performance with certain images (of Kirby 21 specially). The rest of outcomes are quite suited. The models based on a shifting procedure still remain as the best ones. The highest values are obtained by the

17

(a) Original HR image (b) LR image (c) Spline (d) NLMU

(e) LRTV (f) SRCNN3D (g) SRReCNN3D-10L (h) SRReCNN3D-20L

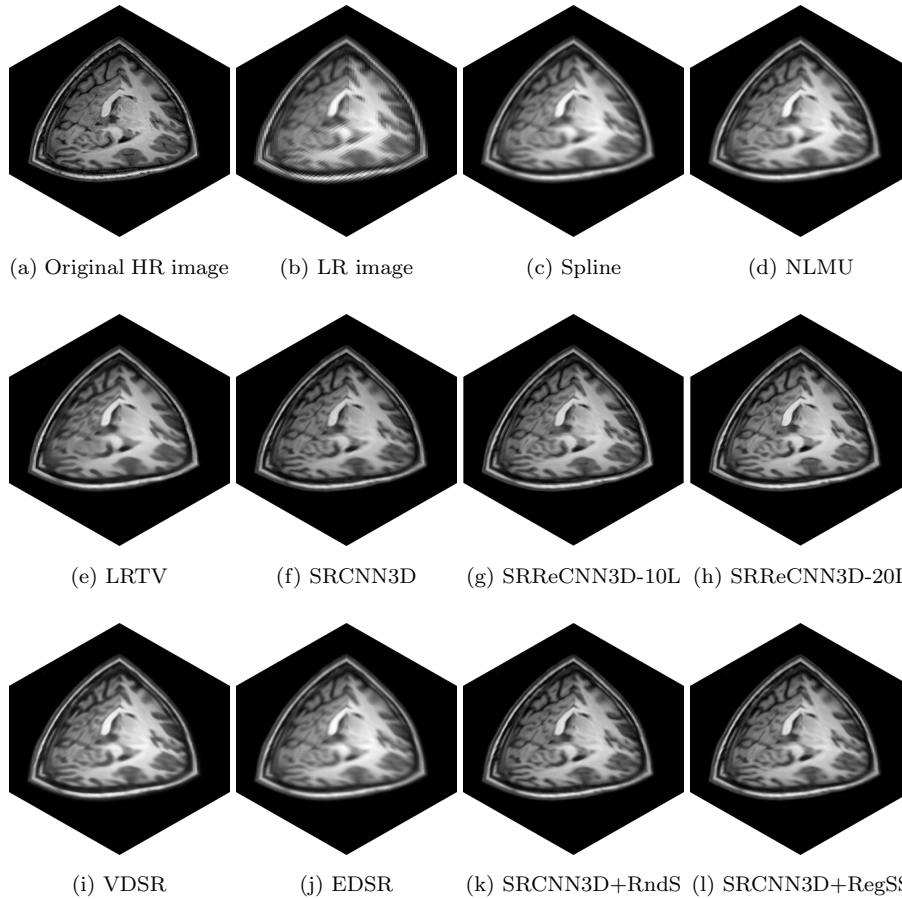(i) VDSR (j) EDSR (k) SRCNN3D+RndS (l) SRCNN3D+RegSS

Figure 5: Qualitative results for KKI2009-11-MPRAGE T1-weighted image for each method, applied with zoom factor 2. Three-dimensional images are shown, where the XY plane corresponds to a slice of the axial view, XZ to a slice of the sagittal view and YZ to a slice of the coronal view.

proposed *SRCNN3D+RegSS*. Finally, the ranking of the processing times is very similar to the previous ones.

In order to assess the methods from a qualitative point of view, in Figure 5 is depicted a three-dimensional representation of the Kirby 21 (image 11) restoration. Middle slices of each plane are shown. The smoothest results are generated by both *Spline* and *NLMU*, which do not correspond with the desired ideal HR image. This may indicate that they depend heavily on the degree of

(a) Spline   (b) NLMU

(c) LRTV   (d) SRCNN3D   (e) SRReCNN3D-10L   (f) SRReCNN3D-20L

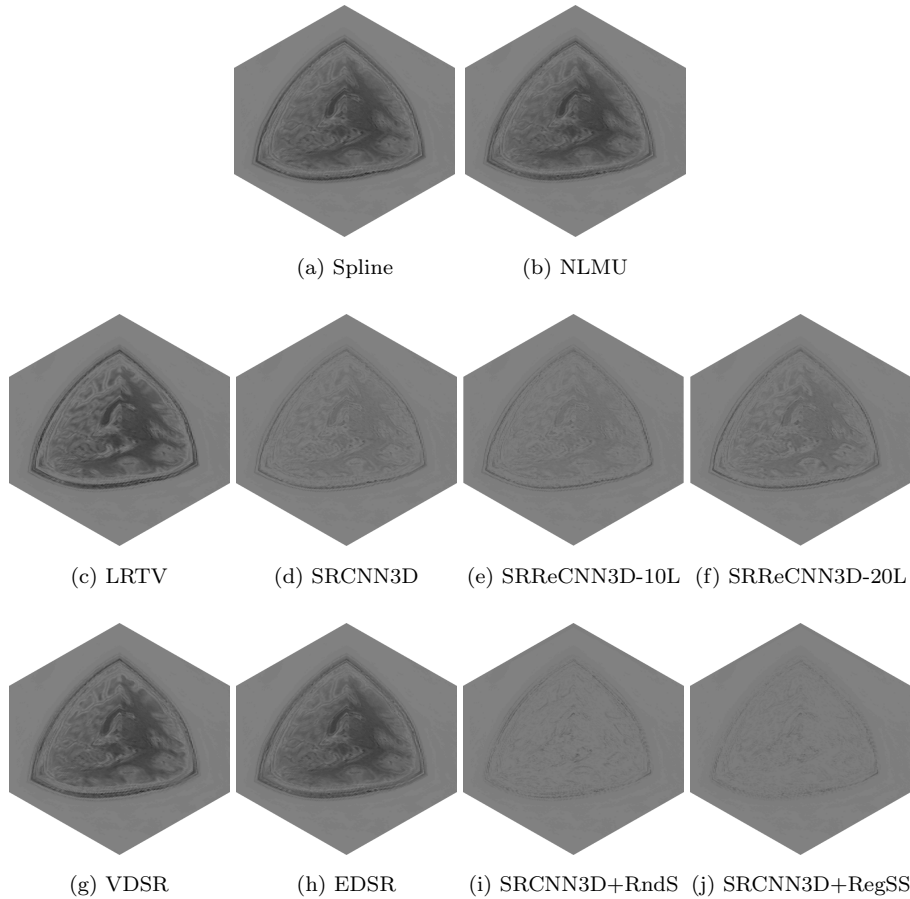(g) VDSR   (h) EDSR   (i) SRCNN3D+RndS   (j) SRCNN3D+RegSS

Figure 6: Residual images for KKI2009-11-MPRAGE T1-weighted image for each method, applied with zoom factor 2. Three-dimensional images are shown, where the XY plane corresponds to a slice of the axial view, XZ to a slice of the sagittal view and YZ to a slice of the coronal view.

smoothness of the input LR image. *LRTV* is also quite different. The CNN methods yield the best restorations, with the exceptions of *EDSR*, which looks over smoothed. In order to evaluate the performance of each method and make a better differentiation, residual images were also computed and the results are shown in Figure 6. Here we can distinguish the best outcomes watching those images that display fewer brain structures. Raw deep learning methods yield good results, but there are still visible dark parts. However, the residual images

19

of the shifting methods are quite gray, and actually in *SRCNN3D+RegSS* the skull is almost indistinguishable.

An example of the result obtained using zoom factor 3 from both the quantitative and qualitative points of view are shown in Figure 7, Figure 8 and Table 2, using the image 5 of the Kirby21 dataset. As before, *Spline* and *NLMU* are clearly worse seeing the restored image. Nevertheless, we have to focus on the residuals to be able to distinguish among the others. Here is obvious that *LRTV* still does not behave well. Comparing CNN methods, there are not great differences, but *SRCNN3D+RndS* and *SRCNN3D+RegSS* seem to yield a better outcome where the brain distortion is not severe. Table 2 allows us to corroborate our assumptions. Deep learning techniques perform better that traditional algorithms. The adaptation of *VDSR* and *EDSR* from 2D to 3D is not enough to outperform the other CNN methods. The inclusion of the shifting model increases the quality of the restored HR image. In some cases the differences are small, but our method yields better results in PSNR, SSIM and BC, which demonstrate that our proposal is effective, with an acceptable processing time.

The real T1-weighted image from CIMES was processed in Figure 9 and Figure 9. Here we can observe the notable performance of our proposal. If we focus on the residual images, we can extract two different conclusions. *SRCNN3D+RndS* and *SRCNN3D+RegSS* yield the most uniform gray residual, which means that voxel intensities of the restored image are very similar with respect to the original HR image. In addition to this, if we compare internal parts of the brain and even the neck (which is clinically irrelevant but useful for comparison purposes), the gray intensities remain stable. CNN and traditional methods have greater non uniform values of gray that could distort the original brain structures and voxel intensities.

A sagittal slice of the super-resolved OAS1_0001_MR1_mpr-1_anon image is shown in Figure 11. Here the differences are almost imperceptible except in the case of *Spline NLMU*, where there are visible smoother brain parts. Not too many differences can be appreciated among the other methods apart from the gray scale intensities shown in the residual images of Figure 12. in some regions

20

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

(d) NLMU   (e) LRTV   (f) SRCNN3D   (g) SRReCNN3D-20L

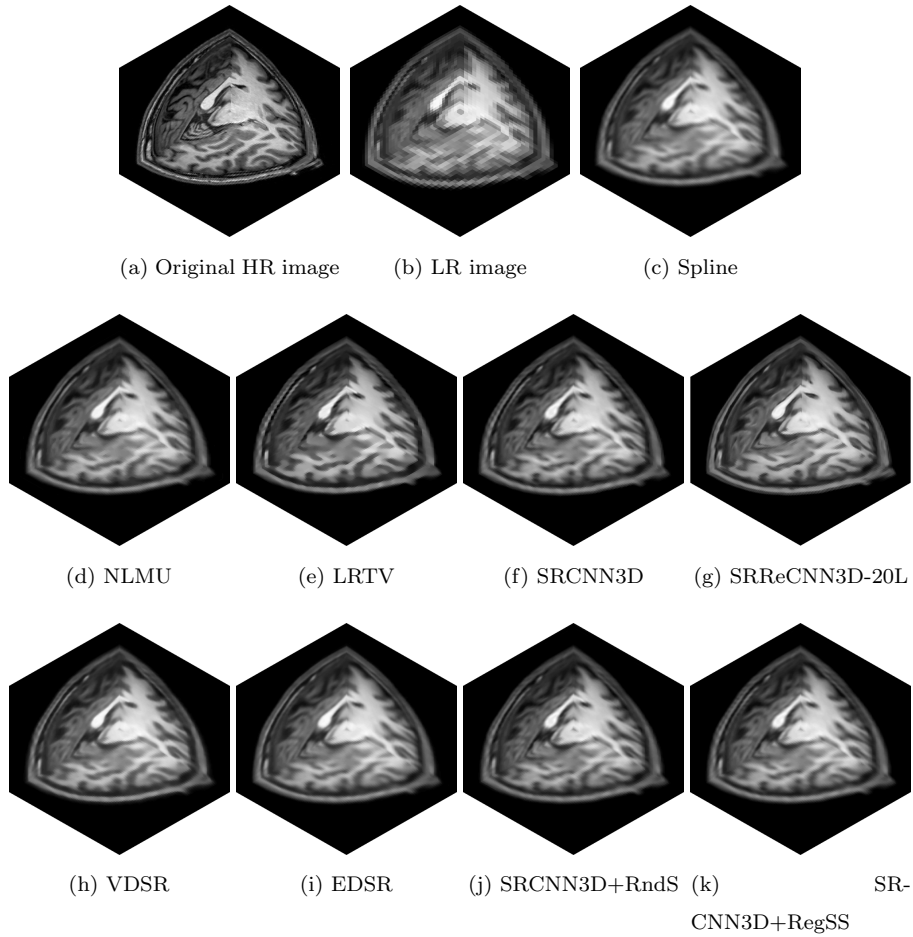(h) VDSR   (i) EDSR   (j) SRCNN3D+RndS   (k)   SR-
CNN3D+RegSS

Figure 7: Qualitative results for KKI2009-05-MPRAGE T1-weighted image for each method, applied with zoom factor 3. Three-dimensional images are shown, where the XY plane corresponds to a slice of the axial view, XZ to a slice of the sagittal view and YZ to a slice of the coronal view.

365   of the brain. We can assure that CNN methods, including *VDSR* and *EDSR*, outperform the traditional methods. *SRCNN3D+RegSS* achieves to recover and does not remove small black spots that are distorted by the others with respect to the HR reference image.
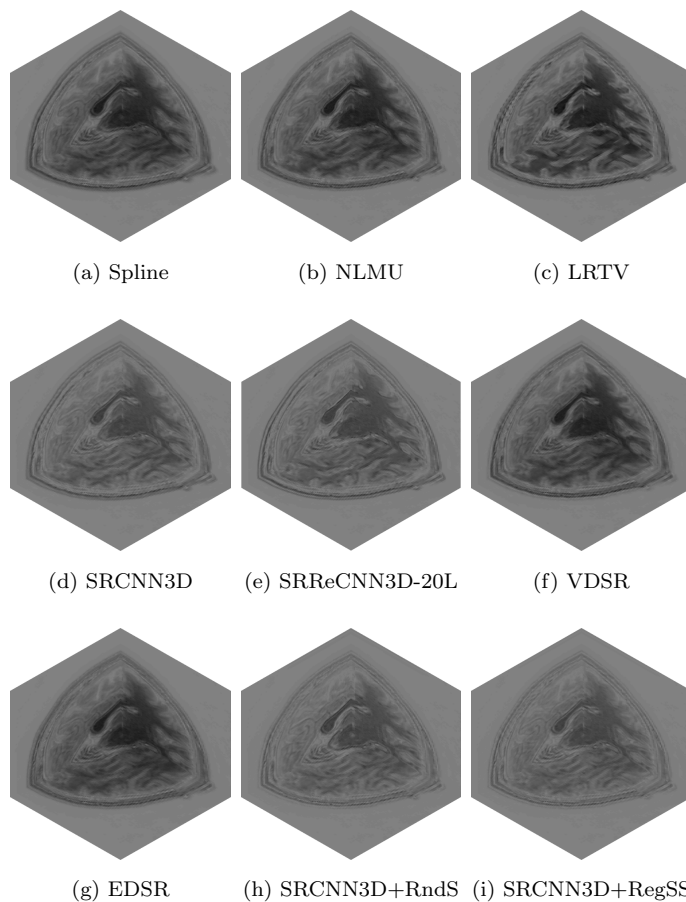
21

Figure 8: Residual images for KKI2009-05-MPRAGE T1-weighted image for each method, applied with zoom factor 3. Three-dimensional images are shown, where the XY plane corresponds to a slice of the axial view, XZ to a slice of the sagittal view and YZ to a slice of the coronal view.

5. Discussion

370   It is important to recall that a representative metric for this kind of images is usually SSIM, because it focuses on the morphology of the brain, and not so much on the intensity values. In this sense, our proposal obtains the best SSIM values. Moreover, the high value (very close to 1) and small standard deviation we have obtained in BC are also remarkable, so we can ensure that the restored

Table 2: Results obtained for each method for the tested images with zoom factor 3 (higher is better for PSNR, SSIM and BC).

KKI2009-05-MPRAGE	PSNR	SSIM	BC	CPU time (sec.)
Spline	20.1656	0.8118	0.9174	**0.7436**
NLMU	20.3356	0.8396	0.9167	78.8499
LRTV	20.1216	0.8324	0.9335	4308.1375
SRCNN3D	23.6425	0.8542	0.9599	7.7902
SRReCNN3D-20L	23.4603	0.8509	0.9432	12.9448
VDSR	20.7754	0.8412	0.9080	16.6636
EDSR	20.6961	0.8459	0.9305	2.0827
SRCNN3D+RndS	24.1791	0.8703	0.9624	53.1471
SRCNN3D+RegSS	**24.3661**	**0.8771**	**0.9630**	216.7438

375 image is close to the original HR image in terms of the distribution of intensity values.

About the required CPU time, our method is not the fastest one. Nevertheless, it should be considered that a single GPU was employed when using it. The use of more GPUs simultaneously may decrease almost linearly the total time. 380 If there are N GPUs and we compute M shifts, with $M \leq N$, it is expected a very similar performance compared to *SRCNN3D*, since the required time for the shift operations is almost insignificant.

According to the Figure 3 and Table 2, it is remarkable the performance difference between the proposed *SRCNN3D+RegSS* and the *SRCNN3D-20L*, 385 since the latter model has a larger number of layers on its architecture. As it is commented in Section 4.3 of [12], not always "deeper is better", since it depends on the amount of data available for training and the number of epochs. In addition to this, the reported results show that the proposed shifting technique *SRCNN3D+RegSS* outperforms the compared state of the art methods and 390 the previous proposal *SRCNN3D+RndS*. This statement is also based on the qualitative results obtained from the residual images, which can be observed in

23

(a) Original HR image   (b) LR image   (c) Spline   (d) NLMU

(e) LRTV   (f) SRCNN3D   (g) SRReCNN3D-10L (h) SRReCNN3D-20L

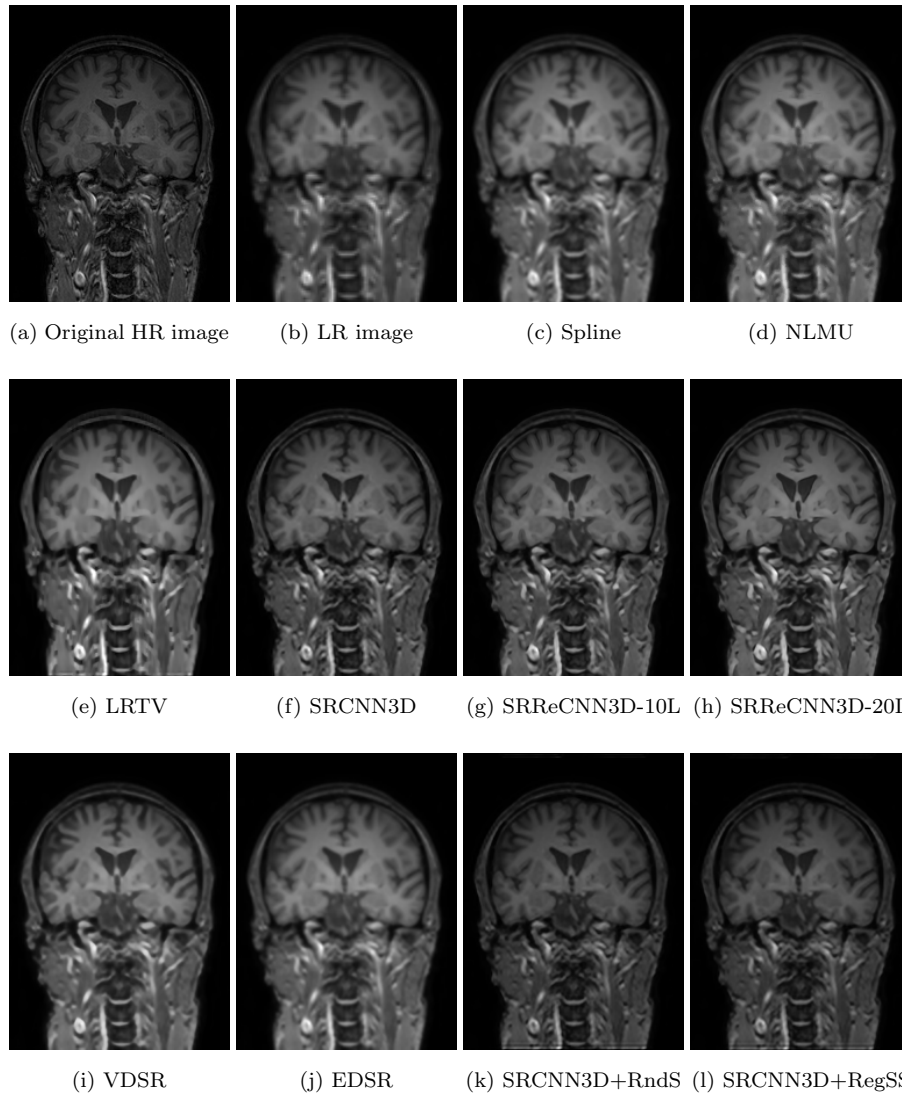(i) VDSR   (j) EDSR   (k) SRCNN3D+RndS (l) SRCNN3D+RegSS

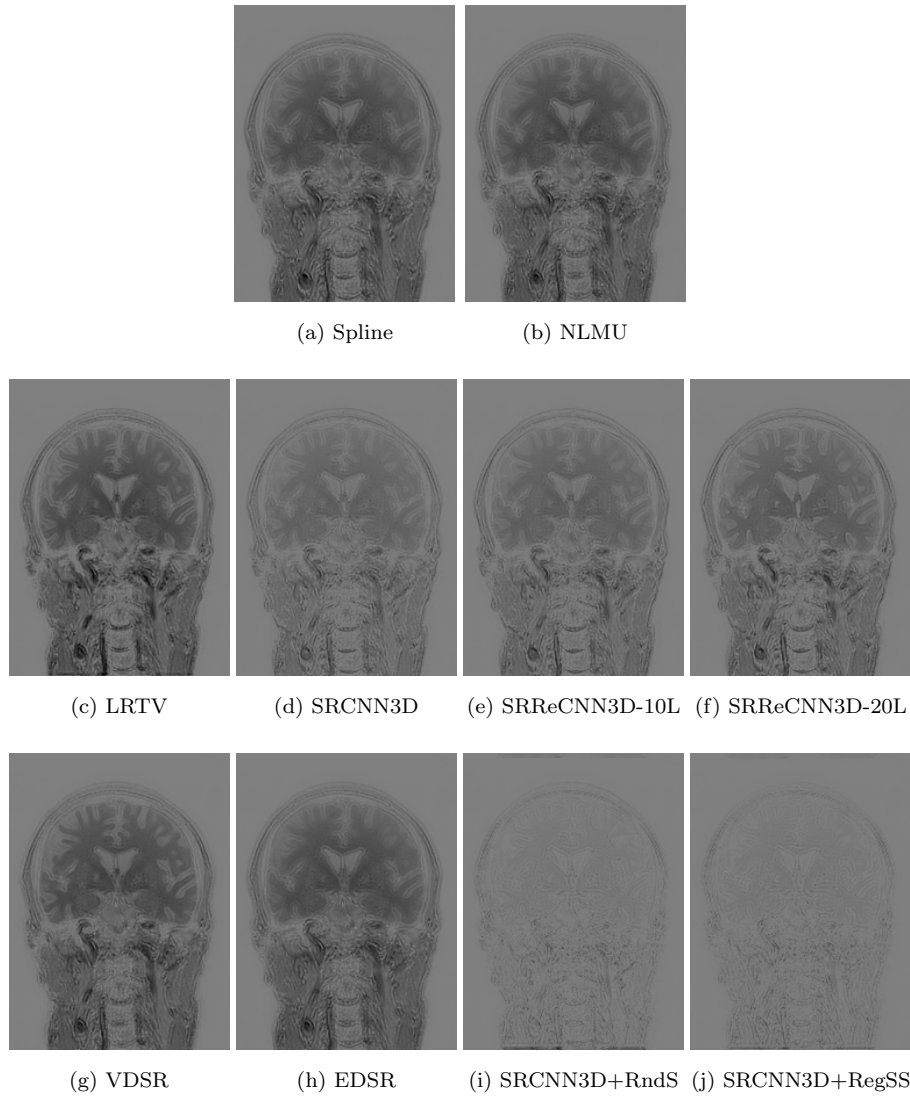Figure 9: Qualitative results for the T1-weighted image from CIMES for each method, applied with zoom factor 2. Coronal view is shown. Second and third row display the image reconstructed by each algorithm.

Figures 6, 8, 10 and 12. The more homogeneous and gray the image, the better the result and consequently the quality of the method. It is possible to observe that our method improves the remaining ones over the datasets studied.

(a) Spline   (b) NLMU

(c) LRTV   (d) SRCNN3D   (e) SRReCNN3D-10L   (f) SRReCNN3D-20L

(g) VDSR   (h) EDSR   (i) SRCNN3D+RndS   (j) SRCNN3D+RegSS

Figure 10: Qualitative results for the T1-weighted image from CIMES for each method, applied with zoom factor 2. Coronal view is shown. Residual images between the reconstructed and the original HR image are displayed.

395   As described in Subsection 2.2, the advantage over the results obtained by a single application of the *SRCNN3D* comes from an image filtering which is carried out on the three dimensional space of possible shifts **a**. The properties

(a) Original HR image   (b) LR image   (c) Spline   (d) NLMU

(e) LRTV   (f) SRCNN3D   (g) SRReCNN3D-10L   (h) SRReCNN3D-20L

(i) VDSR   (j) EDSR   (k) SRCNN3D+RndS   (l) SRCNN3D+RegSS
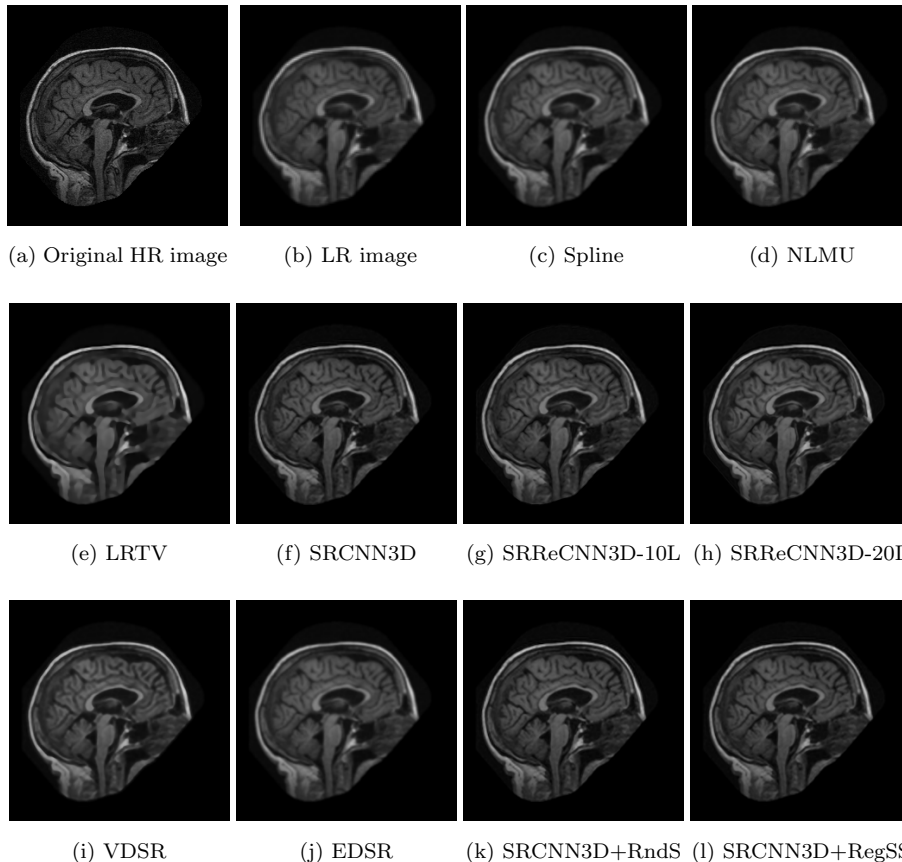
Figure 11: Qualitative results for OAS1_0001_MR1_mpr1_anon image for each method, applied with zoom factor 2. Reconstructed sagittal slices by each algorithm are shown.

of this transformed space are worth being investigated in future works, since it is possible that careful tuning of the filtering in the transformed space could lead to even better results.

5.1. Performance of adapted 2D models

In the present work we include comparisons with two recognised super-resolution methods for natural images, such as *VDSR* and *EDSR*. These very deep networks outperform most of the state-of-art SR algorithms, although they were designed for bidimensional natural images. These fact translates on two
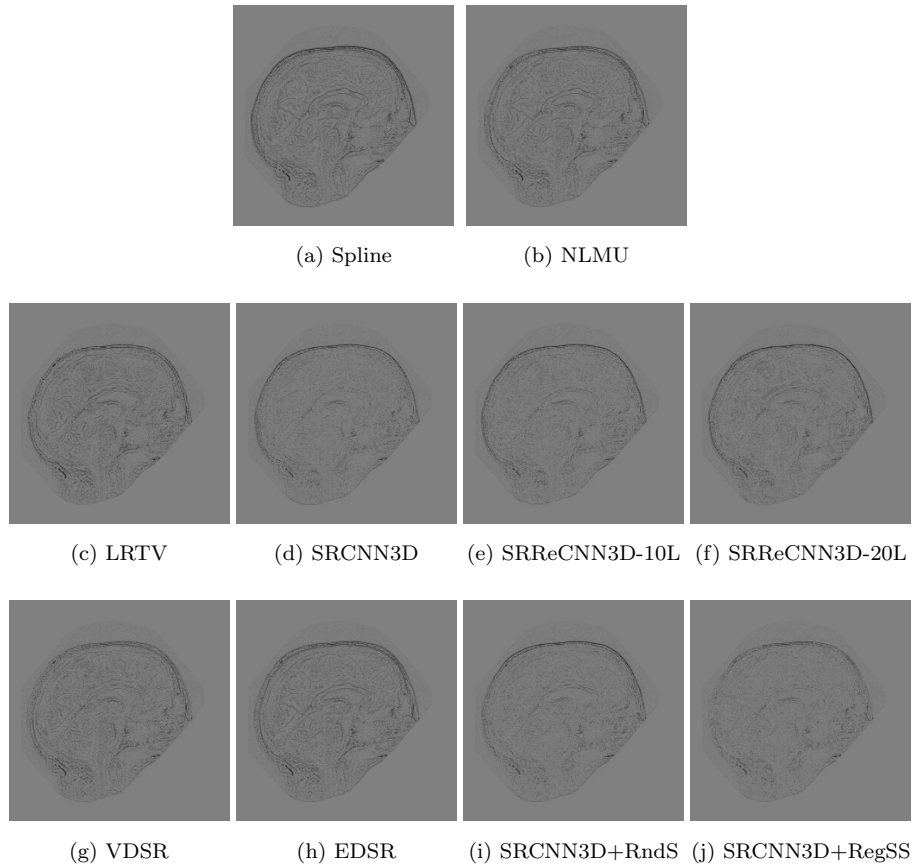
(a) Spline   (b) NLMU

(c) LRTV   (d) SRCNN3D   (e) SRReCNN3D-10L   (f) SRReCNN3D-20L

(g) VDSR   (h) EDSR   (i) SRCNN3D+RndS   (j) SRCNN3D+RegSS

Figure 12: Qualitative results for OAS1_0001_MR1_mpr1_anon image for each method, applied with zoom factor 2. Residual images between the reconstructed and the original HR image are shown (sagittal slice).

important limitations: the range of values and the dimensionality.

Both networks work on a fixed, 8-bit quantized range of values, that is, $[0, 255]$. In particular, in the case of *VDSR*, it performs the training and the restoration of the images on the luminance channel of the $YCbCr$ color space, which is nothing more than a linear transformation of the RGB color space. In the case of magnetic resonance images, the intensity values are not quantized nor bounded, so to apply these kind of methods on these images, a normalization is needed to transform the unbounded intensity values into the 256 possible
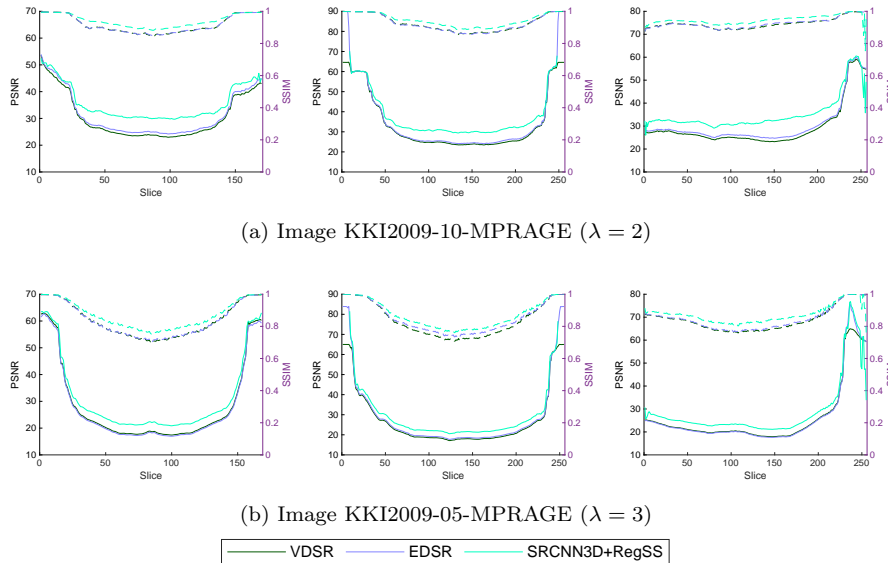
(a) Image KKI2009-10-MPRAGE ($\lambda = 2$)

(b) Image KKI2009-05-MPRAGE ($\lambda = 3$)

| VDSR | EDSR | SRCNN3D+RegSS |

Figure 13: Analysis of the performance of *VDSR* and *EDSR* with respect to *SR-CNN3D+RegSS* for both scale factors 2 and 3. PSNR (left y-axis) is shown with a solid line and SSIM (right y-axis) with a dashed line (higher is better). Quality measures for all slices are depicted. From right to left: traversing saggital, coronal and axial axes.

values of the *RGB* space. This transformation may produce a significant loss of

415 information that the networks cannot recover. Figures 5-10 shows the excessive

smoothness of the SR images obtained by these two methods, which could have

been produced by this fact.

On the other hand, we need to deal with the application of this networks

on 3D images. Depending on the network, different procedures are applied. In

420 the case of *VDSR*, we followed the steps described in both Matlab[5] and source

code[6] examples. We carried out an initial upsampling to reach the desired

image size and after that we apply the network restoration on each slice. *EDSR*

has a different methodology. The input of this network is the LR image and

---

[5]https://www.mathworks.com/help/images/single-image-super-resolution-using-deep-learning.html

[6]https://cv.snu.ac.kr/research/VDSR/

Table 3: Comparison of the image reconstruction of *VDSR* and *EDSR* with respect to *SR-CNN3D+RegSS* analyzing the PSNR and SSIM (higher is better) of 3D SR image reconstructed on each axis.

Image	KKI2009-10-MPRAGE ($\lambda = 2$)			KKI2009-05-MPRAGE ($\lambda = 3$)		
Method	X-axis	Y-axis	Z-axis	X-axis	Y-axis	Z-axis
VDSR	26.4033/0.9313	26.7989/0.9345	26.1664/0.9318	20.9162/0.8396	20.8662/0.8395	21.2268/0.8433
		27.4724/0.9390			20.7754/0.8412	
EDSR	27.5119/0.9311	27.4537/0.9340	27.3726/0.9340	20.4648/0.8386	21.4813/0.8543	20.9599/0.8477
		27.3822/0.9335			20.6961/0.8459	
SRCNN3D+RegSS		32.7381/0.9527			24.3661/0.8771	

the output is the HR image, so after the application on each slice, the sliced dimension must be upsampled again. That is, if the LR image has size $l \times m \times n$ and the HR has size $l' \times m' \times n'$, if we apply *EDSR* on each $m \times n$ slice, then we obtain an anisotropic $l \times m' \times n'$ image. Then we apply bicubic interpolation to restore the first dimension, i.e. to upsample from $l \times m' \times n'$ to $l' \times m' \times n'$.

In order to assess the correct application of these methods and study the effect of the dimensionality, in Figure 13 the PSNR and SSIM measures obtained on each slice along the three dimensions during the restoration procedure are shown. We depicted the performances for two different images and zoom factors, comparing our proposal with *VDSR* and *EDSR* methods. The performance of the three methods is very similar or even better for *EDSR* on boundary slices, that are those one dominated by zero voxel values. However, as we enter into the central part of the image, both PSNR and SSIM values of the 2D methods are worse than our 3D approach. In Figure 13b the differences are a bit smaller, and they are in concordance with Table 2. *VDSR* achieves a better PSNR in two of the three axes, while *EDSR* does the same in SSIM. We summarise the quality of the 3D images obtained along each axis in Table 3. Also, in the central columns the final PSNR and SSIM of the final reconstructed image using the straightforward strategy explained in Section 3 are shown.

Taking into account these analyses, the bad performance of the recognised methods *VDSR* and *EDSR* is probably due to the fact that medical images carry specific 3D structural information, and the reconstruction by a 2D model

29

is not a natural way to keep knowledge from the image. As an example, a brain sulcus may be positioned near to the adjacent slice and also go slightly through it. In this case, this small oscillation might be difficult to distinguish from noise when a 2D model is applied, potentially resulting in noise enhancement in the SR image.

6. Conclusion

In this paper, a method for magnetic resonance image super-resolution is presented. It is based on the combination of two different methodologies. Low-resolution images are processed through a convolutional neural network to perform an image restoration in order to obtain a high-resolution image. The quality of the restored images is increased by applying a regular shifting model to the input images and then recomposing them into a consensus. A variety of images of different datasets were used to evaluate the efficiency of the algorithm, obtaining successful results.

According to the used quality measures (PSNR, SSIM and BC), the proposed method has achieved better results comparing to the state of the art methods. Moreover, SRCNN3D+RegSS has performed the most similar voxel intensities to the original HR images, achieving the most uniform gray residuals, specially in images from CIMES.

Quantitative results show that SRCNN3D+RegSS overcomes other state-of-art methods, enhancing really high values of BC and measures as PSNR or SSIM, which indicates that the brain structures are not distorted. Experimental results show our proposal restores the MR images qualitatively well for different zoom factors, avoiding over-smoothing.

Future lines of research include the development of further tuned filtering methods carried out on the three dimensional shift space which has been proposed in this work.

Acknowledgment

References

[1] E. López-Rubio, Superresolution from a single noisy image by the median filter transform, SIAM Journal on Imaging Sciences 9 (1) (2016) 82–115.

[2] T. M. Lehmann, C. Gonner, K. Spitzer, Survey: interpolation methods in medical image processing, IEEE Transactions on Medical Imaging 18 (11) (1999) 1049–1075.

[3] P. Thevenaz, T. Blu, M. Unser, Interpolation revisited [medical images application], IEEE Transactions on Medical Imaging 19 (7) (2000) 739–758.

[4] A. J. Shah, S. B. Gupta, Image super resolution-a survey, in: 2012 1st

International Conference on Emerging Technology Trends in Electronics, Communication Networking, 2012, pp. 1–6.

[5] C. S. Balure, M. R. Kini, A survey – super resolution techniques for multiple, single, and stereo images, in: 2014 Fifth International Symposium on Electronic System Design, 2014, pp. 215–216.

[6] 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.

[7] H. Zheng, X. Qu, Z. Bai, Y. Liu, D. Guo, J. Dong, X. Peng, Z. Chen, Multi-contrast brain magnetic resonance image super-resolution using the local weight similarity, BMC Medical Imaging 17 (1) (2017) 1–13.

[8] 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.

[9] D. H. Trinh, M. Luong, F. Dibos, J. M. Rocchisani, C. D. Pham, T. Q. Nguyen, Novel example-based method for super-resolution and denoising of medical images, IEEE Transactions on Image Processing 23 (4) (2014) 1882–1895.

[10] D. Zhang, J. He, Y. Zhao, M. Du, MR image super-resolution reconstruction using sparse representation, nonlocal similarity and sparse derivative prior, Computers in Biology and Medicine 58 (2015) 130–145.

[11] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, arXiv preprint arXiv:1408.5093.

[12] 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. doi:10.1109/TPAMI.2015.2439281.

[13] Y. Liang, J. Wang, S. Zhou, Y. Gong, N. Zheng, Incorporating image priors with deep convolutional neural networks for image super-resolution, Neurocomputing 194 (2016) 340–347. `doi:10.1016/j.neucom.2016.02.046`.

[14] Y. Li, J. Hu, X. Zhao, W. Xie, J. Li, Hyperspectral image super-resolution using deep convolutional neural network, Neurocomputing 266 (2017) 29–41. `doi:10.1016/j.neucom.2017.05.024`.

[15] G. Lin, Q. Wu, L. Qiu, X. Huang, Image super-resolution using a dilated convolutional neural network, Neurocomputing 275 (2018) 1219–1230. `doi:10.1016/j.neucom.2017.09.062`.

[16] Q. Chang, K.-W. Hung, J. Jiang, Deep learning based image super-resolution for nonlinear lens distortions, Neurocomputing 275 (2018) 969–982. `doi:10.1016/j.neucom.2017.09.035`.

[17] D. Zhang, J. Han, C. Li, J. Wang, X. Li, Detection of co-salient objects by looking deep and wide, International Journal of Computer Vision 120 (2) (2016) 215–232.

[18] G. Cheng, P. Zhou, J. Han, Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images, IEEE Transactions on Geoscience and Remote Sensing 54 (12) (2016) 7405–7415.

[19] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, F. E. Alsaadi, A survey of deep neural network architectures and their applications, Neurocomputing 234 (November 2016) (2017) 11–26.

[20] M. T. McCann, K. H. Jin, M. Unser, Convolutional neural networks for inverse problems in imaging: A review, IEEE Signal Processing Magazine 34 (6) (2017) 85–95. `doi:10.1109/MSP.2017.2739299`.

[21] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, C. I. Sánchez, A survey on

deep learning in medical image analysis, Medical Image Analysis 42 (2017) 60–88. `doi:https://doi.org/10.1016/j.media.2017.07.005`.

[22] L. Xiang, Y. Qiao, D. Nie, L. An, W. Lin, Q. Wang, D. Shen, Deep auto-context convolutional neural networks for standard-dose pet image estimation from low-dose pet/mri, Neurocomputing 267 (2017) 406 – 416. `doi:https://doi.org/10.1016/j.neucom.2017.06.048`.

[23] Y. Chen, Y. Xie, Z. Zhou, F. Shi, A. Christodoulou, D. Li, Brain mri super resolution using 3d deep densely connected neural networks, in: Proceedings - International Symposium on Biomedical Imaging, Vol. 2018-April, 2018, pp. 739–742. `doi:10.1109/ISBI.2018.8363679`.

[24] Y. Chen, F. Shi, A. G. Christodoulou, Y. Xie, Z. Zhou, D. Li, Efficient and accurate mri super-resolution using a generative adversarial network and 3d multi-level densely connected network, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2018, pp. 91–99.

[25] J. Shi, Q. Liu, C. Wang, Q. Zhang, S. Ying, H. Xu, Super-resolution reconstruction of mr image with a novel residual learning network algorithm, Physics in Medicine and Biology 63 (8). `doi:10.1088/1361-6560/aab9e9`.

[26] J. Park, D. Hwang, K. Kim, S. Kang, Y. Kim, J. Lee, Computed tomography super-resolution using deep convolutional neural network, Physics in Medicine and Biology 63 (14). `doi:10.1088/1361-6560/aacdd4`.

[27] Z. Xiaole, H. Zhang, H. Liu, Y. Qin, T. Zhang, X. Zou, Single mr image super-resolution via channel splitting and serial fusion network, arXiv preprint arXiv:1901.06484.

[28] C. H. Pham, A. Ducournau, R. Fablet, F. Rousseau, Brain mri super-resolution using deep 3d convolutional networks, in: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), 2017, pp. 197–200. `doi:10.1109/ISBI.2017.7950500`.

34

[29] K. Thurnhofer-Hemsi, E. López-Rubio, N. Roé-Vellvé, E. Domínguez-Merino, M. A. Molina-Cabello, Super-resolution of 3D magnetic resonance images by random shifting and convolutional neural networks, in: Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), 2018, pp. 4008–4015.

[30] 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.

[31] 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.

[32] A. Bhattacharyya, On a measure of divergence between two multinomial populations, Sankhy: The Indian Journal of Statistics (1933-1960) 7 (4) (1946) 401–406.
URL http://www.jstor.org/stable/25047882

[33] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, R. L. Buckner, Open access series of imaging studies (oasis): Cross-sectional MRI data in young, middle aged, nondemented, and demented older adults, J. Cognitive Neuroscience 19 (9) (2007) 1498–1507. doi:10.1162/jocn.2007.19.9.1498.
URL http://dx.doi.org/10.1162/jocn.2007.19.9.1498

[34] A. J. Worth, MGH CMA internet brain segmentation repository (IBSR), http://www.cma.mgh.harvard.edu/ibsr/ (2010).

[35] F. Shi, J. Cheng, L. Wang, P. T. 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. doi:10.1109/TMI.2015.2437894.

[36] C.-H. Pham, R. FABLET, F. Rousseau, Multi-scale brain MRI super-resolution using deep 3D convolutional networks, working paper or preprint (Nov. 2017).
URL https://hal.archives-ouvertes.fr/hal-01635455

[37] J. Kim, J. K. Lee, K. M. Lee, Accurate image super-resolution using very deep convolutional networks, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1646–1654. doi:10.1109/CVPR.2016.182.

[38] B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee, Enhanced deep residual networks for single image super-resolution, 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2017) 1132–1140.

Vitae

Karl Thurnhofer-Hemsi (born 1990) received his B.Sc. in Computer Engineering and his M.Sc. in Mathematics degrees from the University of Málaga, Spain, in 2014. He joined the Medical and Health Research Center of the University of Málaga in 2015. He is currently a Ph.D. candidate at the Department of Computer Languages and Computer Science, University of Málaga. His technical interests are in medical image analysis, pattern recognition and image processing.

Ezequiel López-Rubio (born 1976) received his M.Sc. and Ph.D. (honors) degrees in Computer Engineering from the University of Málaga, Spain, in 1999 and 2002, respectively. He joined the Department of Computer Languages and Computer Science, University of Málaga, in 2000, where he is currently a Professor of Computer Science and Artificial Intelligence. His technical interests are in machine learning, pattern recognition and image processing.

Enrique Domínguez (born 1975) received his BSc, MSc and PhD degree in Computer Science from the University of Málaga in 1999, 2000 and 2007, respectively. He joined at the Department of Computer Science of the University of Málaga in 2000, where he is currently an associate professor. He has participated as a member of the program committee of prestigious international conferences and journals. His research areas include neurocomputation, optimization, image/video processing and computer vision.

Rafael Marcos Luque-Baena (born 1982) received the M.S. and Ph.D. degrees in Computer Engineering from the University of Málaga, Spain, in 2007 and 2012, respectively. He moved to Mérida, Spain in 2013, as a lecturer at the Department of Computer Engineering in the Centro Universitario de Mérida, University of Extremadura. Currently, he has come back to the University of Málaga (2016) and has a teaching position in the Department of Languages and Computer Science. He also keeps pursuing research activities in collaboration with other Universities. His current research interests include visual surveillance, image/video processing, neural networks and pattern recognition.

Núria Roé-Vellvé (born 1977) graduated in physics (2004) and obtained a PhD in Biomedicine (2011, honors) at the University of Barcelona. She is a postdoctoral researcher and research assistant at the Molecular Imaging Unit, within the Centro de Investigaciones Mdico-Sanitarias, at the General Foundation of the University of Mlaga. Her research areas include PET and MR image processing and quantification, and their applications to neuroimaging and oncologic studies, including human and small animal studies.