Robust edge-directed interpolation of magnetic resonance images

Zhenhua Mai\(^1\), Jeny Rajan\(^1\), Marleen Verhoey\(^2\) and Jan Sijbers\(^1\)

\(^1\) IBBT-Visionlab, Physics Department, Universiteit Antwerpen, Belgium
\(^2\) Bio-Imaging Lab, Biomedical Department, Universiteit Antwerpen, Antwerpen, Belgium

E-mail: zhenhua.mai@ua.ac.be
Abstract. Image interpolation is intrinsically a severely under-determined inverse problem. Traditional non-adaptive interpolation methods do not account for local image statistics around edges of image structures. In practice, this results in artifacts such as jagged edges and/or blurring. To overcome this short-coming, edge-directed interpolation has been introduced in different forms. One variant, New Edge Directed Interpolation (NEDI), has successfully exploited the “geometric duality” that links the low-resolution (LR) image to its corresponding high-resolution (HR) image. It has been demonstrated that, for scalar images, NEDI is able to produce better results than non-adaptive traditional methods, both visually and quantitatively. In this work, we return to the root of NEDI as a least-squares estimation method of neighborhood patterns, and propose a robust scheme to improve it. The improvement comes two-folds: firstly, a robust least-squares technique is used to improve NEDI’s performance to noise; secondly, the NEDI algorithm is extended with the recently proposed Non-Local Mean (NLM) estimation scheme. Moreover, the edge-directed concept is applied to the interpolation of multi-valued diffusion weighted images (DWI). The framework is tested on phantom scalar images and real diffusion images, and is shown to achieve better results than the non-adaptive methods, in terms of visual quality as well as quantitative measures.

Keywords: Edge-directed interpolation, Diffusion Weighted Images, Least Squares, Non-Local Means, Robust Regression

1. Introduction

Image interpolation (upsampling) aims to resolve unknown, high-resolution (HR) pixels from known, low-resolution (LR) pixels. Since the LR image is an approximation of the HR image, interpolation is an inverse problem. Additionally, as the number of unknown HR pixels usually exceeds that of the known LR pixels, it is in general an ill-posed problem. Certain models concerning the relation between HR and LR pixels will have to be used in order to determine the HR pixels from the LR ones [1]. The most widely used interpolation methods [2], such as bi-linear interpolation [3] and bicubic interpolation [4], readily employ global space-invariant models that fail to respect the local statistics around edges in the image. Consequently, they produce artifacts such as jagged edges, blurring and/or edge halos [1]. Moreover, valid structural information can be lost during the interpolation by bi(tri)-linear or even bi-cubic interpolation, as shown in [8].

In order to improve the interpolation quality, numerous methods based on more sophisticated models have been proposed [9–13,15]. Adaptive interpolation techniques [9] exploit the relation of local image intensities, with “warped distance”, to adapt the linear interpolation coefficients to better capture local features around the edges. Edge-directed interpolation (EDI) techniques [10,11,13] employ models that extract edge information in order to guide the interpolation in various ways. The edge information is often extracted explicitly, for example, in [10], a parameterized edge model is used to predict the HR edge information from the detected LR edge information. Alternatively, in [11], an HR edge map is first generated by filtering the input LR data.
Robust edge-directed interpolation of magnetic resonance images

in [13], the edge orientations are extracted and grouped into a finite set to guide the final interpolation.

Unfortunately, the pitfall of edge-interpolated techniques aforementioned is that they have to explicitly extract the edge information, and/or discretize the interpolation direction into a finite set. The successful extraction of edge information still presents a challenge to date, especially for certain low-contrast medical images. Also, for multi-valued images such as diffusion weighted images (DWI), the definition of “edge” is far from straightforward, in contrast to the scalar case. Moreover, a finite set of interpolation directions can prove to be artificial and unstable.

In order to avoid the problems associated with explicitly estimating edges, Li et al proposed to exploit the “geometric duality” between the covariance of a LR image and a HR image, with the help of linear prediction theory, and guide the interpolation with the implicitly retained edge information in covariance [15]. Harnessing the mathematical elegance, their method (NEDI) is able to produce results both visually pleasing and quantitatively competitive. However, because of the least squares nature of NEDI, it is not robust for outliers. In the presence of heavy noise, this drawback will severely hamper the performance of NEDI, as we will discuss more in detail in later sections. In this work, we utilize a robust least squares technique to improve this drawback. Furthermore, since NEDI uses a least squares fit of the pattern of the neighborhoods to achieve optimal interpolation, we make the connection between NEDI and the Non-Local Mean (NLM) method [16]. As is shown, NLM can provide a measure of the similarity between neighborhood patterns. The NLM concept is incorporated into our algorithm such that it links naturally with the improved least squares framework of NEDI.

As an application, our proposed interpolation method will be applied to DWI MR images. Diffusion imaging processing tasks involve interpolations in both pre- and post-processing stages. As mentioned previously, traditional interpolation methods are shown to be liable to lose image structural information. Moreover, interpolation is further complicated by a low signal-to-noise ratio nature of DWI [5], where the magnitude MR images are plagued by the Rician noise [6](as opposed to the typical Gaussian noise in the common digital images). Sophisticated techniques have already been developed to upsample the diffusion MR images in the slice direction [7], while for in-plane upsampling the traditional bi(tri)-linear interpolation methods are still the most commonly used in practice. By contrast, our algorithm is shown to preserve the image structure for diffusion MR images interpolation.

This paper is organized as follows: Section 2 covers our methodology. Firstly, section 2.1 briefly introduces NEDI; secondly, section 2.2 details our reinterpretation of NEDI as a least squares method; and then section 2.3 provides the details of our improvements based on the new understanding of NEDI. In section 3, experiments on both phantom and real images are described. Finally, in section 4, conclusions are drawn, and prospects for future work are given.
Robust edge-directed interpolation of magnetic resonance images

2. Methods

First, it should be noted that, while the original NEDI algorithm is 2D based, there is no algorithmic barrier that hinders its extension to 3D images. However, in this work, we limit our attention to 2D images and treat the 3D volumes as a series of 2D slices. This is out of consideration that MR images, especially DW images, typically have thick slices, as compared to the much smaller in-plane pixel dimension.

2.1. NEDI

We assume the LR image \( x_{i,j} \) of dimensions \( W \times H \), defined on a domain \( \Omega \) with \( \Omega = \{(i,j)|1 \leq i \leq W, 1 \leq j \leq H\} \), to be a downsampled version of the HR image \( y_{2i−1,2j−1} \) of dimension \((2W − 1) \times (2H − 1)\), i.e., \( y_{2i−1,2j−1} = x_{i,j} \), as shown in Fig. 1. The goal of interpolation is to compute pixels \( y_{2i,2j} \), \( y_{2i−1,2j} \) and \( y_{2i,2j−1} \). Let us consider for example the reconstruction of \( y_{2i,2j} \). It is assumed that this value can be reconstructed from its immediate \( n \times n \) neighbors (without loss of generality, \( n \) is taken to be 2 in the following) by a weighted sum:

\[
\hat{y}_{2i,2j} = \sum_{k=0}^{1} \sum_{l=0}^{1} \alpha_{2k+l} x_{i+k,j+l}
\]

where \( \alpha_i \) is the weight of each neighbor pixel \( x_{i,j} \) in determining \( y_{2i,2j} \). According to classical Wiener filtering theory [17], the optimal linear interpolation coefficients are given by:

\[
\alpha = R^{-1}r
\]

where \( \alpha \) is the vector containing weights \( \alpha_i \), \( R = [R_{kl}] \), \( 0 \leq k, l \leq 3 \) and \( r = [r_k] \), \( 0 \leq k \leq 3 \) are the local covariances, at HR, which are unknown. However, assuming
the so-called “geometric duality” [15], the correspondence between the HR covariance and the known LR covariance can be established, so that the HR covariance can be estimated from the LR one. Therefore, we have

$$R = \frac{1}{N} C^T C, \quad r = \frac{1}{N} C^T x$$

where $x = [x_1, x_2, x_3, x_4]^T$ is the neighbor vector that contains the $N$ (4 in this case) immediate neighbors of pixel $y_{2i,2j}$. Using the correspondence between the LR and HR images as shown in Fig. 1, $x$ can be replaced with $y = [y_1, y_2, y_3, y_4]^T$, where $y_i$ is the corresponding known HR pixel of the LR pixel $x_i$. $C$ is the $4 \times N$ matrix, the $k$th ($k \leq N$) row of which contains the 4 neighbors of $y_k$. Then, the weights $\alpha_i$ can be estimated from $C$ and $y$ as follows:

$$\alpha = (C^T C)^{-1} (C^T y)$$

For pixels $y_{2i-1,2j}$ and $y_{2i,2j-1}$, the computations are similar, except for a rotation and scaling of the neighborhood [15]. It is worth noting that, according to the original NEDI paper, the computation outlined previously is only performed on pixels whose neighbors have an intensity standard deviation larger than a certain threshold. This, as a crude form of edge detection, helps to alleviate the computational load, while still avoiding estimating edges explicitly. We have retained this mechanism in the implementation of our algorithm.

2.2. NEDI reinterpreted as Least Squares Fitting

Here we continue the discussion by looking at the NEDI formula, Eq. (4), from another viewpoint. Eq. (4) was derived from Wiener filtering theory [17]. However, we can immediately see that Eq. (4) is actually the solution to a least squares problem:

$$\hat{\alpha} = \arg \min_{\alpha} \|y - C \times \alpha\|^2$$

Now let us further decompose the above vectorized formula:

$$\hat{\alpha} = \arg \min_{\alpha} \left\| \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} \right\|^2 = \arg \min_{\alpha} \left\| \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} - \begin{bmatrix} y_{1,1}, \cdots, y_{1,4} \\ y_{2,1}, \cdots, y_{2,4} \\ y_{3,1}, \cdots, y_{3,4} \\ y_{4,1}, \cdots, y_{4,4} \end{bmatrix} \times \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} \right\|^2$$

where $r_i$ is the residual for each fit: $r_i = y_i - [y_{i,1}, \cdots, y_{i,4}] \cdot \alpha$. Following the logic in Eq. (1), we can interpret the weights $\alpha$ as characteristics of the pattern of a pixel’s surrounding neighbors. Therefore, the meaning of Eq. (4) becomes much clearer: what it achieves is a least squares fit of the patterns of the neighborhoods (row elements in $C$) in the search area spanned by $y$.

With NEDI interpreted as such, we now look at the algorithm (and its flaws) in a new light. As is well documented, OLS as used in the original NEDI is far from robust, and liable to the influence of outliers and/or noise [20]. While the presence of noise is obviously detrimental to the image interpolation task, “outliers” in the context of NEDI
2.3. Robust NEDI (R-NEDI)

In this section, we propose two improvements to the original NEDI algorithm. Firstly, the estimation is made robust with respect to outliers and/or noise. Secondly, NLM weighting is incorporated for performance improvement.

2.3.1. Robust iterative reweighted least squares fitting

In the literature, numerous methods to overcome the non-robustness of OLS estimation have been proposed [20]. In our work, we start from the well-known iterative reweighted least squares fitting method (IRLS) [21]. In IRLS, the sum of the squared residuals, weighted with a function of the
residuals from the previous iteration, is iteratively minimized, i.e.:

$$\alpha^{(k)} = \arg \min_{\alpha^{(k)}} \sum_i \omega^R(r_i^{(k-1)})(r_i^{(k)})^2$$

(7)

where the superscript \( (k) \) indicates the iteration \( k \). The function \( \omega^R(x) \) is a monotonously non-increasing function of \( x \).

The most commonly used weighting functions include “bisquare function”, “Huber function”, “fair function” and “Welsch function”. For a complete description of each function, readers are referred to [20]. Since the choice of weighting function is application-dependent, and crucial for the outcome, we conducted a simulation test to determine the optimal weighting function. The simulation was set up as in Fig. 3(a), where a 4-neighborhood setup was used. The intensities of the four pixels, \( y_i \), were randomized between a preset range, \( Y_{im} \leq y_i \leq Y_{im} \). The intensity of the central pixel \( Y \) was then set as a weighted sum, with weights \( \alpha \) fixed and known. Each IRLS testing, with different weighting functions, was supplied with 15 sets of neighbors and central pixels (which were corrupted with 10% Rician noise and 10% outliers), in order to perform a robust regression to estimate weights \( \alpha^R \). The error in weights estimation can then be computed as \( \| \alpha - \alpha^R \| \). This simulation was repeated 100000 times, and the accumulated errors for each IRLS with different weighting function are shown in Fig. 3(b). The tests have shown that the “Welsch function” is preferable to other functions, which is adopted in the final implementation of this work.

The power of IRLS can be effectively shown in a 1D demonstration, in Fig. 2, where the weighting function is taken to be the “Welsch function” [20], \( \omega^R(x) = e^{-x^2/c^2} \), with \( c \) controlling the degree of robustness. According to [20], setting \( c = 2.98546 \) can achieve an optimal asymptotic efficiency. In Fig. 2, where the IRLS fitting results for different \( c \) are shown, it is found that \( c \) values clustering around the aforementioned optimum value would result in more or less the same fitting results, while values that deviate far from that (for example, when \( c = 0.2 \) or 7.0) would result in similar results as in OLS. Therefore we adopt the value as recommended in the literature.

2.3.2. Non-local Means weighting

Since we have reinterpreted NEDI as a least squares fit of the neighborhood patterns, we naturally want to maximize its ability for doing so, while keeping in mind the context for “robustness”. Therefore, the newly emerged Non-local Means (NLM) algorithm [16], proved to be able to become a vital component. NLM algorithm was designed specifically for denoising purposes, whose success is largely due to its ability to differentiate between non-local neighborhood patterns (non-local “patches”). Namely, it estimates the underlying, noiseless pixel values from its non-local neighboring pixels based on the similarity of their corresponding neighborhoods. Looking past the algorithmic difference, this is the same principle as NEDI, especially after our reinterpretation. Furthermore, NLM does so with robustness and efficiency. We therefore make the connection of NLM to our algorithm, by incorporating it into the reweighting part of the least squares fit. The NLM-weighted iterative least squares
Robust edge-directed interpolation of magnetic resonance images

(a) Simulation set up  

(b) Simulation errors

Figure 3. A simulation test for determining the optimal weighting function.

The fit will take the following form:

$$\alpha^{(k)} = \arg \min_{\alpha^{(k)}} \sum_i \omega_i^N \cdot \omega^R(r_{i}^{(k-1)}) \cdot (r_{i}^{(k)})^2$$

(8)

Here $\omega_i^N$ is the NLM weight function [16], and is expressed as

$$\omega_i^N = e^{-\frac{\|N^i - N^{\text{current}}\|^2}{h^2}}$$

(9)

where $N^i$ stands for the neighborhood of $i^{th}$ pixel, $N^{\text{current}}$ the neighborhood of current pixel, and $h$ controls the smoothness of the NLM weight function.

According to [16], in a typical denoising application of the NLM, $h \approx 10 \cdot \sigma$, where $\sigma$ is the standard deviation of the noise in the image. For our specific application of NLM for interpolation purpose, we conducted experiments to ascertain the optimal $h$ that suits our need. As shown in Fig. 4(a), we tested with different $h$ values (as a factor of noise standard deviation $\sigma$) on synthetic phantom images (see Section 3) with different $\sigma$ being 10% (black), 20% (red) and 30% (blue) of the mean intensity of the synthetic image. The optimal $h$ values, in terms of PSNR (Peak Signal-to-Noise Ratio, Y axis), are found to coincide on $h = 1 \cdot \sigma$, as indicated by the indigo line in Fig. 4(a). This indicates a uniform scale factor of 1 when it comes to decide the value of $h$ as a function of $\sigma$, regardless of the noise levels (our experiments suggest that the factor of 1 also applies to cases with different upsampling factors, different images, etc.). Note that the scale factor of 1 is also in accordance with further developments of NLM algorithm in the denoising application, e.g., [22]. Fig. 4(b)-4(d) show sample slides from our NLM weighting influence test, with Fig. 4(b) the one without NLM weighting, Fig. 4(c) the normal robust IRLS with NLM weighting, and Fig. 4(d) the one with excessive NLM weighting (with high $h$). The artifacts in Fig. 4(b) mainly come from the singular results during the IRLS computation, which is caused by inappropriately assigning edge-pixels and non-edge pixels into the estimation. It is obvious that, with NLM weighting, the
Robust edge-directed interpolation of magnetic resonance images

![PSNR vs h](image)

(a) PSNR as a function of $h$

(b) No NLM weighting  
(c) Normal Robust OLS and NLM weighting  
(d) Heavy NLM weighting

Figure 4. Fig. 4(a) demonstrates the impact of different $h$ values on interpolation. Noise levels used are 10% (black), 20% (red) and 30% (blue) of the mean intensity of the synthetic image. Fig. 4(b)-4(d) show sample slides from our NLM weighting influence test, with Fig. 4(b) the one without NLM weighting, Fig. 4(c) the normal robust IRLS with NLM weighting, and Fig. 4(d) the one with excessive NLM weighting (high $h$).

Interpolation result is more smooth, and less artifact-prone, while oversmoothing with a high $h$ tends to lose structural details. Since part of the neighborhood pixel values are originally unknown, the computation in Eq. (8) has to be iterative. For the pseudo code of the NLM weighted IRLS fitting, see Fig. 5.

With the two proposed improvements, our algorithm is able to distinguish robustly between neighborhoods of different intensity patterns. As a simple example, we show in Fig. 6 an edge pattern, where four sample neighborhoods are considered. It is easy to see that the drastic difference between $N_2$ and other neighborhood regions ensures that
Robust edge-directed interpolation of magnetic resonance images

Initializing: setting k=1
interpolated, do

$$\min_{a_i} \sum_i (r_i^{(0)})^2$$

$$= \exp\left(-\frac{(r_i^{(k-1)})^2}{c^2}\right), \quad \omega_i^N = \exp\left(-\frac{||N_i^i - N_{\text{current}}||^2}{h^2}\right)$$

$$\min_{a_i} \sum_i \omega_i^N \cdot \omega_i^R (r_i^{(k-1)}) \cdot (r_i^{(k)})^2$$

$$y_i \leftarrow$$

\textbf{Figure 5.} The pseudo code for the NLM weighted IRLS fitting.

$N_2$ will get a small NLM weighting, therefore minimizing its influence on the outcome of the estimation. On the other hand, $N_1$ and $N_3$ are very similar, and therefore should be retained in the estimation. Furthermore, the pattern of $N_4$, under linear regression, is also similar to that of $N_1$ and $N_3$, and would contribute to the final estimation of the intensity pattern as well. As mentioned in the previous section, we retain the simple edge detection scheme from the original NEDI method to alleviate the computation load. Moreover, the computation is carried out in a neighborhood of 7 by 7 pixels, which we have found to be a good balance between computation load and algorithm performance.

3. Results and Discussion

3.1. Experimental Setup

In this section, we present the results for our Robust NEDI (R-NEDI), as compared to bicubic interpolation and NEDI, on geometric phantom images, synthetic images (from Brainweb [23]) and real images. The original 2D images, including the slices from the 3D volumes for both synthetic and real images, were of a dimension of 256 × 256, downsampled by a factor of 4, adding Rician noise of varying levels, and then used as the input for different interpolation methods. For real diffusion MR images, we show here the results from a real rat brain atlas, of size 265 × 256 × 21, with 6 gradient directions, each with 7 repetitions. It is important, however, to note that no special treatment is taken regarding the correlation between channels of the multi-valued DW images, i.e., each channel is interpolated independently. While the inter-channel correlation can be
exploited to produce better interpolation results for certain multi-valued images (e.g., color images in [25] and DT images in [24]), we here emphasize more on the impact of certain methodology on the scalar valued images.

For numerical evaluation, we use PSNR and Structural Similarity index (SSIM) [26] for comparing the scalar image interpolation. For evaluation of diffusion tensor images that resulted from the interpolation for the diffusion weighted images, we used the Overlapping of Eigenvalue/Eigenvector pairs (OVL) [27] to measure how well the interpolation is in agreement with the ground truth.

3.2. Geometric Phantom Image

Firstly, for the phantom images, the interpolation results for three different methods with Rician noise (standard deviation ranging from 5% to 40% of the image mean intensity) are shown in Fig. 7. As in Fig. 7, both edge-directed methods are able to better capture the edges which the bicubic method renders as jagged ones. However, due to the limitations of NEDI discussed in section 2.2, the NEDI result features artifacts around the thin edges, which become worse as the noise level intensifies. A simple explanation for this behavior of NEDI is rooted in its OLS scheme: it treats all neighbors inside a local region with equal importance, regardless of whether they belong to the same edge pattern or not. Consequently, the edge pattern estimated by NEDI will not truly reflect the real one. This drawback becomes apparent when the neighborhood contains small and/or thin edge patterns, and it is further exacerbated by the presence of the noise, as shown in Fig. 7. In comparison, the R-NEDI results not only correct for that, but also better preserve the shape of the rings, even when the noise level is high.
3.3. Synthetic and Real Images

For synthetic and real images, the results (with Rician noise from 5% to 20%) are shown in Fig. 8. Again, the bicubic results feature jagged edges and distorted image structures (as with heavy noise). The NEDI method is able to eliminate the jagged artifacts, yet as the noise increases, it also produces unnatural textures. By contrast, our improved algorithm not only retains the edge-preserving ability, but it can reproduce the image structures more consistently, even in the case of heavy noises. The PSNR and SSIM comparison table in Fig. 10 lends evidence to the statements above.

As for real scalar images, combining Fig. 9 and Fig. 10, a steady trend is the deterioration of the interpolations with respect to the rise of noise level, which is to be expected. Interestingly, bicubic method does not necessarily produce worse results than NEDI method, quantitative measurement wise. For example, when noise level is 10%, the PSNR and SSIM for bicubic method are 34.6 and 0.597, respectively, both better than those of the NEDI method, 34.3 and 0.592, respectively. Visually, it is not difficult to find fault in the bicubic interpolation, as evidenced by the jagged edges around the structure boundaries. However, despite the absence of jagged edges, NEDI produces blurred edges, as well as the textures similar to the synthetic case. In comparison, R-NEDI results (35.4 and 0.616 for PSNR and SSIM) feature smooth and defined edges, while texture artifacts caused by the noise are still largely under control.

The efficiency of the algorithm can also be demonstrated by its performance exclusively on the “edge” pixels. The edge pixels here are detected with one of the widely used methods, Canny detector [28], from the real image, as shown in Fig. 11(a). Fig. 11(b) shows the PSNR of the three methods we compared, with and without edge masking, with respect to varying levels of Rician noise. One can see that the interpolation error increases in general with an edge masking (overall drop in the PSNR), which is quite understandable given the fast changing statistics around edge points and henceforth the associated difficulty of capturing them. However, our edge-directed method is still able to reproduce the edge structure more accurately than either the Bicubic interpolation or NEDI.

Finally, we show the FA overlay, as well as the OVL, in Fig. 12. The FA map of the DTI estimated from the interpolated DWI is given a color code of green. The ground truth FA map (estimated from the ground truth DTI which is derived from the ground truth DWI) is given red. Both are overlaid onto each other, resulting in the FA overlay map. Wherever the interpolation results do not agree with the ground truth, the overlay map will show a patch of either red or green. The comparison shows that the R-NEDI result is characterized with fewer red/green patches, whose presence indicates the inaccuracy of interpolation. As mentioned before, DW images (and henceforth DT images) contain a lot of orientational information that is encoded in the image structures. Non edge-preserving methods will have difficulty in preserving and reproducing the structural information, and further lose the valid orientational information inherent in the DW images, as evidenced in the FA overlay figure. This
further demonstrates visually that the robust edge-preserving ability of our R-NEDI method can translate into the preservation of the essential structural information of the DW images.

4. Conclusion

In this work, we have improved upon NEDI based on our new understanding for its least squares fitting nature, and extended the edge-directed concept onto the diffusion MR image interpolation. The source of non-robustness of NEDI was identified and improvements were suggested accordingly. Our improvements on the original NEDI not only strengthen its ability to implicitly retain the edge information, but also make it more robust to noise. Our experiments have demonstrated that R-NEDI is up to common interpolation tasks, producing superior results to conventional interpolation methods. Moreover, its robust ability to reconstruct the fine details about image structures also makes it suitable for use in certain feature-dense image processing tasks, e.g., atlas construction, where registration could benefit from a more accurate and robust interpolation of the images. As for prospective works, this edge-directed concept for diffusion MR image interpolation can be further investigated with more robust models for least squares estimation. Furthermore, it could also be extended to interpolate on arbitrary spatial points, so that it is “interpolation” in the full sense.

5. ACKNOWLEDGEMENT

This work was financially supported by the Inter-University Attraction Poles Program 6-38 of the Belgian Science Policy, and by the SBO-project QUANTIVIAM (060819) of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT Vlaanderen). The authors would also wish to thank Wolfgang Jacquet for his useful discussions.

References

Figure 7. The comparison on a phantom image of dense thin edge patterns. Top row: from left to right are the original, HR image and the downsampled, LR image. Bottom rows: from left to right are the bicubic interpolation, NEDI result and R-NEDI result, respectively. From second top row to bottom, Rician noises are added in the downsampled images in each row. The standard deviations of the noise are set to be 5%, 10%, 20% and 40% of the mean image intensity, respectively.
Figure 8. The comparison on a synthetic brain image from Brainweb. Top row: from left to right are the original, HR image and the downsampled, LR image. Bottom rows: from left to right are the bicubic interpolation, NEDI result and R-NEDI result, respectively. From second top row to bottom, Rician noises are added in the downsampled images in each row. The standard deviations of the noise are set to be 5%, 10% and 20% of the mean image intensity, respectively.
Figure 9. The comparison on a real rat brain image. Top row: from left to right are the original, HR image and the downsampled, LR image. Bottom rows: from left to right are the bicubic interpolation, NEDI result and R-NEDI result, respectively. From second top row to bottom, Rician noises are added in the downsampled images in each row. The standard deviations of the noise are set to be 5%, 10% and 20% of the mean image intensity, respectively.
Figure 10. Comparison graphs for phantom, synthetic and real images. For phantom images, Rician noise from 2% to 40% is added, while for synthetic and real images, Rician noise from 2% to 20% is added. Figures 10(a), 10(b) and 10(c) are the PSNR comparisons for phantom image, synthetic image and real image, respectively. Figures 10(d), 10(e) and 10(f) are the corresponding SSIM comparisons. Black lines are for the bicubic results, red lines the NEDI results, and blue lines the R-NEDI results.
Figure 11. Algorithm evaluation with edge masking. Fig. 11(a) is the edge map generated with Canny edge detector on a real image. Fig. 11(b) is the PSNR comparison between Bicubic, NEDI and R-NEDI with and without the edge masking. Rician noise from 2% to 20% is added.

Figure 12. The comparison of the FA overlay with the diffusion tensor images as estimated from the interpolated diffusion weighted images. From left to right are the bicubic interpolation, and R-NEDI result, respectively. From top to bottom, the standard deviations of the added Rician noise are set to be 10% and 20% of the mean image intensity, respectively. OVL as a measure of the similarity between the interpolation and the ground truth is included in the brackets.