A Fast Progressive U-Net For MRI Denoising In The K-space

Lorenzo Di Luccio[1], Renato Giamba[1], Adriano Puglisi[1] and Stefano Giagu[2]

[1]*Department of Computer, Control and Management Engineering, Sapienza University of Rome, Italy*
[1]*Department of Physics, Sapienza University of Rome, Italy*

Abstract

Magnetic Resonance Imaging (MRI) is a prevalent non invasive imaging technique that produces high contrast anatomical images without ionizing radiation. However, due to long acquisition times, MRI scans are prone to noise and artifacts corruption. In this paper, we address the denoising problem by leveraging the intrinsic nature of the K-space domain where the noise naturally occurs during acquisition. We propose a light, efficient U-Net architecture that is specifically tailored to operate directly in the K-space. The model considers a residual learning-based estimate of the noise component across a range of noise levels and distributions and in particular for additive Gaussian noise. We also propose a SNR degradation based progressive training scheme that greatly improves performance across a wide range of noise levels. The network is computationally cost-effective and can be run on CPU in acceptable inference time, making it suitable for real time or resource constrained applications.

1. Introduction

MRI is a powerful medical imaging technique that provides detailed images of the internal structures of the body of a patient. The main benefit of MRI is the possibility to acquire images in multiple planes (sagittal, coronal, and axial) without repositioning the patient. MRI provides a very high contrast images of soft tissues, using their water content and molecular properties. It is also non invasive because it doesn't require X-rays or other radiations that could be harmful for the patient.

When a patient is placed inside an MRI scanner, the machine generates a magnetic field that aligns the protons in fat and water molecules along the direction of the field. Once the alignment is established, the scanner applies radio frequency pulses at the natural frequency of the protons. The energy stored in the pulses excites the alignment, and the protons precess along the axis of the magnetic field. When the pulses stop, the protons begin to realign themselves to their original configuration. As they realign, they release energy in the form of radio frequency signals, which are picked up by special receiving coils. They do not appear directly as an image; they are in K-space, a domain in which data is recorded in terms of spatial frequencies. To obtain a doctor-readable image, an inverse Fourier transform is then performed on this complex valued data, resulting in high-resolution grayscale images that relate to the anatomical structure of the tissues being examined.

Although magnetic resonance imaging (MRI) is a valuable tool for medical imaging, the resulting scans are inherently noisy and present some challenges. The acquisition process is time-consuming and expensive, so the patient must remain still throughout the scan, otherwise the resulting images will contain motion artifacts. Additionally, the MRI machine is sensitive to external factors such as temperature and electrical pulses, which can cause additional noise to be introduced. The thermal mobility of protons can potentially introduce noise into the image background, especially in low-signal regions.

Mathematically, certain probability distributions can be used to characterize noise in MRI scans. The equation 1 describes the bell-shaped symmetric probability distribution of Gaussian noise, often known as normal noise. This type of noise is additive, defined by a constant standard deviation across the image, and is typically attributed to thermal motion and electrical noise.

$$P_{\mu,\sigma}(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{1}$$

Rician noise is prominent in the low signal-to-noise ratio (SNR) regions of the scans. It follows a Rician distribution (equation 2) and emerges when the magnitude of the MRI signal is nonnegative and follows a Rayleigh distribution, while the phase is uniformly distributed. This type of noise is particularly difficult to handle compared to other types of noise.

$$P_{\mu,\sigma}(x) = \frac{x}{\sigma^2} e^{-\frac{(x-\mu)^2}{2\sigma^2}} I_0\left(\frac{x\mu}{\sigma^2}\right) \tag{2}$$

Another type of noise is Salt and Pepper noise, which appears as isolated, randomly distributed bright and dark pixels in MRI images. It may be caused by transmission problems in the collected data, random signal spikes, or electrical anomalies and is often referred to as impulse noise.

Low-intensity MRI images often contain background noise. This occurs when the amount of photons detected by the MRI system is subject to statistical fluctuations,

ICYRIME 2025: 10th International Conference of Yearly Reports on Informatics, Mathematics, and Engineering. Czestochowa, January 14-16, 2025

puglisi@diag.uniroma1.it (A. Puglisi)

creating variability in the returned signal. It follows a Poisson distribution (equation 3).

$$P_{\lambda}(k) = \frac{\lambda^{k} e^{-\lambda}}{k!} \tag{3}$$

Finally, the T1 and T2 relaxation phases are usually subject to exponential noise. This type of noise reflects the intrinsic unpredictability of the decay of MRI signals over time and has an exponential distribution (equation 4).

$$P_{\lambda}(x) = \lambda e^{-\lambda x} \tag{4}$$

A modern and popular approach to deal with long acquisition processes is to take a shorter one and capture less details, this involves acquiring only a portion of the K-space data which corresponds to a subset of spatial frequencies. With this process, the generated images contains more artifacts and more noise, but reliable and powerful deep learning models can enhance these scans. This paper is focused on the implementation of a model for denoising that can work with different levels of noise; in particular the focus is on denoising MRI scans with an additive Gaussian noise at the K-space level. The model, which follow a residual approach, should adapt also to other noise probability distributions.

2. Related Works

From image pixel analysis to contemporary deep learning models, a number of methods have been developed recently to solve the issue of denoising in magnetic resonance imaging (MRI). Using the subdivision suggested by J. Mohan et al. [1], we follow an outline of the primary MRI denoising techniques in this section, beginning with those that are not based on deep learning.

2.1. Filtering

These algorithms make use of a weighting kernel on the noisy image to reduce its variance. Damping of noise is done with this operation. Filters are basically of *linear* and *non linear* category.

Linear filters are ideal if the noise is evenly distributed. They are realized with fixed smoothing kernels and are not dependent on the content. There is a subgroup that acts in the spatial domain: the kernel acts on each pixel of a 2D neighborhood. The goal is to smooth, detect edges, or sharpen. Traditional examples include the Gaussian filter, the Sobel operator, and the Laplacian kernel. Linear filters remove Gaussian noise, but blur details as well.

The second type is temporal. There, the kernel is used in the time domain. There, the kernel is applied along time, on the same spatial pixels across two frames. It detects temporal changes, the most widely used ones are temporal averaging, differencing, and optical flow. They reduce noise caused by rapid change or motion but occasionally perturb content as well.

Non-linear filters are applied when the noise is not uniformly distributed. They rely on adaptive kernels or other rules of convolution. They are more advanced and can function differently depending on local intensity patterns.

The second is anisotropic diffusion filtering, which is a variation of standard spatial filtering in that it adapts its behavior to the local image properties. The algorithm adjusts the number of filtering iterations depending on the intensity gradient in the neighborhood: more iterations where there are smooth transitions, fewer where there are sharp transitions. This technique preserves edges more effectively than linear filters, but certain details along boundaries are still lost.

The second technique is the non-local means filter, which better exploits image redundancy. In this case, the algorithm treats larger structures as meaningful features and smaller patterns as noise that should be removed. Each pixel is adjusted by averaging values from a broader neighborhood area weighted by structure similarity. This technique overcomes some of the deficiencies of anisotropic diffusion but at the cost of more computation, since it involves a search in large neighborhoods for each pixel.

Finally, methods that combine domain and range filtering consider both spatial proximity and intensity similarity. A highly successful algorithm in this category is the bilateral filter, initially derived by Tomasi and Manduchi [2], subsequently generalized to MRI denoising by Xie et al. [3]. This filter averages spatial and photometric information to some extent by averaging neighbor pixel values and diminishing the influence of those with extreme intensity differences, thereby actually preserving edges. Its derivative, the trilateral filter, incorporates a time element as well, which causes it to respond more favorably in dynamic or multi-frame situations. These techniques, while effective in noise reduction and structural detail preservation, can produce visible halo artifacts along sharp edges, where the filtered to unfiltered boundary is evident. Domain techniques address image denoising by first converting the image from the spatial domain to a new representation, where noise and appropriate image structures can be separated more easily. Standard filters or operations specifically designed for the operation can then be applied to the new domain. The main classes within this class differ depending on the type of transform used and the character of the resulting representation.

One of the simplest methods is frequency domain denoising, which applies the Fourier transform to represent the image in terms of spatial frequency. Noise, being high frequency by nature, can be minimized in this domain without losing low-frequency data. This method is usually effective when the noise intensity is moderate, but breaks down when the noise increases.

The wavelet transform is an improvement over this method, as it decomposes the image into sets of frequency bands at different scales and allows simultaneous localization in both space and frequency. The multi resolution structure allows for better separation between noise and signal at different frequency levels. However, the wavelet basis can be challenged when representing curved or highly textured structures, which limits its applicability in some medical imaging scenarios.

This drawback was addressed with the creation of the curvelet transform, a generalization of wavelets specifically designed to produce a coarser representation of curves and edges. It is well suited to handle long and anisotropic features, and is ideal for images with highly directional textures. Despite this advantage, the transform produces artifacts in smooth regions, where the redundancy of the representation is lower.

The contourlet transform takes this concept a step further by combining multiscale and directional decomposition. It starts with a Laplacian pyramid for coarse image structures and then uses directional filtering to capture edges at different orientations. This produces a more expressive and flexible representation, especially for contours and geometric shapes. However, the increased complexity of this method leads to increased computational costs, especially for the fine-grained edge preservation required at multiple scales.

2.2. Deep Learning approaches

In recent years, machine learning and then deep learning started to become popular, because of the better computational performance offered by modern GPUs and the brilliant results obtained. The first improvement in the use of deep learning for images was the introduction of hybrid neural networks models [4, 5, 6, 7, 8] and convolutional neural networks (CNNs) [9, 10, 11] with the architecture *LeNet* proposed by LeCun et al. [12]. This model was taken as inspiration by subsequent and popular networks like *AlexNet* [13], *GoogLeNet* [14], *VGG* [15], transformers [16, 17] .

Afterwords, researchers believed that increasing the number of convolutional layers in the networks would have lead to better performances, but it was not true because the models suffered from the vanishing gradient problem. To overcome this issue, He et al. [18] proposed *ResNet*, a new network with the so called residual layers where the new outputs are computed with an additive update from the previous inputs. Moreover, *ResNet* allowed to obtain better performances and a more stable training with less convolutional layers.

ResNet was also one of first deep learning models used for general image denoising, together with the *Autoencoder* presented by Hinton and Salakhutdinov [19]. The main application is to learn a latent and more compact representation of an image by downgrading it in the first half of the model and then reconstructing it back in the second half; this process can be adapted to denoising by passing noisy images as input and expecting as output their denoised version. Similar architectures have also been applied in other areas of the medical sector, such as in the processing of speech signals for the automatic detection of speech disorders [20]. Some years later, Ronneberger et al. [21] proposed a new architecture, the *U-Net*, consisting in a fusion between the *Autoencoder* and the *ResNet*, that achieved good results in biomedical image segmentation. Also the *U-Net* can be adapted to perform general image denoising and so in particular MRI denoising.

Although the deep learning MRI denoising performance was adequate, a new problem arose. MRI scans with noise distributions other than Gaussian could not be accommodated by the networks. This drawback stems from the straightforward methodology of the models, which learned to produce a clean image from a noisy image. This indicated that the models had only worked on denoising one distribution, or perhaps a limited number of related distributions. Zhang et al. [22] proposed a new approach that they called *residual learning* in which the model learns to predict the residual image, i.e. the noise from the noisy image, with the possibility of focusing on more and different noise distributions. *Residual learning* also introduced regularization in the training process and boosted the image denoising performances.

3. Methodology

This work introduces a new method for MRI denoising that try to exploit and merge the benefits of previous works, keeping also an eye on speed.

3.1. Data Acquisition

The dataset is the Information eXtraction from Images (IXI) [23], which includes approximately 600 MRI scans in NIFTI format. Data were acquired from healthy subjects using different acquisition protocols (T1, T2, PD, MRA, DTI) at three hospitals in London, namely Hammersmith Hospital (Philips 3T), Guy's Hospital (Philips 1.5T), and the Institute of Psychiatry (GE 1.5T). Each scan is associated with protocol specific parameters, publicly available on the dataset website.

3.2. MRI Scan Preprocessing

The main preprocessing step is to extract 2D slices from the T1-weighted volumetric MRI of each subject. This procedure offers several advantages. 2D image processing is faster and more suitable for real-time situations, as it uses less processing resources than full-volume processing. A more focused investigation is also possible, as regions of interest are often located in a single plane, such as axial, coronal or sagittal. This makes it possible to focus on the relevant anatomical elements and eliminates the need to consider the entire volume. 2D images are easier to visualise and evaluate and provide a recognisable and comprehensible representation of the data. Another advantage is the reduction in the overall size of the data, which facilitates its transmission or storage, especially in resource limited environments.

Despite these advantages, relying solely on 2D slices has some disadvantages. It may not accurately represent complex structures or dynamic processes and may result in the loss of 3D or temporal information. In some situations, combining both modalities and examining selected slices in the context of the full volume may be more useful. For each of the 581 subjects, for a total of 43,575 images, the 25 central slices in the coronal, axial, and sagittal planes are selected to avoid completely or mostly black images. A padding technique involving continuous zero-filling is used to uniformly scale all photos to the maximum size, since slices acquired from different planes have variable sizes.

After preprocessing, the dataset is split into the training (75%), validation (15%) and test(10%);

Before loading the images in the dataset they are normalized in the $[0, 1]$ range, then the noisy images are generated by adding Gaussian noise in the K-space obtained by the Fourier transform.

In particular, the levels of added noise is based on the Signal-to-Noise Ratio (SNR), with the standard deviation σ computed using equation 5, where $\overline{|\mathcal{F}[I]|}$ is the average of the magnitude of the Fourier transform of the image.

$$\sigma = \frac{\overline{|\mathcal{F}[I]|}}{10^{\frac{\text{SNR}}{20}}} \tag{5}$$

3.3. Proposed Model

The model proposed in this work is a modified version of the classic *U-Net*. The main modifications concern several aspects of the network. In both the descending and ascending parts, five convolutional layers are present. Although additional layers can be added, the size of the preprocessed MRI images is 256×256, so adding more layers would lead to activation maps that are too small. Furthermore, the network is faster and lighter by avoiding additional layers.

Convolutions do not include bias, since the introduction of bias parameters has shown a drastic worsening of the network performance, generating images with anomalous visual artifacts. Similarly, normalization techniques were not used. This is because the U-Net model already tends to adjust the activations thanks to the concatenation mechanism between the old and new features in the upsampling phase. Another reason is related to efficiency, the network remains faster and, in the tests carried out, the denoised images with or without normalization did not present relevant differences. PReLU was adopted as the activation function, since the presence of a trainable parameter improves the overall performance of the network according to the metrics used.

The network can be described with simple equations following its division in layers.

The down part can be seen as the function $\text{DownConv2d}\left(d_i\right)$ defined as in equation 6, in which the d_i parameter represents the features computed by the previous layer.

$$\begin{aligned} z_i &= \text{PReLU}\left(\text{Conv2d}\left(d_i\right)\right) \\ d_{i+1} &= \text{PReLU}\left(\text{Conv2d}\left(z_i\right)\right) \end{aligned} \tag{6}$$

The up part can be seen as the function $\text{UpConv2d}\left(u_i, d_{N-i-1}\right)$ defined as in equation 7, in which the u_i parameter represents the features computed by the previous layer and the d_{N-i-1} parameter represents the correspondent features computed by the down part to be concatenated.

$$\begin{aligned} z_i &= \text{cat}\left(\text{PReLU}\left(\text{ConvT2d}\left(u_i\right)\right), d_{N-i-1}\right) \\ u_{i+1} &= \text{PReLU}\left(\text{Conv2d}\left(z_i\right)\right) \end{aligned} \tag{7}$$

Since the model (figure 1 for a visual representation) uses the residual approach, the entire network can be seen as the function $n = \text{Net}\left(I_n\right)$, where the input i_n is a batch of noisy MRI scans in the K-space and the output n is the estimated batch of noise also in the K-space.

3.3.1. Loss Function

Image denoising in general can be considered a regression task, so a well suited loss function is the *mean squared error* (MSE) defined in equation 8.

$$\text{MSE}\left(y_{pred}, y\right) = \frac{1}{N}\sum_{i=1}^{N}\left(y_{pred,i} - y_i\right)^2 \tag{8}$$

In particular we use the MSE applied to noisy scans and free noise scans is called *pixel loss* (PL) (equation 9), which is the main part of the total loss function.

Some works like Zhao et al. [24] or Mustafa et al. [25] showed that in image regression tasks, the MSE succeeded in performing the task with good results, but it blurred the resulting images. A popular way to overcome this problem is to add other parts in the total loss function, focusing on the features [24], or to design from scratch a specific and more complex loss function [25].

For the proposed network, we integrates other two parts in the total loss function, the *frequency loss* (FL)

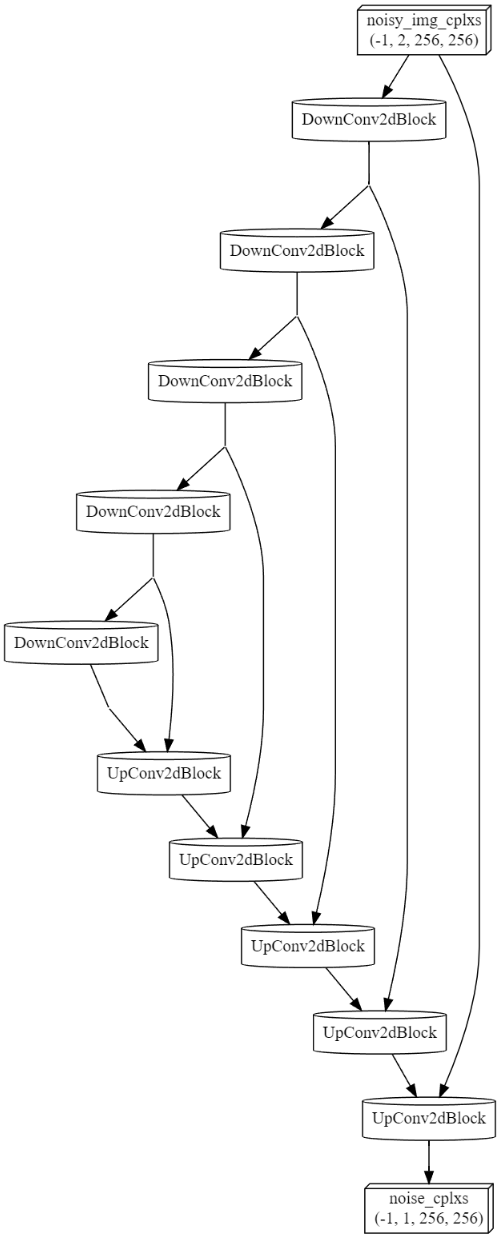


Figure 1: Structure of the Fast U-Net model with 5 downsampling and 5 upsampling layers, connected via skip connections.

(equation 10) that is the sum of MSE applied to real and imaginary part of the Fourier transform of the noisy scans and the free noise ones, and the *edge loss* (EL) (equation 11 that is the MSE applied to some edge features of the noisy scans and the free noisy ones; in particular the f function used are the *Laplacian* and/or the *Sobel* filters.

$$\mathrm{PL}\left(I_{\overline{n}}, I\right) = \mathrm{MSE}\left(I_{\overline{n}}, I\right) \tag{9}$$

$$\mathrm{FL}\left(I_{\overline{n}}, I\right) = \begin{matrix} \mathrm{Re}\left[\mathrm{MSE}\left(\mathcal{F}\left[I_{\overline{n}}\right], \mathcal{F}\left[I\right]\right)\right] + \\ \mathrm{Im}\left[\mathrm{MSE}\left(\mathcal{F}\left[I_{\overline{n}}\right], \mathcal{F}\left[I\right]\right)\right] \end{matrix} \tag{10}$$

$$\mathrm{EL}\left(I_{\overline{n}}, I\right) = \mathrm{MSE}\left(f\left(I_{\overline{n}}\right), f\left(I\right)\right) \tag{11}$$

The total loss function L is the simple sum of the three previously introduced parts: the *pixel loss* (PL), the *frequency loss* (FL) and the *edge loss* (EL).

$$\mathrm{L}\left(I_{\overline{n}}, I\right) = \mathrm{PL}\left(I_{\overline{n}}, I\right) + \mathrm{FL}\left(I_{\overline{n}}, I\right) + \mathrm{EL}\left(I_{\overline{n}}, I\right) \tag{12}$$

4. Experiments & Results

The experiments were conducted on a Google Colab environment with an NVIDIA T4 GPU (CUDA 11.8), 12 GB RAM, and Intel Xeon CPU at 2.20 GHz. Python version 3.10, and the PyTorch version 2.0.1 with CUDA.

We used ADAM optimizer, the starting learning rate of 0.001 and batch size of 64. A learning rate scheduler was used to reduce the learning rate by a factor of 0.1 at every instance that the validation loss failed to decrease for two successive epochs. Training was also monitored using an early stopping criterion that terminated if validation loss failed to drop for five epochs.

Model comparison was based on two common image reconstruction evaluation metrics,namely the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). The PSNR is defined as

$$\mathrm{PSNR}(I, J) = 10 \log_{10}\left(\frac{\max(I)^2}{\mathrm{MSE}(I, J)}\right), \tag{13}$$

where I and J are the reference and reconstructed images, respectively, and MSE is the mean squared error. The SSIM is computed as

$$\mathrm{SSIM}(I, J) = \frac{(2\mu_I\mu_J + c_1)(2\sigma_{IJ} + c_2)}{(\mu_I^2 + \mu_J^2 + c_1)(\sigma_I^2 + \sigma_J^2 + c_2)}, \tag{14}$$

where μ_I, μ_J are the local means, σ_I^2, σ_J^2 the variances, and σ_{IJ} the covariance between I and J, with c_1 and c_2 being constants to stabilize the division. Two pairs of training experiments can be distinguished. The first uses MSE loss, while the second uses MAE loss plus another type of normalization. The studies vary in how noisy scans are produced within each pair. Specifically, the second uses a progressive technique, while the first uses a random one.

4.1. Evaluation under MSE and MAE

The training phase was structured around two main configurations, each evaluated under different conditions. The first group of experiments employed the Mean Squared Error (MSE) as loss function, while the second used the Mean Absolute Error (MAE), along with a different input normalization strategy based on centering the data with zero mean and unit standard deviation, instead of scaling it to the [0,1][0,1] interval. Within each group, two noise injection strategies were considered: random and progressive.

In the first experiment, the training set was corrupted by Gaussian noise with a signal to noise ratio (SNR) randomly sampled from a discrete uniform distribution between -5 and 5. For the validation set, the SNR was fixed at -5 to maintain a challenging evaluation scenario. The model was trained with early stopping based on validation loss.

The second experiment introduced noise in a progressive manner, the training began with images at SNR = 5 and gradually moved down to SNR = -5. At each level, the model was trained separately using a validation set with matching noise level. This setup was designed to let the model adapt incrementally to increasing noise intensity.

The third and fourth experiments mirrored the structure of the first and second, respectively, but used the MAE loss instead of MSE, and applied the aforementioned centered normalization. These variations aimed to evaluate the impact of a loss function less sensitive to outliers and a normalization scheme more common in deep learning workflows. Detailed results for all experiments are reported in Table 1.

4.2. The MAE and the centered standard normalization

Although the random approach introduces stochasticity in the training procedure allowing the model to be more robust to different levels of noise earlier, the progressive approach has faster convergence time because the model only has to adapt to the next level of noise from the previous without the abrupt shifts.

Moreover, MSE can be sensitive to the presence of noise or outliers in the data because it strongly penalizes large deviations from the true values, due to the squared function. So the resulting denoised images could be overly influenced by noise. Instead, the Mean Absolute Error (MAE) (equation 15) is more robust to noise and outliers.

$$\text{MAE}\left(y_{pred}, y\right) = \frac{1}{N}\sum_{i=1}^{N}\left|y_{pred,i} - y_i\right| \qquad (15)$$

It treats errors uniformly, due to the absolute value function making it less susceptible to extreme values in the data. Consequently, the denoised images produced tend to be more noise resistant. Also, MAE tends to produce images that are visually crisper, but with better preservation of fine structures. MSE instead tends to suppress high-frequency details and edges in favor of noise reduction. For this reasons, the other two experiments are made with the same loss function, but substituting each instance of MSE with MAE.

Another change is to have the normalized MRI scans centered with mean 0 and standard deviation 1 instead of having them normalized in the interval between 0 and 1. This because this type of normalization is widely used in lots of works and tends to have better performances in most of the cases.

4.3. Results

From Table 1 and Figures 2a–3d, the following trends are easily observable. The most obvious is that progressive noise injection consistently outperforms random noise injection. With both loss functions, MSE and MAE, the progressive setting improves the average by about 4.8 dB for PSNR and from 0.14 to 0.15 for SSIM. In addition to achieving higher scores, this training approach is also more stable, with lower variance at various noise levels. Comparing the two loss functions, the model trained with MSE has slightly better SSIM values. The difference is small, about 0.01 on average, but constant. This indicates that the model trained with MSE is better at keeping structural details intact. In contrast, the model trained with MAE is superior in PSNR by about 0.2 dB on average. This holds true for all noise levels and suggests that MAE may be better at removing noise, albeit at the expense of slightly worse structural fidelity These results highlight the behavior of the proposed architecture under different training conditions. To further contextualize its performance, we compare it against existing state-of-the-art denoising approaches.

4.4. Comparisons with other methods

To evaluate the denoising performance of the networks, they are compared with other four denoising methods. In particular, with Optimized NLM by Coupé et al. [26], WSM by Coupé et al. [27], MCDnCNNg and MCDnCNNs by Jiang et al. [28].

It is useful to point out that the four chosen methods work with different datasets and have a different noise generation procedure. Therefore, the collected metrics to make comparisons are different (a kind of average on the level of noise) from the real ones.

Analyzing the table 2, the two fast U-Net trained with the progressive approach have overall very similar metrics with the other methods. In particular they are better on MRI scans with low noise, while they lose some performances on scans with high noise. This is due to the trade-offs made to keep the network fast.

Table 1
Metrics. The table shows the results of the experiments.

Noise Level (SNR)	Architecture	PSNR	SSIM
$\sim \mathrm{U}\,(-5, 5)$	*fast U-Net (MSE)*	29.77	0.74
	fast U-Net (MAE)	30.10	0.72
5	*fast U-Net (MSE)*	38.33	0.93
	fast U-Net (MAE)	38.66	0.93
4	*fast U-Net (MSE)*	37.74	0.93
	fast U-Net (MAE)	37.89	0.92
3	*fast U-Net (MSE)*	36.99	0.92
	fast U-Net (MAE)	37.16	0.92
2	*fast U-Net (MSE)*	36.20	0.91
	fast U-Net (MAE)	36.34	0.90
1	*fast U-Net (MSE)*	35.41	0.90
	fast U-Net (MAE)	35.57	0.89
0	*fast U-Net (MSE)*	34.58	0.89
	fast U-Net (MAE)	34.78	0.87
-1	*fast U-Net (MSE)*	33.80	0.88
	fast U-Net (MAE)	34.02	0.86
-2	*fast U-Net (MSE)*	33.02	0.86
	fast U-Net (MAE)	33.22	0.84
-3	*fast U-Net (MSE)*	32.27	0.85
	fast U-Net (MAE)	32.46	0.82
-4	*fast U-Net (MSE)*	31.46	0.83
	fast U-Net (MAE)	31.70	0.80
-5	*fast U-Net (MSE)*	30.72	0.81
	fast U-Net (MAE)	30.96	0.78
Average $(-5 \div 5)$	*fast U-Net (MSE)*	34.59 ± 2.57	0.88 ± 0.04
	fast U-Net (MAE)	34.80 ± 2.57	0.87 ± 0.05

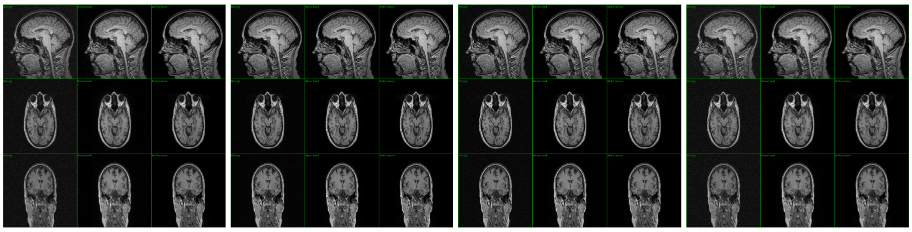
Table 2
Metrics from other works. The following table shows averaged metrics of other methods and the ones presented to make comparisons.

Method	Low Noise PSNR	High Noise PSNR	High Noise SSIM	High Noise SSIM
Optimized *NLM*	32.96	29.01	0.85	0.74
WSM	33.14	28.95	0.85	0.73
MCDnCNNg	35.46	31.20	0.90	0.81
MCDnCNNs	36.24	32.58	0.91	0.85
fast U-Net (MSE) (ours)	36.93	32.25	0.92	0.85
fast U-Net (MAE) (ours)	37.12	32.47	0.91	0.82

5. Conclusions

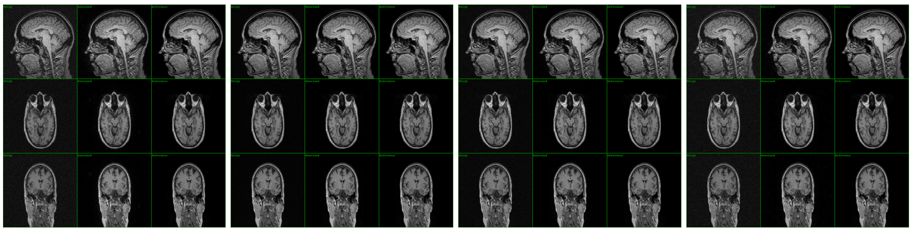
In this work, a new approach to MRI denoising is presented, introducing several novelties in both the noise generation phase and the training strategy, as well as an architectural modification of the classical U-Net to improve its speed while maintaining good performance.

The noise generation procedure, based on the K-space and the signal-to-noise ratio (SNR), allows to obtain more realistic noisy scans. This realism is also due to the dependence of the noise on the mean of the Fourier transform modulus. However, this type of generation makes the



(a) MSE, random (b) MSE, SNR=5 (c) MSE, SNR=0 (d) MSE, SNR=-5

Figure 2: Fast U-Net denoiser results using MSE loss under different training regimes.



(a) MAE, random (b) MAE, SNR=5 (c) MAE, SNR=0 (d) MAE, SNR=-5

Figure 3: Fast U-Net denoiser results using MAE loss under different training regimes.

Table 3
Time Performances. The following table shows averaged execution times on different batch sizes of the two architectures.

Batch Size	L1 Time	MSE Time
1 img	130 ms	141 ms
2 imgs	279 ms	239 ms
3 imgs	373 ms	409 ms
4 imgs	555 ms	544 ms
16 imgs	2.39 s	2.39 s
64 imgs	10.4 s	9.95 s

comparison with other methods more complex, since in most works a simulated noise based on predefined percentages is preferred.

The progressive training approach, where the SNR varies from 5 to -5, represents a turning point for the improvement of the results both in terms of PSNR and SSIM, as well as leading to a clear visual quality in the denoised images. Furthermore, the adaptive nature of this strategy allows the network to maintain the denoising performance already acquired along all the considered noise levels.

The proposed U-Net, composed of only five layers and without normalizations, can run in short times even on CPU. This compromise in terms of speed does not lead to a significant degradation of the metrics, which remain comparable with those obtained by other methods based on deep learning. It should also be noted that, although processing time is rarely analyzed in the literature, today in the medical field it is increasingly important to obtain reliable results in short times.

6. Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT, Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] J. Mohan, V. Krishnaveni, Y. Guo, A survey on the magnetic resonance image denoising methods., Biomed. Signal Process. Control. 9 (2014) 56–69.

[2] C. Tomasi, R. Manduchi, Bilateral filtering for gray and color images, 6th Int. Conf. Comput. Vis., Bombay, India. (1998) 839–846.

[3] J. Xie, P.-A. Heng, M. Shah, Image diffusion using saliency bilateral filter., IEEE Trans. Information Technology in Biomedicine 12 (2008) 768–771.

[4] G. Capizzi, C. Napoli, L. Paternò, An innovative hybrid neuro-wavelet method for reconstruction of missing data in astronomical photometric sur-

veys, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 7267 LNAI, 2012, p. 21 – 29. doi:10.1007/978-3-642-29347-4_3.

[5] C. Napoli, G. Pappalardo, E. Tramontana, R. K. Nowicki, J. T. Starczewski, M. Woźniak, Toward work groups classification based on probabilistic neural network approach, in: Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), volume 9119, 2015, p. 79 – 89. doi:10.1007/978-3-319-19324-3_8.

[6] F. Bonanno, G. Capizzi, C. Napoli, Some remarks on the application of rnn and prnn for the charge-discharge simulation of advanced lithium-ions battery energy storage, in: SPEEDAM 2012 - 21st International Symposium on Power Electronics, Electrical Drives, Automation and Motion, 2012, p. 941 – 945. doi:10.1109/SPEEDAM.2012.6264500.

[7] G. Capizzi, G. L. Sciuto, C. Napoli, E. Tramontana, A multithread nested neural network architecture to model surface plasmon polaritons propagation, Micromachines 7 (2016). doi:10.3390/mi7070110.

[8] G. Lo Sciuto, G. Capizzi, R. Shikler, C. Napoli, Organic solar cells defects classification by using a new feature extraction algorithm and an ebnn with an innovative pruning algorithm, International Journal of Intelligent Systems 36 (2021) 2443 – 2464. doi:10.1002/int.22386.

[9] F. Fiani, V. Ponzi, S. Russo, Keeping eyes on the road: Understanding driver attention and its role in safe driving, in: CEUR Workshop Proceedings, volume 3695, 2023, p. 85 – 95.

[10] E. Iacobelli, D. Pelella, V. Ponzi, S. Russo, C. Napoli, A fast and accessible neural network based eye-tracking system for real-time psychometric and hci applications, in: CEUR Workshop Proceedings, volume 3870, 2024, p. 32 – 41.

[11] F. Fiani, S. Russo, C. Napoli, A fully automatic visual attention estimation support system for a safer driving experience, in: CEUR Workshop Proceedings, volume 3695, 2023, p. 40 – 50.

[12] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Computation 1 (1989) 541–551.

[13] 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.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, 2015.

[15] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014.

[16] A. Alfarano, G. De Magistris, L. Mongelli, S. Russo, J. Starczewski, C. Napoli, A novel convmixer transformer based architecture for violent behavior detection, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 14126 LNAI, 2023, p. 3 – 16. doi:10.1007/978-3-031-42508-0_1.

[17] S. Russo, F. Fiani, C. Napoli, Remote eye movement desensitization and reprocessing treatment of long-covid- and post-covid-related traumatic disorders: An innovative approach, Brain Sciences 14 (2024). doi:10.3390/brainsci14121212.

[18] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, 2015.

[19] G. E. Hinton, R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (2006) 504–507.

[20] L. Corvitto, L. Faiella, C. Napoli, A. Puglisi, S. Russo, et al., Speech and language impairment detection by means of ai-driven audio-based techniques, in: CEUR WORKSHOP PROCEEDINGS, volume 3869, CEUR-WS, 2024, pp. 19–31.

[21] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, 2015.

[22] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising., CoRR abs/1608.03981 (2016).

[23] Biomedical Image Analysis Group, Imperial College London, Ixi dataset, https://brain-development.org/ixi-dataset/, 2006. RRID:SCR_005839.

[24] H. Zhao, O. Gallo, I. Frosio, J. Kautz, Loss functions for image restoration with neural networks., IEEE Trans. Computational Imaging 3 (2017) 47–57.

[25] A. Mustafa, A. Mikhailiuk, D. A. Iliescu, V. Babbar, R. K. Mantiuk, Training a task-specific image reconstruction loss, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 2319–2328.

[26] P. Coupé, P. Yger, S. Prima, P. Hellier, C. Kervrann, C. Barillot, An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images., IEEE Trans. Med. Imaging 27 (2008) 425–441.

[27] P. Coupé, P. Hellier, S. Prima, C. Kervrann, C. Barillot, 3d wavelet subbands mixing for image denoising., Int. J. Biomedical Imaging 2008 (2008) 590183:1–590183:11.

[28] D. Jiang, W. Dou, L. P. J. Vosters, X. Xu, Y. Sun, T. Tan, Denoising of 3d magnetic resonance images with multi-channel residual learning of convolutional neural network., CoRR abs/1712.08726 (2017).