Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Circumventing the curse of dimensionality in magnetic resonance fingerprinting through a deep learning approach.

(Article begins on next page)

03 May 2024

Circumventing the Curse of Dimensionality in Magnetic Resonance Fingerprinting through a Deep Learning Approach

Marco Barbieri*, Leonardo Brizi*, Enrico Giampieri*, Francesco Solera†, Gastone Castellani*, Claudia Testa* and Daniel Remondini*

* Department of Physics and Astronomy, University of Bologna, Italy

† Deep Vision Consulting, Modena, Italy

Abstract

MR fingerprinting (MRF) is a rapid growing approach for fast quantitave MRI. A typical drawback of dictionary-based MRF is its explosion in size as a function of the number of reconstructed parameters, according to the curse of dimensionality. Deep Neural Networks (NNs) have been proposed as a feasible alternative, but these approaches are still in their infancy.

We tested different NN pipelines on simulated data: we studied optimal training procedures by including different strategies of noise addition and parameter space sampling, to achieve better accuracy and robustness to noise. Four MRF sequences were considered, two of them designed to be more specific for B_1^+ parameter encoding: IR-FISP, IR-FISP-B_1, bSSFP and IR-bSSFP-B_1. A comparison between NN and the dictionary approaches was performed using a numerical brain phantom.

Results demonstrated that training with random sampling and different levels of noise variance yielded the best performance. NN performance was greater or equal than dictionary-based approach in reconstructing MR parameter maps: the difference in performance increased with the number of estimated parameters, because the dictionary method suffers from the coarse resolution of the MR parameter space sampling. The NN approach resulted more efficient in terms of memory and computational burden, and thus has great potential in large-scale MRF problems.

I. INTRODUCTION

Magnetic Resonance Fingerprinting (MRF) is a fast quantitative MRI technique able to obtain multi-parametric maps in one-shot measurement [1]; many applications of the technique have been investigated since its birth, ranging from brain imaging [2] to body MRI [3], [4].

The key concept of MRF is to apply a train of RF pulses with flip angle (FA) and repetition time (TR) varying according to a pattern designed to differentiate as much as possible a signal response for different tissues, so that for each voxel a so called fingerprint of the tissue is acquired. The original methodology is based on the template matching of the experimental fingerprint with a precomputed dictionary of simulated signal evolutions, and the dot

Dr. Claudia Testa is corresponding author, Email: claudia.testa@unibo.it

product is usually adopted as a fast similarity measure. Dictionary exploding size, matching accuracy, robustness to noise and under sampling artifacts are the main challenges to face to bring MRF into clinical applications. The more parameters are encoded into the pulse sequence simulations (T_1, T_2, B_0, B_1^+, etc...) the bigger the size of the precomputed dictionary has to be. However, confounding factors such as B_1^+ field inhomogeneities are known to be a source of artifacts in MRI [5], and researchers are taking into account these confounding factor also in MRF to improve parameters estimation [6], [7]. However, big dictionaries are hard to handle because they are costly both in memory usage efficiency and in computational time for the matching procedure, with size up to 150 GB [7]. This problem has driven the MRF community forward methods aiming to compress the size of the dictionaries by both applying Single Value Decomposition (SVD) to generate low rank approximation of the fingerprinting signals, and reducing the entries of the dictionaries by using a polynomial fitting [8], [9], [10]. Nevertheless, these methods make an additional approximation, which can affect accuracy.

A strategy to overcome the limitations of the dictionary based template matching is taking advantage of Machine Learning algorithms to learn a model able to predict, after a supervised training procedure, the MR parameters given the experimental fingerprint as input. Among these, Neural Networks (NN) algorithm is a feasible tool to accomplish such a task, as NNs have been demonstrated to be universal function approximators given enough train data and model complexity [11]. Hence, a sufficiently large NN can learn the Inverse Transfer Function (ITF) to map the acquired MRF signal into the MR parameter space. Moreover, once the NN is trained, the prediction operation has many advantages with respect to the template matching, such as being computationally efficient, because no exhaustive search has to be performed among a dictionary. More importantly, it is able to predict MR parameters of unknown signals, whereas template matching approximates the MR parameters prediction to those present in the dictionary. This feature can limit quantization artefacts that can arise from dictionary approach. Depending on the task and the architecture, NN can require thousands of training data, which in case of MRI can be both expensive and time-consuming. However, for application to MRF the NN model can be trained using simulated data, which are easy to produce by using a simulator. Simulating the data is the original way in which dictionaries are produced in MRF, and the reliability of them has been widely demonstrated by the literature [1]. Few works already investigated the feasibility of applying Neural Networks to MRF, both with numerical simulations [12], [13] and both with phantoms and in-vivo acquisitions [14], [15]. The results of these works have demonstred the feasibility of NN approaches using both fully connected and convolutional neural networks. However, the focus of these works was primarily on demonstrating the feasibility of NN approach, and less attention has been given to which pipeline should be used to make the Neural Network model to learn an accurate and robust ITF for a given a MRF pulse sequence. For example, Virtue et al [12] created a training data set by sampling randomly the MR parameter space, whereas Hoppe et al [13] and Cohen et al [14] used a grid sampling. Moreover, different strategies have been applied for taking into account the noise, such as in [12], [13] where no noise was added in the training procedure, whereas in [14] white Gaussian noise with zero mean and 1% standard deviation was added during the training phase to promote robustness to noise.

The aim of this work, carried out by means of simulations, is to set and test different pipelines to apply Deep Neural Networks to MRF data to learn an accurate and robust ITF for a given MRF pulse sequence. In particular

this work investigates the generalization capability of NNs in relation to the method used to sample the parameter space, either uniform random sampling or grid sampling. Then, noise robustness is increased applying different data augmentation strategies, adding white Gaussian noise to MRF training signal examples. Once found a suitable pipeline to train the NN, a comparison between the NN approach and standard approach based on the dictionary matching in reconstructing brain quantitative maps is reported by means of a numerical simulations based on brain MR parameter values.

To show the ability of generalization of the NN approach, and how its performance scale with the number of predicted MR parameters in comparison with the dictionary approach, the NN has been applied to different MRF pulse sequences, i.e. IR balanced Steady State Precession (IR-bSSFP) [1], IR Fast Imaging with Steady state Precession (IR-FISP)[16] and its variant for B_1^+ estimation [6]. Moreover, a new MRF bSSFP sequence is here proposed to simultaneously estimate T_1, T_2, B_0 off-resonances and B_1^+ field inhomogeneities. This procedure differs from the one presented in [7], where the B_1^+ is measured with a standard MRI protocol independently from the MRF acquisition, and different MRF dictionaries encoding T_1, T_2 and B_0 are computed for each B_1^+ value.

The reported results may help the community working in MRF in evaluating advantages of deep learning approaches, pushing deep learning approaches and eventually take advantage of them, and pushing fingerprinting pulse sequences design to estimate other MR parameters, such as diffusion, without the limitation of dictionary sizes and of the accuracy of parameters prediction.

A. Background

Magnetic Resonance Fingerprinting has two main steps to be optimized: an MRI acquisition scheme, which involves pulse sequence design and k-space sampling, and a reconstruction algorithm able to map the acquired signal $S(t)$ to the MR parameter space (T_1, T_2,B_0, B_1^+, etc...). The latter step is mathematically well described by the concept of the Inverse Transfer Function.

Let us say $\mathbf{x}$ is the acquired MRF signal, with $\mathbf{x} \in \mathcal{R}^n$, where n is the size of the acquired signal, and let us say $\mathbf{p} \in \mathcal{R}^m$ to be a m-pla in the MR parameter space, where m is the number of parameters such as T_1, T_2, B_0 ect.

Applying a pulse sequence to a system characterized by MR parameters $\mathbf{p}$ means that exits a function $g(\mathbf{p})$ that maps $\mathbf{p}$ into the output signal $\mathbf{x}$. Quantitative mapping applies g^{-1} to the acquired signal $\mathbf{x}$ to obtain $\mathbf{p}$. Inverse mapping is actually strongly related to pulse sequences design; pulse sequences have usually been designed to obtain the magnetization evolution to be described by an analytic function of the MR parameters, that is to find an analytic form of $g(\mathbf{p})$. Examples are the Inversion Recovery sequence for T_1 mapping, and Carr-Purcell-Meiboom-Gill sequence for T_2 mapping, in which a mono-exponential model or a multi-exponential model are used to describe g, and for the multi-exponential case Inverse Laplace Transform is commonly used to obtain T_1 and T_2 distributions from the acquired signals [17][18].

In MRF, the pulse sequence is designed to make the magnetization evolution sensitive to multiple parameters at the same time using variable FAs and TRs patterns, making hard to model the transfer function $g(\mathbf{p})$ (and $g^{-1}(\mathbf{p})$ by consequence) with analytic functions. In such a scenario, template matching with a precomputed dictionary is a way to overcome this lack of knowledge. An outcomes of the transfer function can be simulated for a m-

pla of MR parameters $\mathbf{p}_i$ by using Bloch equations, so that an estimated $\hat{g}(\mathbf{p}_i)$ can be obtained. Hence, for a given MRF pulse sequence, a dictionary is built by sampling a set of MR parameters $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_l\}$, where $\mathbf{p} \in \mathcal{R}^m$. Then, the simulations of the Transfer Function related to that pulse sequence are stored into a dictionary $\mathcal{D} = \{\hat{g}(\mathbf{p}_1), \hat{g}(\mathbf{p}_2), ..., \hat{g}(\mathbf{p}_l)\}$, where l is the length of the dictionary and $\hat{g}(\mathbf{p}_i) \in \mathcal{R}^n$. When a real data is acquired from a voxel, the signal is described by $g(\mathbf{p}_{GT})$ where $\mathbf{p}_{GT}$ indicates the ground truth MR parameter vector characterizing the voxel tissue. Dictionary based approach maps $g(\mathbf{p}_{GT})$ into $\mathcal{D}$ by computing the nearest neighbor between $g(\mathbf{p}_{GT})$ and each $\hat{g}(\mathbf{p}_i)$ present in $\mathcal{D}$ according to a similarity measure, which usually is the dot product. The main limitation of this approach is related to the a priori design of the dictionary. The number of entries scales rapidly the more MR parameters are encoded into the dictionary according to the curse of dimensionality [19]. Adding a new parameter to be retrieved with MRF leads to an exponential growth of the number of entries to be inserted into the dictionary without affecting resolution. Since both computational and memory usage limitation has to be taken into account, the number of entries should be kept under control. This could increase sparsity into the dictionary, which can produce high biases in nearest neighbor algorithms [20], especially when they are applied to high dimensional data spaces, such as the case of MRF where usually signals are represented by a 1000 points vector.

With NN approach the limitations of MRF related to dictionaries can be strongly reduced. Indeed, given a training data set $\mathcal{T} = \{\hat{g}(\mathbf{p}_1), \hat{g}(\mathbf{p}_2), ..., \hat{g}(\mathbf{p}_h)\}$, built by Bloch equation simulations, a NN approximation of the inverse transfer function $g^{-1}(S(t))$, defined $\hat{g}^{-1}(S(t))$, can be learned [11].

Once trained, a MRF voxel signal $S(t)$ is the input of the NN model, which applies $\hat{g}^{-1}(S(t))$ to retrieve $\hat{\mathbf{p}}$, which is an estimation of $\mathbf{p}_{GT}$. Being a regression procedure, the model can correctly predict unknown examples, depending on how well it is trained. This limits the issues of the template matching approach. Moreover, a strong mathematical background exits behind these applications, and in 1989 Hornik demonstrated that standard multilayer feedforward networks are capable of approximating any measurable function to any desired degree of accuracy [11].

B. Background: state of the art of NN applied to MRF

To our knowledge four works investigated the application of neural networks to MR Fingerprinting. In this subsection the main characteristics of these works have been summarized to highlight the different strategies that have been used so far to apply neural networks to MRF.

- Virtue et al. [12] train both a real and complex valued neural networks to estimate T_1, T_2 and B_0 off-resonances from a 500 time points fingerprint given as input, produced by a IR-bSSFP sequence. In particular, in this implementation there are three neural networks, each with the same architecture, that process in parallel the input fingerprint to predict a MR parameter each. The neural networks have been trained with simulated data. The main focus of this paper is to investigate if a complex valued neural networks are better than real valued nets. Thus, the NN models have been implemented to be real valued and complex valued. the The MR parameter space has been sampled using a uniform random sampling. No noise has been added during the training, while white Gaussian noise with SNR = 40 dB has been added for testing the performance. The work has been carried out only with simulated data.

- Hoppe et al. [13] train a convolutional neural network (CNN) with 3 hidden layers for predicting T_1 and T_2 parameters given a 3000 time points fingerprint as input, produced by a IR-FISP sequence. The network have been trained with simulated data. The MR parameter space has been sampled using a grid sampling. No noise was added during the training neither during the test carried out to asses the performance.

- Cohen et al. [14] train a neural network composed by 2 fully connected hidden layers to predict T_1 and T_2 parameters. The pulse sequences used in this work are an optimized EPI-MRF pulse sequence [21], composed by 25 pulses, and IR-FISP pulse sequence with a sliding-window reconstruction [22] to reduce undersampling artefacts. The neural network has been trained with simulated data, and the MR parameter space has been sampled in a grid way. White Gaussian noise with zero mean and 1% standard deviation was added during the training procedure. The work presents tests both on simulated data and on real data.

- Balsiger et al. [15] train a CNN to predict T_1, T_2 and PD using an optimized IR-FISP sequence and giving a 4D spatio-temporal patch as input. In particular, the authors want to take advantage of the neighbor pixels during reconstruction to avoid noisy reconstructed parameter maps. With this aim they design the NN application to process 5×5 patches in order to predict the MR parameter of the fingerprint in the center of the patch. The training phase is carried out using real data: six brains where scanned with MRF as well with standard protocols to obtain ground truth maps. A leave-one-out cross-validation has been used to train and test the network performance.

The aim of our work is to further investigate the application of NN to perform MRF parameter estimation. For this reasons, both IR-FISP and IR-bSSFP MRF sequences have been tested, as well different training strategies to train the NN models. In particular an accurate study regarding how adding noise during training can affect the generalization performance has been carried out using white Gaussian noise as source of noise. Trainig and test phases have been carried out with simulated data.

The NN approach in the present study is used to perform a pixel wise parameter prediction given a MRF signal. Hence, no convolutional layers have been inserted in the NN models. Convolutional neural networks are meant to take advantage of local connections and correlations in the processed signals [23]. Moreover, CNN learns filters, meaning that the networks learn to identify a feature independently by its positions within the signal. In case of MRF pixel-wise processing, those characteristics are less meaningful, while are clearly significant for the method used in [15].

II. METHODS

A. MRF pulse sequences and simulations

The simulations have been performed using four MRF pulse sequences: the MRF IR-FISP pulse sequence as described in [16], which encodes T_1 and T_2, and its variant which accounts for B_1 inhomogeneity adding flip angles abrupt changes as described in [6], labeled IR-FISP B_1; the MRF IR-bSSFP as described in the original MRF article [1], which encodes T_1, T_2 and B_0 off resonance, and a new modification here proposed, labeled IR-bSSFP B_1, to take into account for B_1 field inhomogeneities. The latter sequence consists in adding abrupt changes at the end

of the IR-bSSFP sequence, following the same criterion used for the IR-FISP B_1 sequence. Flip angle and time repetition patterns are depicted in Fig. 1.

The simulations have been carried out using MATLAB (MathWorks), where Block equations have been used for IR-bSSFP type sequences, whereas the Extended Phase Graph (EPG) algorithm [24] has been used for IR-FISP type sequences simulations.
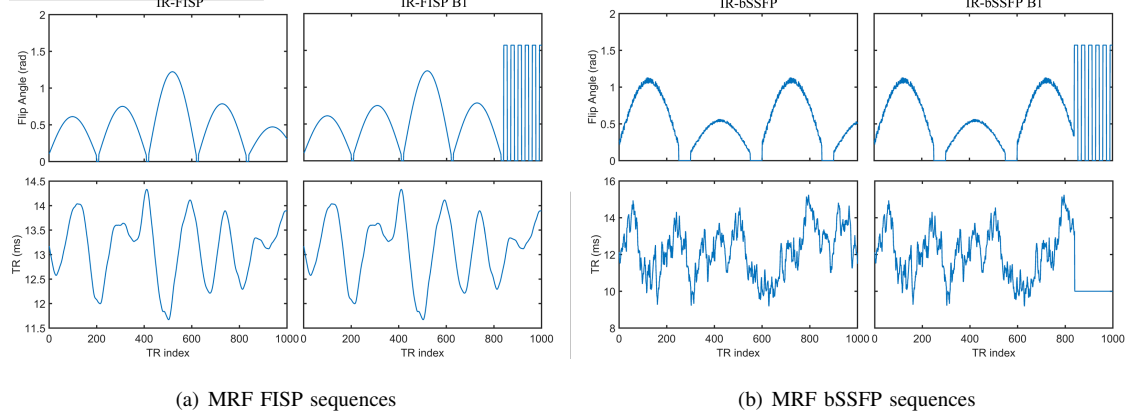

(a) MRF FISP sequences   (b) MRF bSSFP sequences

Fig. 1: MRF pulse sequences used for generating the synthetic signals.

B. Noise

In this work, noise affecting MRF signals has been considered as complex white Gaussian noise where the variance of the noise is expressed in terms of Signal to Noise Ratio (SNR). SNR has been defined in two ways as reported in equations 1 and 2, where P_{signal} and P_{noise} represent the power of the MRF signal and the noise, respectively, and A_{signal} represents the signal intensity of the MRF signal and σ_{noise} represents the standard deviation of the noise. It is worthy to point out that equation 2 expresses SNR in dB, but in this work all SNRs are expressed in the linear scale by applying the proper conversion. Which definition of SNR is used during the different experiments of this work is explitly pointed out in the corresponding sections.

$$SNR = \frac{\sigma^2_{signal}}{\sigma^2_{noise}} = \frac{P_{signal}}{P_{noise}} \tag{1}$$

$$SNR = 20 \times \log_{10}\left(\frac{A_{signal}}{\sigma_{noise}}\right) \tag{2}$$

White Gaussian noise in MRI simulation is used as the most common noise distribution, and, as shown by Griswald et al [8], severe k-space under-sampling artifacts in MR Fingerprinting can be well described by Gaussian noise with high variance. Being the simulated MRF signal a complex signal, Gaussian noise is added both to the real and imaginary part of it by assigning half of the noise variance per each channel. This indeed simulates a two channels acquisition, and preserves the Rician distribution when the magnitude of the signal is considered [25].

C. Feed Forward Neural Network models

The Deep Neural Network application has been developed using the Python package Keras with TensorFlow [26] backend.
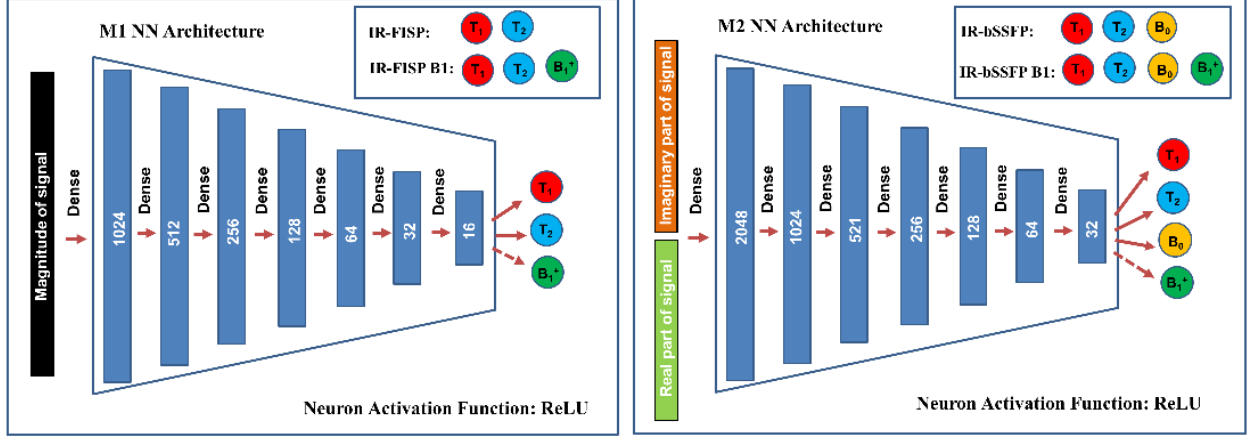

Fig. 2: NN architectures for model M1 (left box) and model M2 (right box). The two models are fully connected NN with ReLU used as neuron activation function for hidden layers, while linear activation function has been used for the output layer. Blue boxes represents layers and the numbers inside indicate the number of neurons. Model M1 has been used for IR-FISP and IR-FISP B_1 pulse sequences, whereas model M2 has been used for IR-bSSFP and IR-bSSFP B_1 sequences.

Two NN models have been defined, one architecture to process IR-FISP data and another architecture to process IR-bSSFP data. All of them are feed forward nets with 9 fully connected layers, and the Rectified Linear Unit (ReLU) has been used as activation function for the neurons in the firsts 8 layers, while a linear activation function has been chosen for the output layer. The NN architectures are reported in Fig.2 and described below by indicating the number of neurons per layer:

- M1: 1000 (input layer) - 1024 - 512 - 256 - 128 - 64 - 32 - 16 - 2 or 3 (output layer);
- M2: 2000 (input layer)- 2048 - 1024 - 512 - 256 - 128 - 64 - 32 - 3 or 4 (output layer).

M1 takes the magnitude of the complex MR signal produced by a IR-FISP or IR-FISP B_1 sequence as input, while M2, which is asked to estimate B_0, takes in input the concatenated real and imaginary parts of the complex MR signal produced by a IR-bSSFP and IR-bSSFP B_1 sequence. The output layer sizes match the number of MR parameters one wants to retrieve from the prediction, and they are:

- IR-FISP: T_1 and T_2;
- IR-FISP B_1: T_1, T_2 and B_1^+;
- IR-bSSFP: T_1, T_2 and B_0;
- IR-bSSFP B_1: T_1, T_2, B_0 and B_1^+.

where B_0 indicates stationary field off-resonances, and B_1^+ indicates B_1 excitation field inhomogeneities expressed in terms of correction ratio, being 1 the nominal flip angle excitation.

It is important to point out some of the motivations that have driven the design of the NN model architectures. A bottleneck shape has been selected for two main reasons. The first is avoiding exploding number of model parameters: many layers bring to overfitting, since fully connected layers with large number of neurons increase rapidly the number of model parameters. The second is that a bottleneck shape forces the network to encode more meaningful representations the deeper is the layer considered [23].

Preliminary experiments, not included in this work, where different network sizes, batch sizes and learning rates have been tested have guided to the used architectures. In particular, they showed that the architecture used for IR-bSSFPs sequences works well even for IR-FISPs sequences, whereas the vice-versa does not hold. However, because the smaller is the size of the network, the faster is the computational time, the smallest architecture has been selected for IR-FISPs sequences.

The training procedure was supervised, using the Mean Squared Error (MSE) between NN estimated parameters and ground truth parameters as loss function, and the Adam algorithm [27] has been used for model weights optimization. In particular, 500 epochs with 1000 gradient steps for each epoch have been used, with a fixed batch size of 500. Initial learning rates of 3×10^{-5}, for model M1, and 1×10^{-4}, for model M2, have been used. The application has been run on a cluster with 16 dual cores CPUs.

D. Training strategies: training sets, test sets and data augmentation

1) Parameter space sampling, random uniform and grid sampling: to assess how well the network models learn the ITF depending on the training set distribution, for each pulse sequence two training sets of the same size were generated: a random uniform set, in which the parameter space is sampled using a random uniform distribution; a gridded set, where the parameter space is sampled with a fixed mesh grid.

Table I summarizes the training set characteristics, where the label *R* refers to the random sampling, whereas label *G* refers to grid sampling.

2) Data augmentation and preprocessing: three data augmentation strategies have been tested by training the network models with different noise adding procedures:

- *W/O Noise*: using no data augmentation during the training, which means that only noise-free examples are fed to the network models, as in reference [12];

- *Fixed variance*: feeding the networks with noisy inputs affected by noise with 1% standard deviation, equals to a variance of 10^{-4}, as in reference [14]. This means that the network during the training sees just one fixed SNR given a set of MR parameters;

- *Variable variance*: feeding the networks with inputs affected by noise with different variances, which are expressed in terms of SNRs. For each training batch, a vector of SNRs is generated by randomly sampling the SNR values in the range 2 to 100 so that the batch contains data with different SNRs, and the variances of the noises to add to the batch examples have been estimated using equation 1.

The data augmentation step is done on-line. Hence, given a training set, no new data have to be stored neither in hard neither in flash memories to carry out the three data augmentation strategies. After the data augmentation step, each input is then normalized to have norm equals to 1 to generalize the NN model.

Model	Sequence	Set Label	T_1 (ms)	T_2 (ms)	B_0 (Hz)	B_1^+	Count
M1	IR-FISP	R1	[10 ÷ 4000]	[1 ÷ 3000]	0	1	75 555
		G1	[10 ÷ 4000]*	[1 ÷ 3000]*	0	1	
M1	IR-FISP B_1	R2	[10 ÷ 4000]	[1 ÷ 3000]	0	[0.5 ÷ 1.5]	164 475
		G2	[10 ÷ 4000]**	[1 ÷ 3000]**	0	[0.5 ÷ 1.5]**	
M2	IR-bSSFP	R3	[10 ÷ 4000]	[1 ÷ 3000]	[-400 ÷ 400]	1	175 041
		G3	[10 ÷ 4000]***	[1 ÷ 3000]]***	[-400 ÷ 400]***	1	
M2	IR-bSSFP B_1	R4	[10 ÷ 4000]	[1 ÷ 3000]	[-400 ÷ 400]	[0.5 ÷ 1.5]	396 550
		G4	[10 ÷ 4000]****	[1 ÷ 3000]****	[-400 ÷ 400]****	[0.5 ÷ 1.5]****	

TABLE I: Summary of the training set used to train the NN models. For G sets the parameter spaces were sampled using the following grids. *) T_1 and T_2 where incremented with steps of 10 ms; **) T_1 = [10:10:800, 850:50:1000, 1100:100:2000, 2500:500:4000] ms, T_2 = [1, 10:10:300, 350:50:1000, 600:100:1000, 1500:500:3000] ms and B_1^+ = [0.5:0.02:1.5]; ***) T_1 = [10, 20:20:500, 600:50:1000, 1100:100:2000, 2500:500:4000] ms, T_2 = [1, 10:10:500, 550:50:1000, 600:100:1000, 1500:500:3000] ms and B_0 = [-400:50:150, -100:10:-70, -60:2:60, 70:10:100, 150:50:400] Hz; ****) T_1 = [10, 20:20:300, 350:50:500, 600:100:1000, 1250:250:2000, 2500:500:4000], T_2 = [1, 10:10:300, 350:50:1000, 600:100:1000, 1500:500:3000] ms. B_0 = [-400:50:150, -100:10:-70, -60:2:60, 70:10:100, 150:50:400] Hz and B_1^+ = [0.5:0.1:1.5].

For each model, the performance in predicting a MR parameter has been estimated in terms of Mean Absolute Percentage Error (MAPE) of the predicted parameter, evaluated on a test set composed by 30,000 fingerprints generated using the same pulse sequence as in training procedure, and sampling the parameter space with a random uniform distribution. To test noise robustness, the described prediction procedure has been repeated giving to each model data with different SNRs = [3, 5, 10, 20, 30, 40, 50, 60, 70 ,80, 90, 100]. Moreover, to assess the variance of the MAPE as a function of the SNR, for each noise level the operation has been repeated 10 times and the standard deviations of the MAPEs has been considered as measure of variance. The MAPE is computed following equation (3), where $\widehat{p_k}$ is the estimated parameter value while p_k is the ground truth parameter value and N is the number of considered examples.

Other error measurements used in this work are the absolute error (AE) and the root mean squared error (RMSE) defined in equations 4 and 5 respectively.

$$MAPE(\%) = \sum_{k=1}^{N} |\frac{\widehat{p_k} - p_k}{p_k}| \times 100 \tag{3}$$

$$AE = \widehat{p_k} - p_k \tag{4}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{p_k} - p_k\right)^2} \tag{5}$$

E. Numerical Brain Phantom Simulations

Realistic T_1, T_2, B_0 and B_1^+ maps where obtained by processing real acquisitions downloaded from the Multi-Modal MRI Reproducibility Resource repository. Landman et al [28] performed MRI acquisitions with standard quantitative protocols at 3 T: variable flip angle (VFA) imaging for T_1 mapping; double echo time imaging for T_2 mapping; two sequential 2D gradient echo with different echo times for B_0 mapping and Actual Flip-Angle Imaging (AFI) for B_1^+ mapping. The in vivo brain images acquired with these protocols have been then processed with MATLAB to obtain the quantitative maps. In particular, the qMRLab software [29] has been used to process the VFA applying B_1^+ correction, since B_1^+ is known to be a confounding factor to estimate the correct T_1 in VFA imaging [30]. Once obtained, these quantitative maps, reported in Fig. 3, have been used as ground truth to simulate, pixel wise, the MRF acquisition with the four pulse sequences described in section II-A. Complex white Gaussian noise was added to the simulated data using the SNR defined in equation 2 where, in this case, A_{singal} indicates the average signal intensity within a region of the white matter in the first image of the MRF time series.

It is worthy to point out that all the MR parameters have been used to simulate the signals. Hence, the B_1^+ map has been taken into account both for all the four sequences; in the IR-FISP and IR-FISP B_1 the B_0 has not be considered because not encoded, while in the IR-bSSFP sequences the B_0 is encoded and thus it has been considered in the simulations. A synthetic B_1^+ has also been created with a Gaussian shaped profile exploring a wider range of B_1 inhomogeneities as can be found commonly in acquisition with 7 T MRI scanners [6].

For each pulse sequence, the parameter maps have been reconstructed by processing the MRF data with both the trained NN models and using the usual dictionary method based on the dot product. For this latter algorithm, the data sets G1, G2, G3 and G4 have been used as dictionaries for the corresponding pulse sequences. The MAPE and RMSE have been used as global measure of reconstruction quality, whereas the AE is used as local measure. To have a complete overview of the performance all three errors have to be assessed. MAPE gives information about the mean relative error done in parameter estimation, which has the advantage of giving an immediate sense of the global performance, however it could be misleading when very small values are taken into account, such in the case of B_0 off-resonances, because small absolute errors give high relative errors. In such cases, a more reliable global measure is the RMSE, which expresses a global error in the same unit of measure of the considered parameter. It is also interesting to check a measure of the local error as the AE, because can be used to assess if the error has a noise-like distribution in the image of it shows strucutres.

III. RESULTS AND DISCUSSION

A. Training data distribution: Grid Vs Random sampling

The results of the training phases of the NN models, fed with noiseless examples with random and grid sampling are summarized in Fig. 4, where training and test losses are plotted as function of the training epoch.
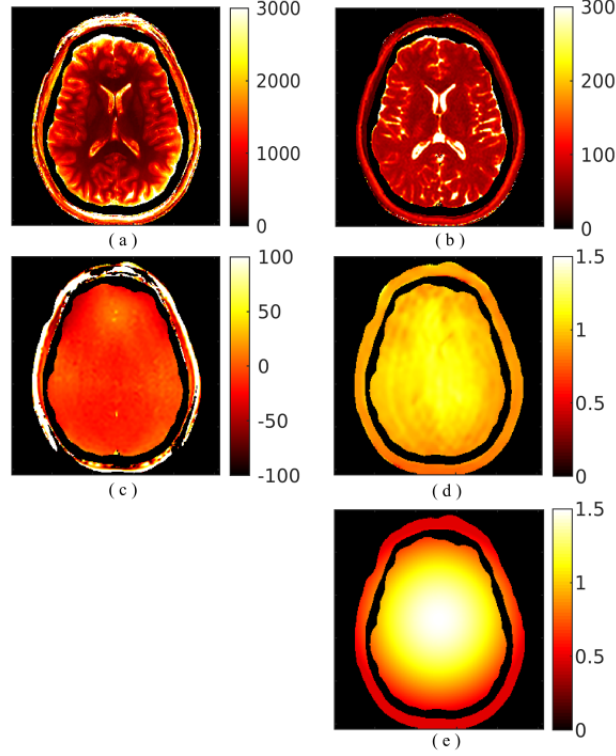
Fig. 3: Quantitative brain maps computed from the Multi-Modal MRI Reproducibility Resource: (a) T_1 (ms), where T_1 values from 3000 ms to 4000 ms are showed as equal to 3000 ms); (b) T_2 (ms), where T_2 values from 300 ms to 2500 ms are showed as equal to 300 ms; (c) B_0 (Hz), where B_0 values less than $|500|$ Hz are showed as equal to $|100|$ Hz ; (d) B_1^+; (e) synthetic B_1^+ (Gaussian shaped).

For model M1, which processes IR-FISPs type data, there is a strong overlap of the training and test losses as clearly visible looking at Fig. 4 (a) and (b), indicating an overall well training phase. On the contrary, for model M2, which processes IR-bSSFPs type data, the grid training losses are about one order of magnitude lower than the random counterparts, but test losses do not improve during the NN training as clearly visible looking at Fig. 4 (c) and (d), indicating the NN is overfitting the training data set. Hence, model M2, trained with examples sampled in a grid way, overfits the training set. In figure 5 the ground truth MR parameters of the test set against those predicted by the NN model, when trained with randomly and grid sampled IR-bSSFP examples, are shown to appreciate the differences in the performance in estimating the MR parameters. Lin's concordance coefficient [31] between predicted and GT values has been reported as quantitative measure of agreement. The same kind of results holds for IR-bSSFP B_1 sequence (not shown in the work).

Although the grid is a reasonable way to built dictionaries to be used with the classical approach of MRF, indeed it gives information about the accuracy one can expect from the matching algorithm, it is not well performing in train NN in case of IR-bSSFP type sequences (ρ_c is always less than 0.5). The grid introduces a strong and wrong *a priori* about the real world, which can affect the generalization capacity of the NN, as in the case of IR-bSSFP
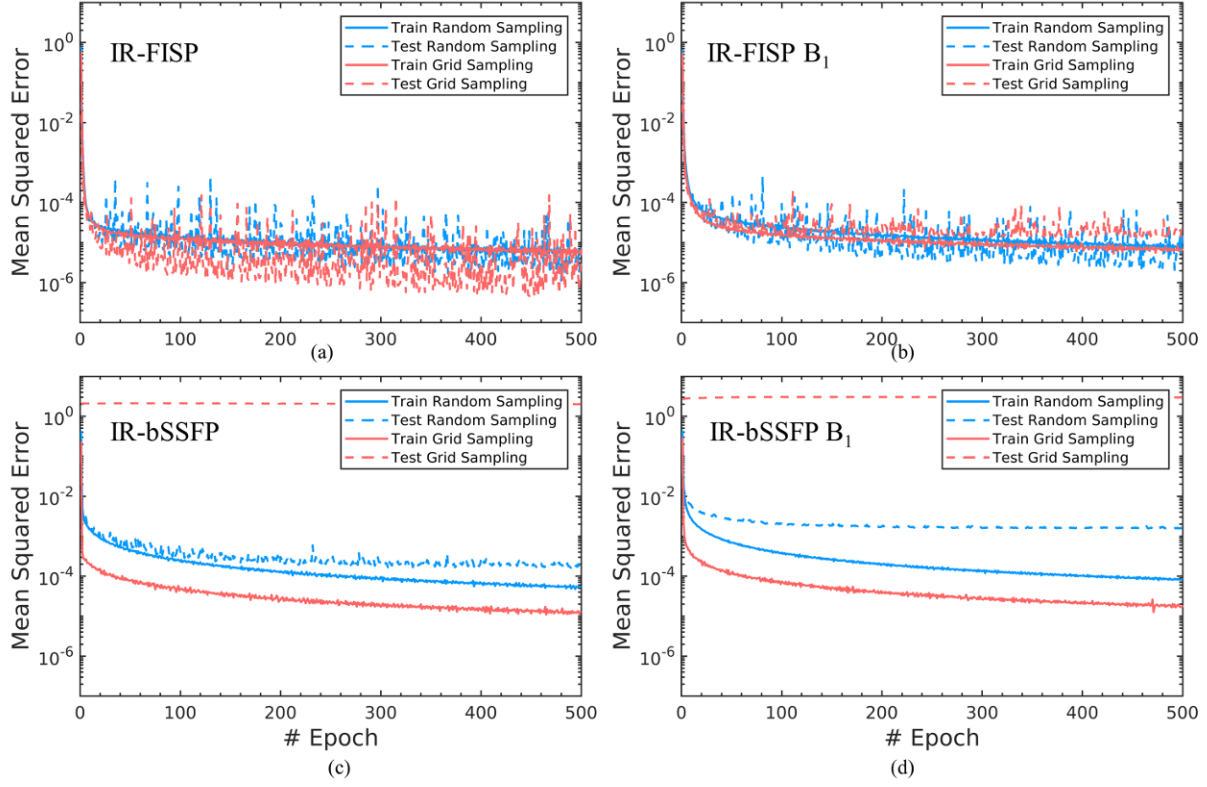
Fig. 4: Comparison of the training and test loss functions for the four NN models trained with random sampled and grid sampled training data sets.

type pulse sequences.

B. Noise robustness: learning less to learn better

The training and test losses of the NN models trained with different data augmentation strategies are reported in Fig. 6. It is worthy to notice that for each model, the test loss, where the MSE is used as loss function, is checked at the end of each epoch by asking the model to predict the MR parameters for few never seen examples which however have the same noise adding strategy set for the training step. The results showed in Fig. 6 demonstrate that the best training is reached when the W/O noise data augmentation strategy is used, since both training and test losses reach the lowest values at the end of training, whereas the worst condition for both training and test losses at the end of the 500 training epochs is when the *Variable variance* strategy is used. Moreover, looking at the gap between training and test losses, model M1 does not have gaps between training and test loss regardless to the data augmentation strategy used, while model M2 always has a gap between the two losses, indicating overfitting.

The MAPE of parameters evaluated on test sets for different data augmentation strategies as a function of the SNR are reported in Fig. 7. The error bar indicates two standard deviations of the MAPE, which comes from 10 repetitions of the prediction procedure on noisy data. Although Fig. 6 showed *Variable variance* to be the worst performing strategy during training, it is the most robust to noise among the three. Looking at Fig. 7 one can
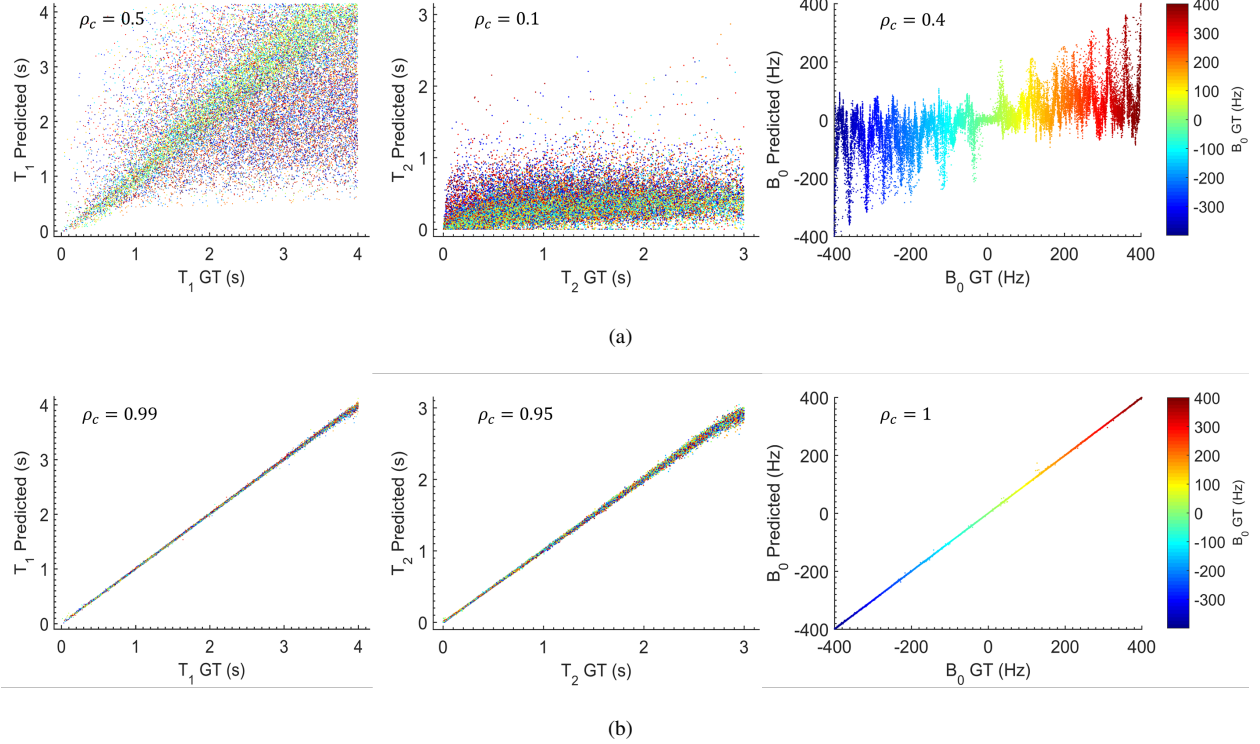
Fig. 5: Predicted MR parameters against Ground Truth (GT) parameters using NN model M2 trained with IR-bSSFP MRF examples taken from sets G3 (grid sampling) and R3 (random sampling). Lin's concordance coefficients are also reported as quantitative measures of agreement. Note that the reported predictions refer to test data without added noise.

appreciate the significant improvement in high noise level robustness using this latter strategy for T_2 parameter in particular. For all the models, T_2 MAPE never goes over 12% and decreases rapidly under 10%, while MAPE for the other parameters is always lower than 5%. The explanation for this behavior is straightforward: with the *Variable variance* strategy, the NNs get to see more diverse data, and improve generalization. On the contrary, with the *Fixed variance* strategy the model eventually overfits for data with high SNRs, and, as expected, with the *W/O Noise* strategy the models perform the worst in terms of noise robustness. Moreover, since the noise adding happens during the training, and no new data are stored neither in hard and flash memory the *Variable variance* strategy does not affect memory usage efficiency. Finally, the standard deviation values are in the order of 10^{-2}, which make them barley visible in the plots, indicating all models show low variance for experiment repetitions.

Figure 7 shows that for IR-FISP type sequences there is no bias between training and test losses, whereas there is a consistent bias between them for IR-bSSFP type sequences. In general a difference in training and test losses is expectable, in this case we are testing the models with data coming from the same distribution, although they have never been taken into account for weights optimization during the training phase. Hence, one could expect training and test losses decrease closely together. A behavior like the one enhanced here means model M2 is overfitting the
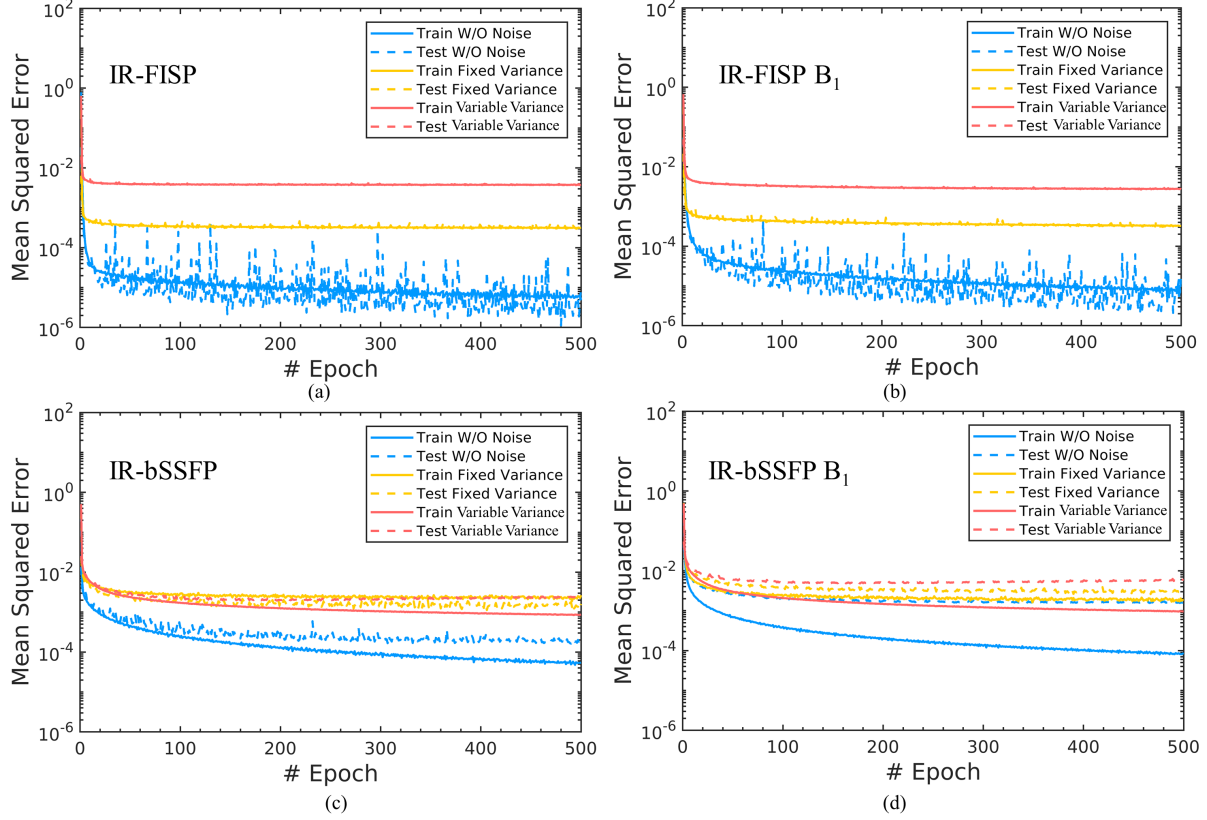
Fig. 6: Training and test loss functions for the NN models trained with different data augmentation strategies: without noise, *Fixed variance* and *Variable variance*.

training sets and it is likely due to the size of R3 and R4 training sets. Because the training data are simulated data, a straightforward way to test this hypothesis is to increase the training set size for IR-bSSFP and IR-bSSFP B_1 pulse sequences. A new data set of 1,000,000 examples has been created using the IR-bSSFP B_1 sequence. The 97% of it has been used as training set, while the 3% as test set for checking test loss during the training phase, which has been performed with the *Variable variance* data augmentation strategy. In Figure 8 the comparison between the training and test losses evaluated during the training of NN model M2 with 396 550 and 970 000 examples is reported. Although the training loss worsens when model M2 is trained with more examples, the gap between training and test loss is considerably reduced and the test loss reaches a lower value than the case in which the model is trained with fewer examples. Since the test loss is an estimator of the generalization performance of the NN model, it is possible to observe that the generalization of M2 model is increased when more training examples are added.

C. Brain maps reconstruction: Neural Networks vs Dictionaries

In figure 9 and Fig. 10 the absolute error maps for each parameter relative to NN reconstruction and the dictionary matching are reported respectively. They have been evaluated by computing, pixelwise and for each parameter map,
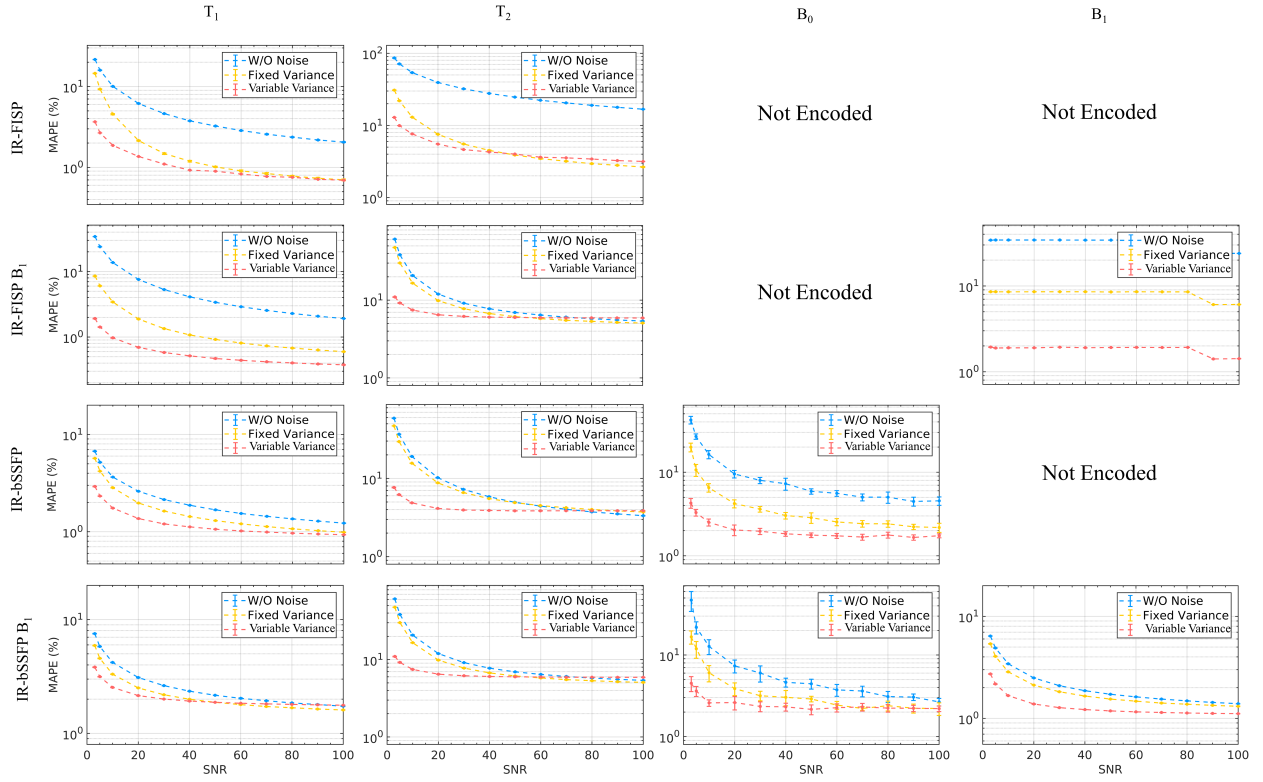
Fig. 7: Comparison of the parameter mean absolute percentage errors (MAPEs) evaluated on test sets for the NN models trained with different data augmentation strategies (best view on a screen).
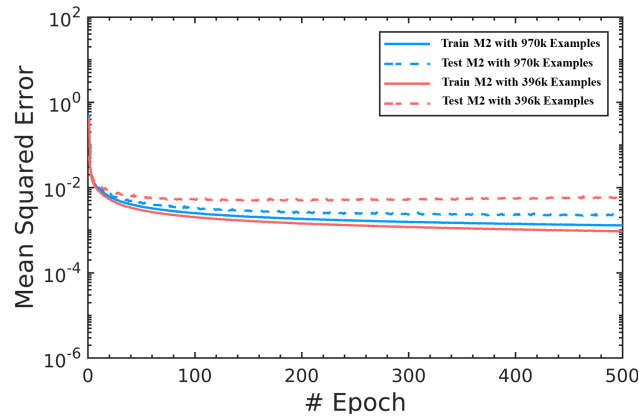

Fig. 8: Training and test loss functions for the NN model M2 trained with 396 550 training examples and 970 000 training examples.
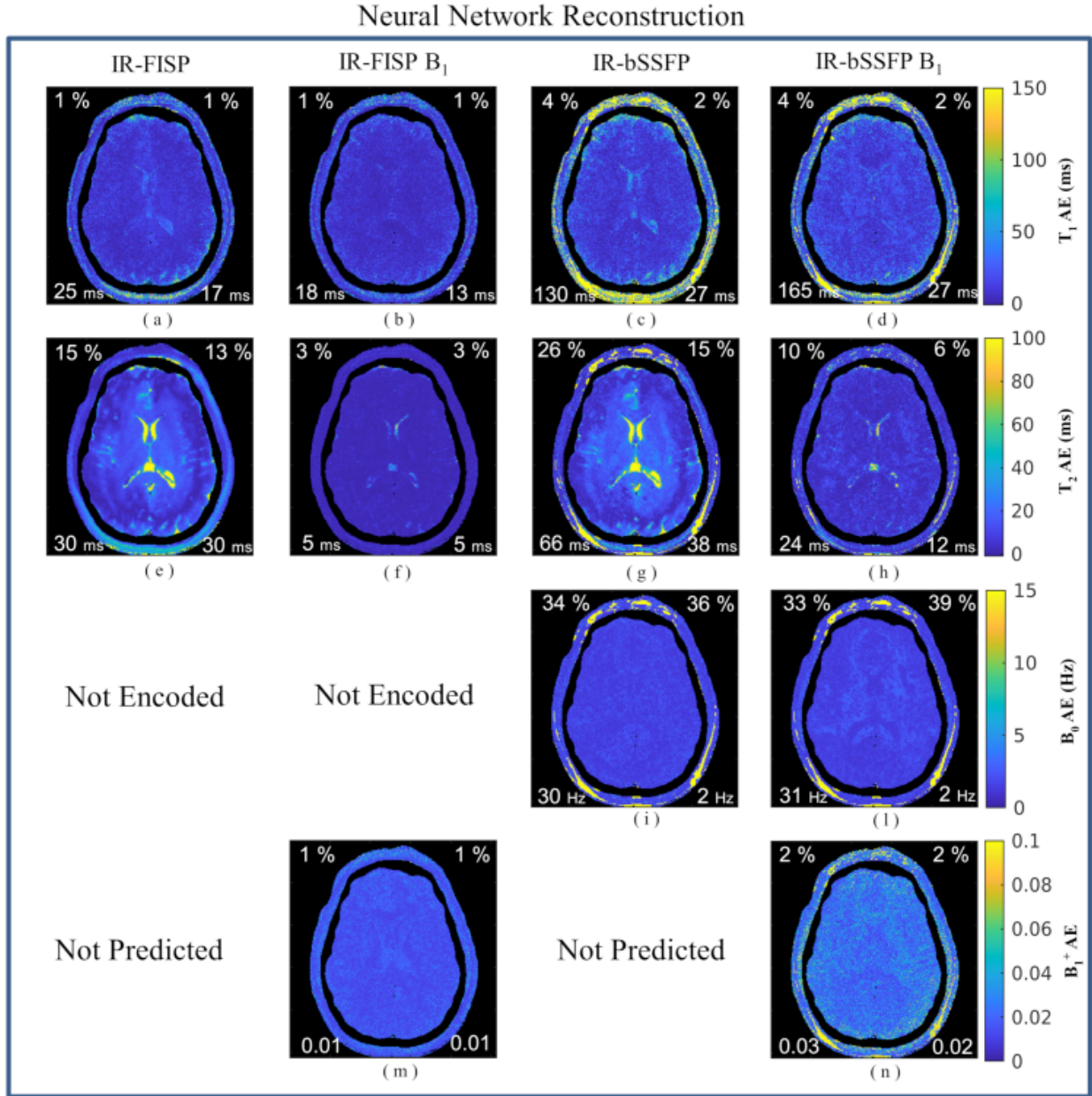
Fig. 9: Absolute error maps between NN reconstructed and ground truth parameter maps. Global error estimators are reported on the four corners of each image: upper left corner reports the MAPE; upper right corner reports MAPE evaluated without considering the scalp region; lower left corner reports the RMSE while lower right corner reports RMSE evaluated without considering the scalp region.
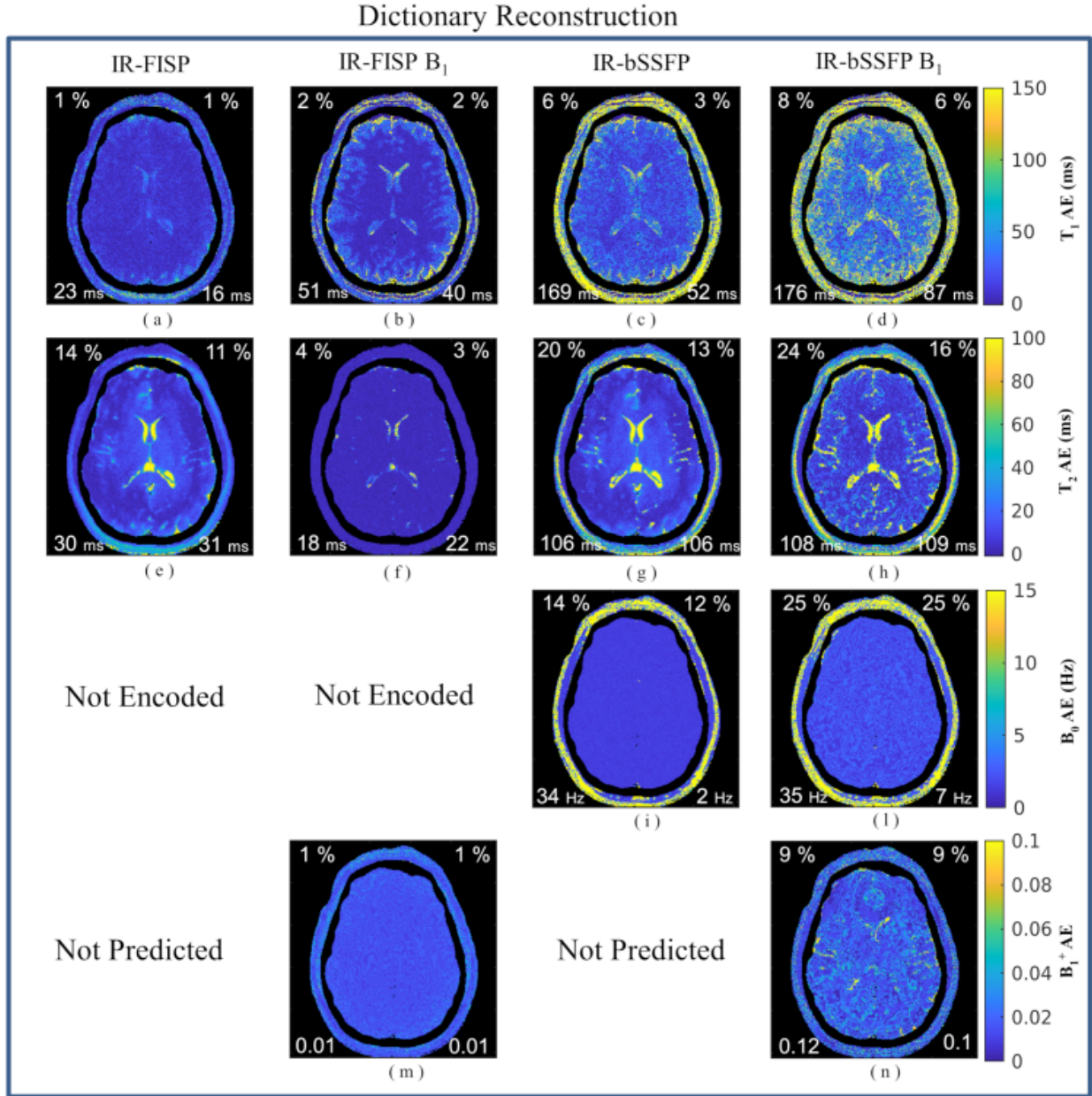
Fig. 10: Absolute error maps between dictionary reconstructed and ground truth parameter maps. Global error estimators are reported on the four corners of each image: upper left corner reports the MAPE; upper right corner reports MAPE evaluated without considering the scalp region; lower left corner reports the RMSE while lower right corner reports RMSE evaluated without considering the scalp region

the absolute error between the ground-truth parameter value and the predicted value. It is worthy to re-mark that the reconstruction has been performed with noisy data with SNR equals to 5. For each map, four global error estimators have been reported on the corners of the images:

- upper left corner reports the MAPE between the ground-truth pixel values and those reconstructed;
- upper right corner reports the MAPE between the ground-truth pixel values and those reconstructed, considering the brain without the scalp;
- lower left corner reports the RMSE between the ground-truth pixel values and those reconstructed;
- lower right corner reports the RMSE between the ground-truth pixel values and those reconstructed, considering the brain without the scalp.

Overall the NN approach performs better than the dictionary approach. This is particularly evident considering the IR-bSSFP B_1 sequence, where all the error estimators are lower than those characterizing the dictionary matching. This difference is even more evident if one looks at the error estimators evaluated without considering the scalp, where the B_0 off resonances go up to +/- 500 Hz, which are out of the range used for training the NN model M2, and, as one would expect, this affects parameters estimation accuracy. The scalp region is of poor clinical interest as most of the information is in the brain region. For this reason, the global error estimators have been computed also without considering the scalp. In the latter case the error estimators show a better agreement with ground truth values even if the SNR is very low. It is worthy to point out that MAPE is a poor measure of quality for B_0 estimation, because small absolute errors made by the NN models are high percentage errors, as highlighted by looking at the RMSEs, which are in the order of 2 Hz.

Considering the IR-FISP the two approaches have similar performance probably because the number of the examples used for build the dictionaries was high enough to ensure a proper MR space sampling also for the dictionary approach. For the IR-FISP B_1, instead, the NN approach performs better in estimating T_1 and T_2. MAPEs and RMSEs evaluated using the NN recontruction are less or equal than MAPE and RMSE evaulated using dictionary reconstruction. Analyzing more in detail Fig. 9, one can appreciate how not taking into account for B_1^+ field inhomogeneities drives to drop in accuracy in parameter estimation (Fig. 9e, Fig. 9g, Fig. 10w and 10g), being T_2 the parameter mostly affected. In particular, T_2 MAPE increases from 2.5% to 13% (Fig. 9f and Fig. 9e respectively), when B_1^+ is not taken into account during network training for IR-FISP sequences. The same drop in T_2 accuracy happens for IR-bSSFP sequences (Fig. 9g and Fig. 9h). Considering Fig. 10, one can conclude that taking into account for B_1^+ field inhomogeneities does not improve parameter estimations for IR-bSSFP sequences. On the contrary, errors are higher when B_1^+ field inhomogeneities are predicted in the dictionary, a trend that is opposite to what obtained using NN approach. This behaviour is because the number of entries of the dictionary is not enough to guarantee an accurate sampling of the 4D MR parameter space. It is an example of the curse of dimensionality of nearest neighbor algorithms.

All these considerations are reinforced by looking at Fig. 11 and Fig. 12, where the same measurements have been repeated while considering not the original B_1^+ map, but the synthetic Gaussian shaped B_1^+ map. IR-bSSFP B_1 sequence is a clear example that highlights the difference between NN and dictionary approaches when more

than two or three parameters are encoded in the simulations. Indeed, even if the same number of examples have been given to the two approaches, the dictionary method suffers from the coarse resolution of the MR parameter space sampling, whereas the NN, trained with the same number of examples, is able to predict parameter values with relative high accuracy.

The NN reconstruction lasts about 1.5 s for IR-FISP type sequences and 4 s for IR-bSSFP type sequences to elaborate 35,000 pixels. No significant time increase has been observed when estimating 2 or 3 parameters with the IR-FISP type sequences and 3 or 4 parameters with the IR-bSSFP type sequences. This is because the only difference in NN models for IR-FISP type sequences, and NN for IR-bSSFP type sequences, is in the output layers, and this just adds a small number of operations. On the contrary, with dictionary matching the computational time strongly scales with the number of parameters, because the size of the dictionary increases, being 10 s, 17 s, 77 s and 160 s for IR-FISP, IR-FISP B_1, IR-bSSFP and IR-bSSFP B_1 respectively. Overall the NN reconstruction is more efficient both in memory usage, since it does not need to store any dictionary in the RAM memory, both in computational time.

IV. CONCLUSIONS

In this work a deep neural network applied to MRF data has been proposed, tested both for prediction accuracy and noise robustness by means of simulated data. Intrinsic limitations of dictionary-based approaches for MRF, and how NNs approaches can overcome them, have been discussed in details. In particular, it has been studied how to better design parameter space sampling and how to achieve better noise robustness during NN training. To demonstrate the generalization of the proposed NN, four different MRF pulse sequences have been considered: the IR-FISP sequence and its variant to account for B_1^+ field inhomogeneities, the original IR-bSSFP and a newly proposed variant to account for B_1^+ field inhomogeneities. The results have demonstrated the low accuracy of grid sampling to build training data sets, since this method introduces a strong bias due to its regularity, and this leads the NN to overfit, as shown for IR-bSSFP type sequences. In fact, random uniform sampling has been found to perform better in spanning the parameter space for the IR-bSSFP type sequence, whereas not significant differences have been found for the IR-FISP type sequence between grid and random sampling.

Then, three different data augmentation strategies have been tested in order to promote robustness to white Gaussian noise. Feeding the network models with noisy data with different noise variances have yielded the best results in terms of noise robustness providing a MAPE lower than 15% (evaluated on a wide range of parameter values) even in case of severe noisy data (SNR = 3). These results exceed those obtained with the data augmentation strategies used in [12] and [14], which have also been tested.

Finally, a comparison between the NN and the dictionary approaches has been performed using a numerical brain phantom built from real standard quantitative MRI protocols. The results of this comparison have shown that NN performance is greater or equal to those of dictionary-based approach, depending on the pulse sequence.

A neural network model learns an approximation of the inverse transfer function from a set of training examples. Once trained, the processing is computationally fast and efficient, not requiring a large memory usage for dictionary storage. To increase parameter estimation accuracy is important to take into account as many scanner artifacts as
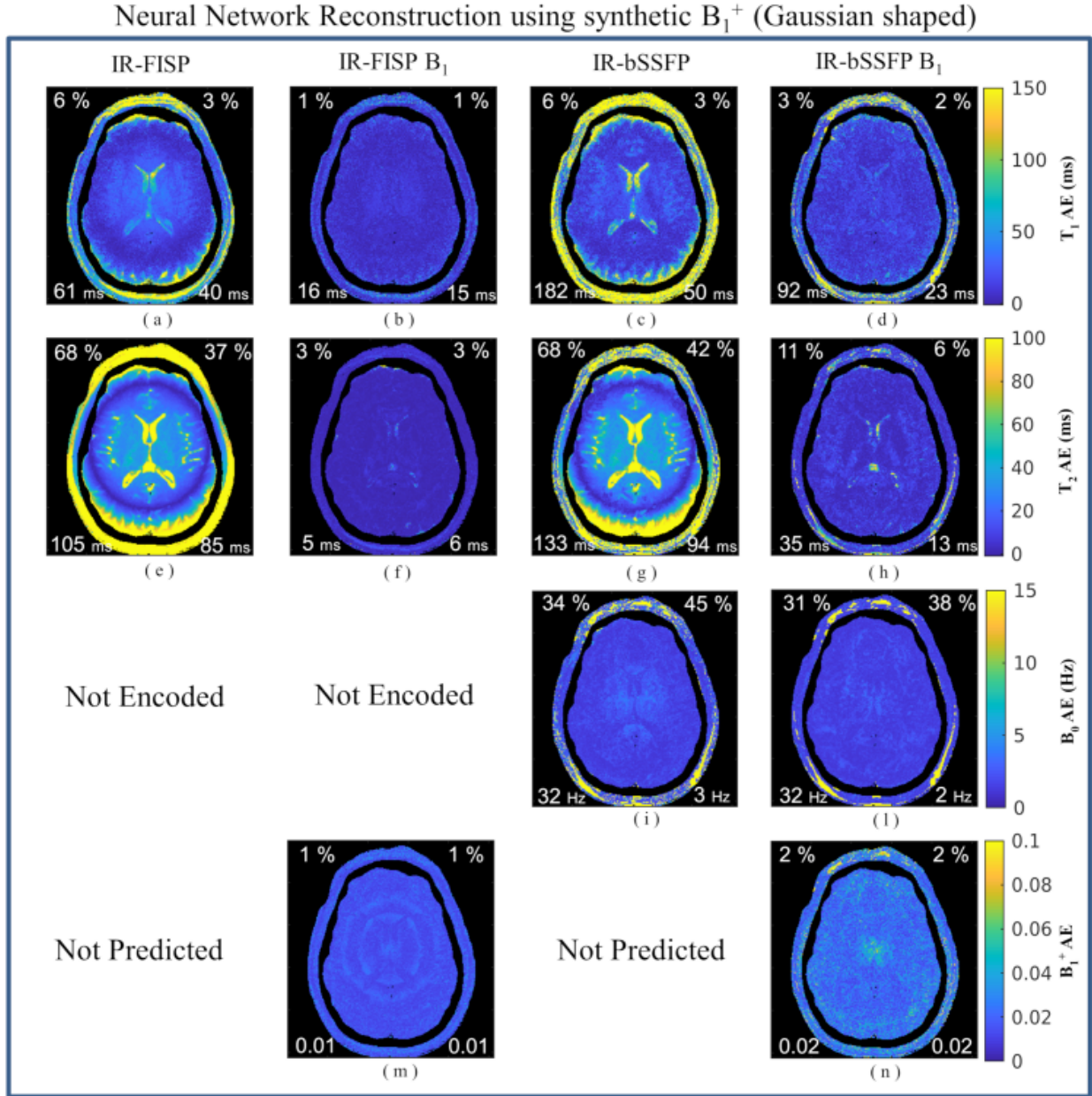
Fig. 11: Absolute error maps between NN reconstructed and ground truth parameter maps with Gaussian shaped B_1^+ field inhomogeneities. Global error estimators are reported on the four corners of each image: upper left corner reports the MAPE; upper right corner reports MAPE evaluated without considering the scalp region; lower left corner reports the RMSE while lower right corner reports RMSE evaluated without considering the scalp region.
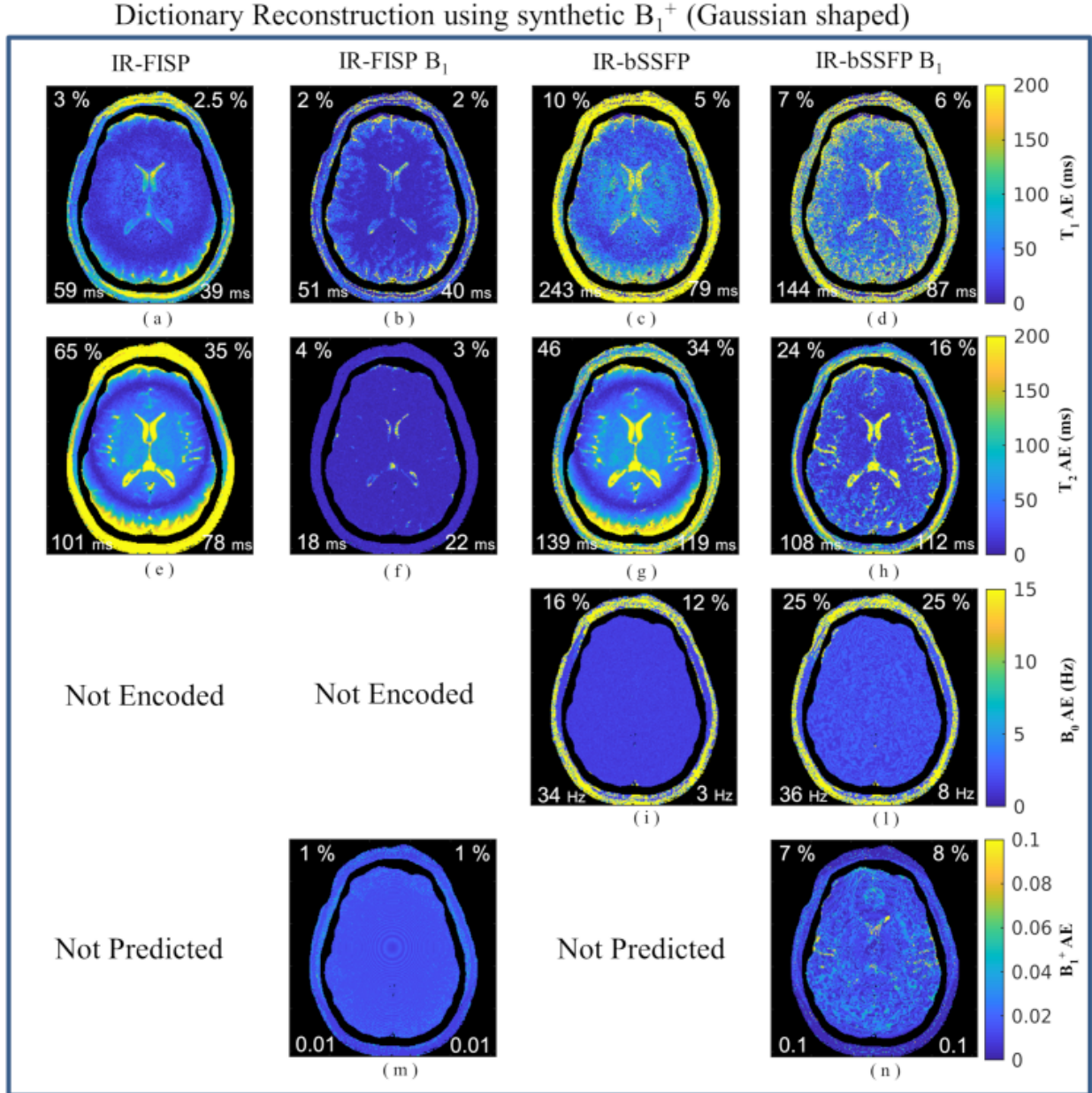
Fig. 12: Absolute error maps between dictionary reconstructed and ground truth parameter maps with Gaussian shaped B_1^+ field inhomogeneities. Global error estimators are reported on the four corners of each image: upper left corner reports the MAPE; upper right corner reports MAPE evaluated without considering the scalp region; lower left corner reports the RMSE while lower right corner reports RMSE evaluated without considering the scalp region.

possible, such as B_0 and B_1^+ inhomogeneities. With the classical dictionary-based approach for MRF, each added parameter results in an exponential increase of dictionary entries, making it difficult to add other relevant MR parameters, such as diffusion. These results show the advantages of NN approaches in MR Fingerprinting parameters estimation. Thousands of training examples might be required to learn an accurate and robust ITF with NN, as shown for IR-bSSFP and IR-bSSFP B_1 sequences. However, because the NN model is trained with batches, the data can be stored in the hard memory, and just few of them loaded into the RAM in each step of the training. Moreover, because the training examples are synthetic data, one can even make the simulation step within the training pipeline, without the need to store any data even in the hard memory. The latter step requires the simulation code to be fast and efficient, otherwise it becomes a bottleneck in terms of time needed to train the network model. However, parallel coding is well suited for this kind of problem and can be used to make computation faster.

The above results describe NN approaches as promising for MRF application. Hence, this work may help the community working in MRF to build more trust in deep learning approaches and eventually take advantage of them, and pushing fingerprinting pulse sequences design to add more meaningful MR parameters, such as diffusion, without the limitations due to dictionary size.

REFERENCES

[1] D. Ma, V. Gulani, N. Seiberlich, K. Liu, J. L. Sunshine, J. L. Duerk, and M. A. Griswold, "Magnetic resonance fingerprinting," *Nature*, vol. 495, p. 187192, Mar. 2013.

[2] B. Lemasson, N. Pannetier, N. Coquery, L. S. B. Boisserand, N. Collomb, N. Schuff, M. Moseley, G. Zaharchuk, E. L. Barbier, and E. L. Christen, "Mr vascular fingerprinting in stroke and brain tumors models," *Sci. Rep.*, vol. 6, p. 37071, Nov. 2016.

[3] Y. Chen, Y. Jiang, S. Pahwa, D. Ma, L. Lu, M. D. Twieg, K. L. Wright, N. Seiberlich, M. A. Griswold, and V. Gulani, "Mr fingerprinting for rapid quantitative abdominal imaging," *Radiology*, vol. 279, no. 1, pp. 278–286, 2016. PMID: 26794935.

[4] Y. Chen, A. Panda, S. Pahwa, J. I. Hamilton, S. Dastmalchian, D. F. McGivney, D. Ma, J. Batesole, N. Seiberlich, M. A. Griswold, D. Plecha, and V. Gulani, "Three-dimensional mr fingerprinting for quantitative breast imaging," *Radiology*, 2018. PMID: 30375925.

[5] J. Wang, W. Mao, M. Qiu, M. B. Smith, and R. T. Constable, "Factors influencing flip angle mapping in mri: Rf pulse shape, slice-select gradients, off-resonance excitation, and b0 inhomogeneities," *Magn. Reson. Med.*, vol. 56, no. 2, pp. 463–468, 2006.

[6] G. Buonincontri and S. J. Sawiak, "Mr fingerprinting with simultaneous b1 estimation," *Magn. Reson. Med.*, vol. 76, pp. 1127–1135, Sept. 2015.

[7] D. Ma, S. Coppo, Y. Chen, D. F. McGivney, Y. Jiang, S. Pahwa, V. Gulani, and M. A. Griswold, "Slice profile and b1 corrections in 2d magnetic resonance fingerprinting," *Magnetic Resonance in Medicine*, vol. 78, no. 5, pp. 1781–1789, 2017.

[8] D. F. McGivney, E. Pierre, D. Ma, Y. Jiang, H. Saybasili, V. Gulani, and M. A. Griswold, "Svd compression for magnetic resonance fingerprinting in the time domain," *IEEE Trans. Med. Imaging*, vol. 33, no. 12, pp. 2311–2322, Dec. 2014.

[9] G. Mazor, L. Weizman, A. Tal, and Y. C. Eldar, "Low Rank Magnetic Resonance Fingerprinting," *ArXiv e-prints*, Jan. 2017.

[10] M. Yang, D. Ma, Y. Jiang, J. Hamilton, N. Seiberlich, M. A.Griswold, and D. McGivney, "Low rank approximation methods for mr fingerprinting with large scale dictionaries," *Magn. Reson. Med*, vol. 79, pp. 2392–2400, Aug. 2017.

[11] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Networks*, vol. 4, no. 2, pp. 251 – 257, 1991.

[12] P. Virtue, S. X. Yu, and M. Lustig, "Better than Real: Complex-valued Neural Nets for MRI Fingerprinting," *ArXiv e-prints*, June 2017.

[13] E. Hoppe, G. Körzdörfer, T. Würfl, J. Wetzl, F. Lugauer, J. Pfeuffer, and A. K. Maier, "Deep learning for magnetic resonance fingerprinting: A new approach for predicting quantitative parameter values from time series," *Studies in health technology and informatics*, vol. 243, pp. 202–206, 2017.

[14] O. Cohen, B. Zhu, and M. S. Rosen, "Mr fingerprinting deep reconstruction network (drone)," *Magn. Reson. Med*, vol. 80, pp. 885–894, Apr. 2018.

[15] F. Balsiger, A. Shridhar Konar, S. Chikop, V. Chandran, O. Scheidegger, S. Geethanath, and M. Reyes, "Magnetic Resonance Fingerprinting Reconstruction via Spatiotemporal Convolutional Neural Networks," *ArXiv e-prints*, July 2018.

[16] Y. Jiang, D. Ma, N. Seiberlich, V. Gulani, and M. A. Griswold, "Mr fingerprinting using fast imaging with steady state precession (fisp) with spiral readout," *Magn. Reson. Med.*, vol. 74, pp. 1621–1631, Dec. 2014.

[17] G. C. Borgia, R. J. S. Brown, and P. Fantazzini, "Uniform-penalty inversion of multiexponential decay data," *J Magn Reson*, no. 132, p. 6577, 1998.

[18] V. Bortolotti, L. Brizi, P. Fantazzini, G. Landi, and F. Zama, "Filtering techniques for efficient inversion of two-dimensional nuclear magnetic resonance data," *Journal of Physics: Conference Series*, vol. 904, no. 1, p. 012005, 2017.

[19] R. Bellman, *Adaptive control processes: a guided tour*. Princeton University Press, 1961.

[20] J. H. Friedman, "On bias, variance, 0/1–loss, and the curse-of-dimensionality," *Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 55–77, 1997.

[21] O. Cohen and M. S. Rosen, "Algorithm comparison for schedule optimization in mr fingerprinting," *Magnetic Resonance Imaging*, vol. 41, pp. 15 – 21, 2017.

[22] X. Cao, C. Liao, Z. Wang, Y. Chen, H. Ye, H. He, and J. Zhong, "Robust sliding-window reconstruction for accelerating the acquisition of mr fingerprinting," *Magnetic Resonance in Medicine*, vol. 78, no. 4, pp. 1579–1588, 2017.

[23] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, p. 436444, May 2015.

[24] H. Jrgen, "Echoeshow to generate, recognize, use or avoid them in mr-imaging sequences. part i: Fundamental and not so fundamental properties of spin echoes," *Concepts in Magnetic Resonance*, vol. 3, no. 3, pp. 125–143, 1991.

[25] H. Gudbjartsson and S. Patz, "The rician distribution of noisy mri data," *Magn. Reson. Med.*, vol. 34, no. 6, pp. 910–914, 1995.

[26] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org.

[27] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *ArXiv e-prints*, Dec. 2014.

[28] 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, S. A. Smith, S. Joel, S. Mori, J. J. Pekar, P. B. Barker, J. L. Prince, and P. C. van Zijl, "Multi-parametric neuroimaging reproducibility: A 3-t resource study," *NeuroImage*, vol. 54, no. 4, pp. 2854 – 2866, 2011.

[29] J.-F. Cabana, Y. Gu, M. Boudreau, I. R. Levesque, Y. Atchia, J. G. Sled, S. Narayanan, D. L. Arnold, G. B. Pike, J. Cohen-Adad, T. Duval, M.-T. Vuong, and N. Stikov, "Quantitative magnetization transfer imaging made easy with qmtlab: Software for data simulation, analysis, and visualization," *Concepts in Magnetic Resonance Part A*, vol. 44A, no. 5, pp. 263–277, 2015.

[30] M. Boudreau, C. L. Tardif, N. Stikov, J. G. Sled, W. Lee, and G. B. Pike, "B1 mapping for bias-correction in quantitative t1 imaging of the brain at 3t using standard pulse sequences," *J. Magn. Reson. Imaging*, vol. 46, no. 6, pp. 1673–1682, 2017.

[31] L. I.-K. Lin, "A concordance correlation coefficient to evaluate reproducibility," *Biometrics*, vol. 45, no. 1, pp. 255–268, 1989.