Physics-Driven Deep Learning for Computational Magnetic Resonance Imaging

Kerstin Hammernik, Thomas Küstner, Burhaneddin Yaman, Zhengnan Huang,
Daniel Rueckert, Florian Knoll, Mehmet Akçakaya

Abstract

Physics-driven deep learning methods have emerged as a powerful tool for computational magnetic resonance imaging (MRI) problems, pushing reconstruction performance to new limits. This article provides an overview of the recent developments in incorporating physics information into learning-based MRI reconstruction. We consider inverse problems with both linear and non-linear forward models for computational MRI, and review the classical approaches for solving these. We then focus on physics-driven deep learning approaches, covering physics-driven loss functions, plug-and-play methods, generative models, and unrolled networks. We highlight domain-specific challenges such as real- and complex-valued building blocks of neural networks, and translational applications in MRI with linear and non-linear forward models. Finally, we discuss common issues and open challenges, and draw connections to the importance of physics-driven learning when combined with other downstream tasks in the medical imaging pipeline.

Index Terms

Accelerated MRI, parallel imaging, iterative image reconstruction, numerical optimization, machine learning, deep learning, physics-driven learning.

K. Hammernik and D. Rueckert are with the Institute of AI and Informatics in Medicine, Technical University of Munich and the Department of Computing, Imperial College London.

T. Küstner is with the Department of Diagnostic and Interventional Radiology, University Hospital of Tuebingen.

B. Yaman and M. Akçakaya are with the Department of Electrical and Computer Engineering, and Center for Magnetic Resonance Research, University of Minnesota, USA.

Z. Huang is with the Center for Biomedical Imaging, Department of Radiology, New York University.

F. Knoll is with the Department Artificial Intelligence in Biomedical Engineering, Friedrich-Alexander University Erlangen.

We acknowledge grant support from the NIH, Grant Numbers: R01HL153146, P41EB027061, R01EB024532 and P41EB017183; from the NSF, Grant Number: CAREER CCF-1651825; and from the EPSRC Programme Grant, Grant Number: EP/P001009/1.

I. Introduction

Magnetic resonance imaging (MRI) is a non-invasive radiation-free imaging modality with a plethora of clinical applications and extensively-studied physics underpinnings. The relationship between the acquired MRI data and the underlying magnetization is characterized by Bloch equations, and depends on a number of parameters, including the magnetic fields (e.g. the static B_0 magnetic field), relaxation effects (e.g. T_1, T_2 relaxation), motions at different scales (e.g. physiological, flow, diffusion and perfusion), and acquisition parameters (e.g. echo time, flip angle) [1]. These intricate dependencies are encoded in the so-called k-space, corresponding to the spatial Fourier transform of the object's magnetization. The acquired k-space signal $y(t)$ at time t, prior to discretization, is given as

$$y(t) = \int M(\rho(\mathbf{r}), \boldsymbol{\vartheta}, t, \mathbf{r}) \mathrm{e}^{-j2\pi \mathbf{k}(t)\cdot\mathbf{r}} \,\mathrm{d}\mathbf{r} + n(t) \tag{1}$$

where $\mathbf{r}$ is the spatial location; $\rho(\mathbf{r})$ is the underlying spin densities/transverse magnetization; $\boldsymbol{\vartheta}$ is a set of (potentially unknown) parameters that model physiological or systemic changes, and themselves may depend on $\mathbf{r}$; $k(t)$ is the k-space location at time t sampled along a k-space trajectory under the influence of spatially and temporally varying magnetic fields; and $n(t)$ is measurement noise. The physics-based signal model $M(\rho(\mathbf{r}), \boldsymbol{\vartheta}, t_j, \mathbf{r})$, sampled at times t_j, thereby describes the effects that influence the underlying magnetization, based on pre-specified and known image acquisition parameters. It depends on the imaging sequence and reflects physiological, functional or hardware characteristics. For many applications, an analytical expression can be derived (e.g. via hard pulse approximation from the Bloch equations) for which a few examples are summarized in Table I (linear and non-linear models). If no analytical expression can be derived for the imaging sequence, the Bloch equations need to be integrated directly as the signal model.

TABLE I: Analytically derived physics-based signal models for pre-specified imaging sequences for set and known image acquisition parameters (dominant one influencing the signal model is depicted) and the to be estimated unknown parameters [1, 2].

physical effect	image acquisition parameters	unknown parameter $\boldsymbol{\vartheta}$	signal model $M(\rho(\mathbf{r}), \boldsymbol{\vartheta}, t_j, \mathbf{r})$
off-resonance	echo time t_j	$\Delta\omega$	$\mathrm{e}^{j\Delta\omega(\mathbf{r})t_j}\rho(\mathbf{r})$
motion	echo time t_j	motion field $\mathbf{U}_j$	$\rho\left(\mathbf{U}_j(\mathbf{r})\right) \lvert \det(\nabla\mathbf{U}_j)(\mathbf{r}) \rvert$
T_1 relaxation	inversion times t_k, equilibrium magnetization ρ_0	T_1	$\rho_0(\mathbf{r})(1 - \mathrm{e}^{-\frac{t_j}{T_1(\mathbf{r})}})$
T_2 relaxation	echo time t_j	T_2	$\mathrm{e}^{-\frac{t_j}{T_2(\mathbf{r})}}\rho(\mathbf{r})$
flow velocity $\mathbf{v}$	flow-encoded acquisitions $\mathbf{V}_j$	$\mathbf{v}$	$\mathrm{e}^{j\mathbf{v}\cdot\mathbf{V}_j}\rho(\mathbf{r})$
diffusion tensor $\mathbf{D}$	diffusion-encoded acquisitions $\mathbf{b}_j$	$\mathbf{D}$	$\mathrm{e}^{-\mathbf{b}_j^T\mathbf{D}\mathbf{b}_j}\rho(\mathbf{r})$

For a simplified acquisition model, the signal in $M(\rho(\mathbf{r}), \boldsymbol{\vartheta}, t, \mathbf{r})$ is often characterized as $x(\mathbf{r})$, which absorbs the dependencies on the physiologic or systemic effects, as well as the signal evolution (or time-course), into the image/magnetization of interest. For example, this type of simplification is used when referring to images with different contrast weightings, such as T_1 or T_2 weighting. In this setup, following discretization, the physics-based forward model becomes linear and can be expressed as

$$\mathbf{y} = \mathbf{E}\mathbf{x} + \mathbf{n}, \tag{2}$$

where $\mathbf{x} \in \mathbb{C}^n$ is the image/magnetization of interest, $\mathbf{y} \in \mathbb{C}^m$ denotes the corresponding k-space measurements, $\mathbf{E}$: $\mathbb{C}^n \rightarrow \mathbb{C}^m$ is the forward MRI encoding operator, and $\mathbf{n} \in \mathbb{C}^m$ is discretized measurement noise. In its simplest form, $\mathbf{E}$ corresponds to a sub-sampled discrete Fourier transform matrix $\mathbf{F}_\Omega$: $\mathbb{C}^n \rightarrow \mathbb{C}^m$ which samples the k-space locations specified by Ω. In practice, however, all clinical MRI scanners from all vendors are equipped with multi-coil receiver arrays, and the corresponding multi-coil forward operator $\mathbf{E}$: $\mathbb{C}^n \rightarrow \mathbb{C}^{m \cdot n_c}$ is given as

$$\mathbf{E} = \begin{bmatrix} \mathbf{F}_\Omega \mathbf{C}_1 \\ \vdots \\ \mathbf{F}_\Omega \mathbf{C}_{n_c} \end{bmatrix},$$

where n_c is the number of coils in the receiver array, and $\mathbf{C}_q$: $\mathbb{C}^n \rightarrow \mathbb{C}^n$ is a diagonal matrix containing the sensitivity profile of the q^{th} receiver coil. These coil sensitivities are typically pre-estimated from subject-specific calibration data [3]. We note that while $\mathbf{F}_\Omega$ typically refers to a sub-sampled Cartesian acquisition that can be implemented efficiently with a fast Fourier transform, non-Cartesian acquisitions are also used in some clinical applications.

Formation of images and other information from these measured k-space data constitutes the basis of computational MRI, which in itself has a rich history. The canonical inverse problem for computational MRI relates to the formation of images from sub-sampled/degraded k-space data. Solving such inverse problems often necessitates incorporation of additional information about MRI encoding and/or the nature of MR images. Earlier works concentrated on the properties of the k-space, such as partial Fourier imaging methods that utilize Hermitian symmetry. With the advent of multi-coil receiver arrays, the redundancies among these coil elements became the important information for the next generation of inverse problems [3].

In addition to the above canonical linear inverse problems, there is a class of computational MRI methods that deals with the more complicated non-linear forward models incorporating physical, systemic and physiological parameters, as stated in Table I. The forward model in this case can be broadly given as [1]:

$$\mathbf{y} = \mathcal{E}(\mathbf{v}) + \mathbf{n}, \tag{3}$$

where $\mathbf{v} \in \mathbb{C}^{n_v}$ is a vector that includes all unknown imaging/quantity and parameters of interest that describes the signal evolution in Eq. (1), and $\mathcal{E} : \mathbb{C}^{n_v} \to \mathbb{C}^{m \cdot n_c}$ is a non-linear encoding operator, i.e. the signal evolution arising from the physics-based signal model of Eq. (1). It can be decomposed into $\mathcal{E} = \mathbf{E}\mathcal{M}$, where $\mathbf{E}$ is the canonical multi-coil forward operator and $\mathcal{M} : \mathbb{C}^{n_v n_\vartheta} \to \mathbb{C}^{n}$ is the discretized signal model describing the spin physics. Here, we make the distinction that $\mathbf{v}$ includes all unknown quantities of interest that describes the signal evolution, as opposed to just an image as in Eq. (2). This broad definition is necessary to incorporate different setups [1], which are partially described in Table I. For example, for the motion model in Table I, $\mathbf{v}$ includes both the motion field and the image of interest. For this model, the former was specified by $\boldsymbol{\vartheta}$ as the unknown physiological parameter, but one is typically interested in recovering the image itself. For a relaxation model, $\mathbf{v}$ includes both the magnetization and the relaxation map (e.g. T_1 or T_2). In this setup, the quantity of interest is the relaxation map, which was specified by $\boldsymbol{\vartheta}$ as the unknown physical parameter, but the magnetization also needs to be recovered to fully describe the model. Thus, it is not straightforward to tease out the unknown parameter from the magnetization in all cases, where the object of clinical interest may be either. Hence the unified notation [1] makes the inverse problem easier to specify without going into application details. Traditionally, the inverse problem corresponding to Eq. (3) is solved using model-based reconstructions.

Recently, deep learning methods have emerged as a powerful tool for solving many inverse problems in computational MRI. Among these, MRI reconstruction for accelerated acquisitions remains the most well-studied [4–6], along with several strategies for quantitative MRI [7], motion [8] and other non-linear physical models [9]. Out of a plethora of approaches for these problems, physics-driven methods,which explicitly use the known physics-based forward imaging models in deep learning architectures and/or training to control the consistency of the reconstruction with k-space measurements, have emerged as the most well-received deep learning techniques by the MRI community due to their incorporation of the MR domain knowledge. The goal of this manuscript is to provide a comprehensive review of inverse problems for computational MRI, and how physics-driven deep learning techniques involving the raw k-space data are being used for these applications.

II. Classical approaches for computational MRI

The simplest image reconstruction problem for computational MRI concerns the case where $\mathbf{E}$ is exactly the discrete Fourier transform matrix in Eq. (2), corresponding to Nyquist rate sampling for a given resolution and field-of-view. In this case, the image of interest is recovered via inverse discrete Fourier transform.

In practical settings, often a sub-Nyquist rate is employed to enable faster imaging, where the previous simple strategy of taking the inverse Fourier transform leads to aliasing artifacts. Thus, in this regime, an inverse problem, incorporating additional domain knowledge, needs to be solved for image formation. The most commonly used clinical strategy for accelerated MRI is parallel imaging [3], which uses the redundancies among these coil elements for image reconstruction. Succinctly, parallel imaging methods that work in image domain [3] solved

$$\hat{\mathbf{x}}_{PI} = \arg\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - \mathbf{E}\mathbf{x}||_2^2 = (\mathbf{E}^H\mathbf{E})^{-1}\mathbf{E}^H\mathbf{y}, \tag{4}$$

where H denotes the Hermitian transpose. In theory, with n_c coil elements, the ratio between the image size and the cardinality of Ω, or the acceleration rate (R), can be as high as n_c. However, due to spatial or statistical dependencies between $\{\mathbf{C}_k\}$ and ill-conditioning of $\mathbf{E}$ that leads to noise amplification due to the matrix inversion [3], the achievable rates are often limited. Subsequently, compressed sensing methods were proposed to utilize the compressibility of MR images to reconstruct images from sub-sampled k-space data. These methods solve a regularized least squares objective function [10]

$$\hat{\mathbf{x}}_{CS} = \arg\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - \mathbf{E}\mathbf{x}||_2^2 + \tau||\mathbf{W}\mathbf{x}||_1, \tag{5}$$

where $||\cdot||_1$ denotes the ℓ_1 norm, $\mathbf{W}$ is a sparsifying linear transform, and τ is a weight term. Unlike (4), the objective function does not have a closed form solution. We also note that Eq. (5) corresponds to the analysis formulation of ℓ_1 regularization, while the synthesis formulation which performs regularization in the transform domain directly also remains popular. The two are equivalent when $\mathbf{W}$ is a unitary transformation. Both the synthesis and analysis formulations lead to a convex problem, and can be solved using numerous iterative algorithms [10].

A. Solving the linear inverse problem in classical computational MRI

In general, we will consider a regularized least squares objective with a broader class of regularizers:

$$\hat{\mathbf{x}}_{reg} = \arg\min_{\mathbf{x}} \frac{1}{2}||\mathbf{y} - \mathbf{E}\mathbf{x}||_2^2 + \mathcal{R}(\mathbf{x}), \tag{6}$$

where $\mathcal{R}(\cdot)$ may be one of the aforementioned regularizers, such as the ℓ_1 norm of transform domain coefficients, or implicitly implemented via machine learning techniques, as we will later see.

There is a number of iterative algorithms for solving such objective functions, especially when it is convex [10]. A classical approach, when $\mathcal{R}(\cdot)$ is differentiable, is based on gradient descent:

$$\mathbf{x}^{(i)} = \mathbf{x}^{(i-1)} + \eta\mathbf{E}^H(\mathbf{y} - \mathbf{E}\mathbf{x}^{(i-1)}) - \eta\nabla_{\mathbf{x}}\mathcal{R}(\mathbf{x})|_{\mathbf{x}=\mathbf{x}^{(i-1)}}, \tag{7}$$

where $\mathbf{x}^{(i)}$ is the image of interest at the i^{th} iteration. However, often times nonsmooth regularizers are used in computational MRI. In this case, proximal algorithms are widely used [10]. One such method is proximal gradient descent, which amounts to solving two sub-problems:

$$\mathbf{z}^{(i)} = \mathbf{x}^{(i-1)} + \eta \mathbf{E}^H(\mathbf{y} - \mathbf{E}\mathbf{x}^{(i-1)}), \tag{8a}$$

$$\mathbf{x}^{(i)} = \arg\min_{\mathbf{x}} \frac{1}{2}||\mathbf{z}^{(i)} - \mathbf{x}||_2^2 + \eta\mathcal{R}(\mathbf{x}) \triangleq \text{prox}_{\mathcal{R},\eta}(\mathbf{z}^{(i)}) \tag{8b}$$

where $\mathbf{x}^{(i)}$ and $\mathbf{z}^{(i)}$ are the image of interest and an intermediate image at the i^{th} iteration respectively, Eq. (8b) corresponds to the so-called proximal operator for the regularizer, Eq. (8a) encourages data consistency, and η is a step size.

Another class of popular approaches rely on variable splitting, such as the alternating direction method of multipliers (ADMM), which solves:

$$\mathbf{x}^{(i)} = \left(\mathbf{E}^H\mathbf{E} + \rho\mathbf{I}\right)^{-1}\left(\mathbf{E}^H\mathbf{y} + \rho(\mathbf{z}^{(i-1)} - \mathbf{u}^{(i-1)})\right), \tag{9a}$$

$$\mathbf{z}^{(i)} = \arg\min_{\mathbf{z}} \frac{1}{2}\big|\big|(\mathbf{x}^{(i)} + \mathbf{u}^{(i-1)}) - \mathbf{z}\big|\big|_2^2 + \frac{1}{\rho}\mathcal{R}(\mathbf{z}), \tag{9b}$$

$$\mathbf{u}^{(i)} = \mathbf{u}^{(i-1)} + (\mathbf{x}^{(i)} - \mathbf{z}^{(i)}), \tag{9c}$$

where $\mathbf{x}^{(i)}$ is the image of interest at the i^{th} iteration, $\mathbf{z}^{(i)}$ and $\mathbf{u}^{(i)}$ are intermediate images, and ρ is a penalty weight. Here, (9a), (9b) and (9c) corresponds to data consistency, proximal operator and dual update sub-problems respectively. A simpler version of variable splitting is based on a quadratic penalty [10], which leads to the following equations:

$$\mathbf{x}^{(i)} = \left(\mathbf{E}^H\mathbf{E} + \rho\mathbf{I}\right)^{-1}\left(\mathbf{E}^H\mathbf{y} + \rho\mathbf{z}^{(i-1)}\right), \tag{10a}$$

$$\mathbf{z}^{(i)} = \arg\min_{\mathbf{z}} \frac{1}{2}\big|\big|\mathbf{x}^{(i)} - \mathbf{z}\big|\big|_2^2 + \frac{1}{\rho}\mathcal{R}(\mathbf{z}), \tag{10b}$$

B. Solving the non-linear inverse problem in classical computational MRI

The general unconstrained optimization problem for a model-based reconstruction for the forward model in Eq. (3) can be stated as:

$$\hat{\mathbf{v}} = \arg\min_{\mathbf{v}} \|\mathbf{E}\mathcal{M}(\mathbf{v}) - \mathbf{y}\|_2^2 + R_{\mathbf{v}}(\mathbf{v}) \tag{11}$$

where $R_{\mathbf{v}}$ is a (combination of) regularizer that acts on all unknown quantities of interest. While the notation is general, the regularizer may be separable among different quantities, e. g. different regularizers for the motion field and the image of interest. This description can be used to combine parallel imaging, compressed sensing and model-based reconstruction in a unified formulation. In addition to the motion and T_2 mapping models discussed earlier in Section I, a non-linear forward model can also be used to describe dynamic imaging scenarios, such as contrast-enhanced imaging. Consider the signal model in Eq. (1), and time instances specified $\tau_0(t) \triangleq 0, \tau_1(t), \tau_2(t), \ldots, \tau_n(t)$. These time instances may correspond to different physical events, e.g. RF excitation for single-shot EPI acquisitions, sampling after an inversion pulse for T_1 mapping, or cardiac triggering for myocardial parameter mapping or perfusion cardiac MRI. Let the discretized k-space measurements between $\tau_{i-1}(t)$ and $\tau_i(t)$ be denoted by $\mathbf{y}_i$. Thus, each $\mathbf{y}_i$, corresponding to $\{y(t)\}_{\tau_{i-1}(t) \leq t < \tau_i(t)}$, essentially captures a snap-shot of this dynamic process between the specified sample instances. In the same vein as Eq. (3), these can be vectorized into $\mathbf{y}$, where the corresponding $\mathbf{v}$ models the relevant pharmacokinetic quantities.

For the inverse problems with non-linear forward operators, the algorithms are less standardized, and typically application-dependent. Eq. (11) is usually non-convex, making its optimization a challenging task. Furthermore, inaccuracies or incompleteness of the modelling can further influence the optimization. One approach is to employ a Gauss-Newton algorithm, and linearize the problem around the solution of the previous iteration or by approximating the non-linear behaviour with a linear combination of basis functions.

III. Physics-driven ML methods in computational MRI

Deep learning methods have recently emerged as a powerful tool for computational MRI. These methods can be broadly split into two classes: purely data-driven and physics-driven [6]. The former methods are typically implemented in image space, as removing artifacts from aliased images. These image enhancement networks are typically trained to map corrupted and undersampled images to artifact-free images. Indeed, learning image enhancement networks is the key ingredient to remove artifacts in image domain. However, when only image enhancement methods are used, the information of the acquisition physics is entirely discarded, hence, k-space consistency cannot be guaranteed. In this section, we will give an extensive overview on physics-driven deep learning methods for computational MRI, ranging from physics-informed enhancement methods to learned unrolled optimization, as well as reconstruction with generative models and plug-and-play priors.

A. Physics Information in Image/k-space Enhancement Methods

As aforementioned, image enhancement networks typically learn a mapping from the aliased/degraded image, such as the zero-filled reconstruction, to a reference image, without consideration of the measured k-space data during the reconstruction process. For Cartesian sampling, several attempts have been made to incorporate physics information in this line of work [11, 12] , including enforcing k-space consistency directly after image enhancement [11], or adding k-space consistency as an additional cost function term during training [12]. The former approach directly replaces the measured k-space lines, which may lead to artifacts, while the latter cannot guarantee k-space consistency during inference, especially for cases with unseen pathologies. Specifically, in [11], enforcing hard k-space consistency directly after image enhancement was proposed, where the enhanced image was transformed into Fourier space, and the points at the sampled locations were replaced by the original k-space measurements. However, we note that this approach cannot be applied to more complex sampling trajectories in non-Cartesian imaging. In [12], k-space consistency was added as an additional cost function term during training. However, this approach cannot guarantee k-space consistency during inference, especially for cases with unseen pathologies. Similarly, enhancement has been proposed in k-space, as a method of interpolation [13], where a non-linear interpolation function is learned from calibration data. This can be seen as an extension to the linear convolution kernels used in generalized autocalibrating partially parallel acquisitions (GRAPPA). As only the calibration data is required for training, this approach can be used when large training databases are not available, but its performance may be limited at high acceleration rates where the calibration data may be insufficient [6].

B. Plug-and-play Methods with Deep Denoisers

Plug-and-play (PnP) algorithms decouple image modeling from the physics of the MRI acquisition, by noting that the proximal operators in Eq. (8b) or Eq. (9b) correspond to conventional denoising problems [14]. In the proximal-based formulation, these proximal denoisers are replaced by other powerful denoising algorithms, which do not necessarily have a corresponding closed form $\mathcal{R}(\cdot)$ expression, such as BM3D [14]. A related approach is the regularization by denoising (RED) framework, which considers finding $\mathbf{x}$ that solves the first-order optimality condition

$$\mathbf{0} = \mathbf{E}^H(\mathbf{E}\mathbf{x} - \mathbf{y}) + \lambda(\mathbf{x} - d(\mathbf{x})), \tag{12}$$

where $d(\cdot)$ is the plug-in denoiser [14], and $\lambda > 0$ denotes the regularization parameter. The advantage of the RED formulation is that under certain conditions, the regularizer, $\mathcal{R}(\cdot)$ can be explicitly tied to the denoiser, $d(\cdot)$. We refer the reader to a comprehensive review article on the subject [14] for more details. We also note that, beyond the computational MRI community, there has been work characterizing the guaranteed convergence of plug-and-play networks.

Recently, more effort has been made towards implementing CNN-based denoisers in these PnP frameworks [14, 15], depicted in Figure 1. These denoisers are typically trained using reference images in a supervised manner, where different levels of noise are retrospectively added to these images, and a

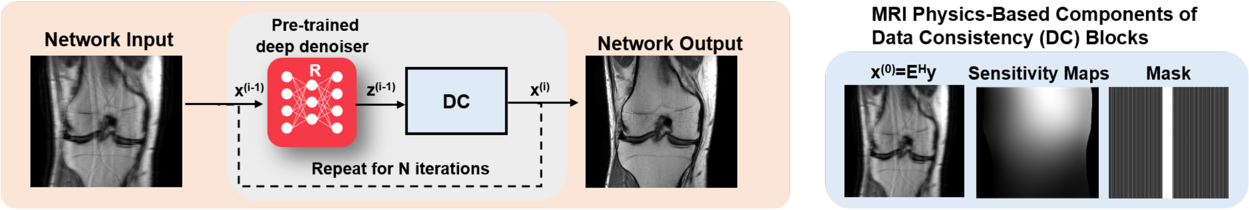

Fig. 1: Overview of the PnP framework in physics-driven deep learning methods for computational MRI. The data consistency layer enforces fidelity with k-space measurements based on the known forward model. Note that a Cartesian sampling scheme is shown for easier depiction, but data consistency is also applicable to non-Cartesian trajectories.

mapping from the noisy images to reference images are learned [14]. In applications, where reference images are unavailable, Noise2Noise denoising framework has been proposed for training using pairs of noisy images. Extending on this work, regularization by artifact removal (RARE) trained CNN denoisers on a database of pairs of images with artifacts generated from non-Cartesian acquisitions [15]. These pairs were generated by splitting the acquired measurements in half, and reconstructing these with least squares as in Eq.(4), corresponding to a parallel imaging reconstruction, which led to starting images of sufficient quality for non-Cartesian trajectories that oversample the central region of k-space. The appeal of these methods is that the CNN-based denoisers are trained independently of the broader inverse problem. Thus, only the denoising network has to be stored in memory, allowing for easier translation to larger-scale as proposed in RARE [15] for 3D MRI datasets . This approach is also appealing since only one denoiser has to be trained on any data. Hence, this denoiser can, in principle, be applied across different rates or undersampling patterns. In practice, it is beneficial to provide the denoiser with additional information, such as the undersampling artifacts arising from uniform undersampling pattern in order to recognize characteristic aliasing artifacts.

C. Generative Models

While we have reviewed explicit regularization in Eq. (6), regularization can also be achieved by an implicit prior in order to constrain the solution space for our optimization problem. This concept is proposed by Deep Image Prior (DIP) as follows:

$$\min_{\boldsymbol{\theta}} \frac{1}{2}\|\mathbf{E}G_{\boldsymbol{\theta}}(\mathbf{z}) - \mathbf{y}\|_2^2, \tag{13}$$

where a generator network, $G_{\boldsymbol{\theta}}$: $\mathbb{C}^d \rightarrow \mathbb{C}^n$ parametrized by $\boldsymbol{\theta} \in \mathbb{R}^p$, reconstructs an image $\mathbf{x} = G_{\boldsymbol{\theta}}(\mathbf{z}) \in \mathbb{C}^n$ from a random d-dimensional latent vector $\mathbf{z} \in \mathbb{C}^d$. This loss function is used to train the generator network with parameters $\boldsymbol{\theta}$. This formulation has the advantage that it works for limited (even single) datasets without ground-truth. However, early stopping has to be performed to not overfit to the noisy measurements. An extension of the DIP framework to dynamic non-Cartesian MRI was proposed in [16]. A mapping network first generated an expressive latent space from a fixed low-dimensional manifold, e.g. a straight-line manifold, using fully connected layers and non-linearities. A subsequent generative CNN generates the finally reconstructed image.

An alternative line of work is based on generative adverserial networks (GANs), where a generator and a discriminator network play a minimax game. The generator network samples from a fixed distribution in latent space such as Gaussian distribution and aims to map the sampling to a real data distribution in the ambient image space. Conversely, the discriminator network aims to differentiate between generated and real samples. The minimax training objective is defined as

$$\min_{\boldsymbol{\theta}_G} \max_{\boldsymbol{\theta}_D} \mathcal{L}_{GAN}(\boldsymbol{\theta}_D, \boldsymbol{\theta}_G) \triangleq \mathbb{E}_{\mathbf{x}}[\log D_{\boldsymbol{\theta}_D}(\mathbf{x})] + \mathbb{E}_{\mathbf{z}}[\log(1 - D_{\boldsymbol{\theta}_D}(G_{\boldsymbol{\theta}_G}(\mathbf{z}))], \tag{14}$$

where the distribution on $\mathbf{x}$ is the real data distribution, whereas the one on $\mathbf{z}$ is a fixed distribution on the latent space, and $\mathbb{E}_{\mathbf{z}}$ and $\mathbb{E}_{\mathbf{x}}$ denote the expected values defined over the random variables $\mathbf{z}$ and $\mathbf{x}$. The generator $G_{\boldsymbol{\theta}_G}$, parametrized by $\boldsymbol{\theta}_G \in \mathbb{R}^{p_G}$, tries to map samples from the latent space to samples from the ambient image space, and the discriminator $D_{\boldsymbol{\theta}_D}$, parametrized by $\boldsymbol{\theta}_D \in \mathbb{R}^{p_D}$, tries to differentiate between the generated and the real samples.

The idea of using GANs in computational MRI was first proposed in [17]. In this case, the generator network used the zero-filled images as input instead of a random distribution, leading to the loss function

$$\min_{\boldsymbol{\theta}_G} \max_{\boldsymbol{\theta}_D} \mathbb{E}_{\mathbf{x}}[\log D_{\boldsymbol{\theta}_D}(\mathbf{x})] + \mathbb{E}_{\mathbf{y}}[\log(1 - D_{\boldsymbol{\theta}_D}(G_{\boldsymbol{\theta}_G}(\mathbf{E}^H\mathbf{y}))]. \tag{15}$$

At inference time, the generator was used to produce the desired output. Here $G_{\boldsymbol{\theta}_G}$ was an image enhancement network, followed by a data consistency step, for instance implemented by a gradient descent step as in Eq. (8a); while the discriminator $D_{\boldsymbol{\theta}_D}$ was essentially used to implement an adversarial loss term to improve the recovery of finer details. Thus, this formulation used supervised training with paired data. A high-level overview of this approach is shown in Figure 2. A more recent work replaced this generator with a variational autoencoder based generator that also allowed for uncertainty quantification [18].

Another approach is based on inverse GANs, which utilize generative learning, followed by optimization similar to the DIP [19]. First, a GAN is trained to generate an MR image from a latent noise vector. The GAN does not involve any physics-based knowledge, as only clean MRI reference images are used for

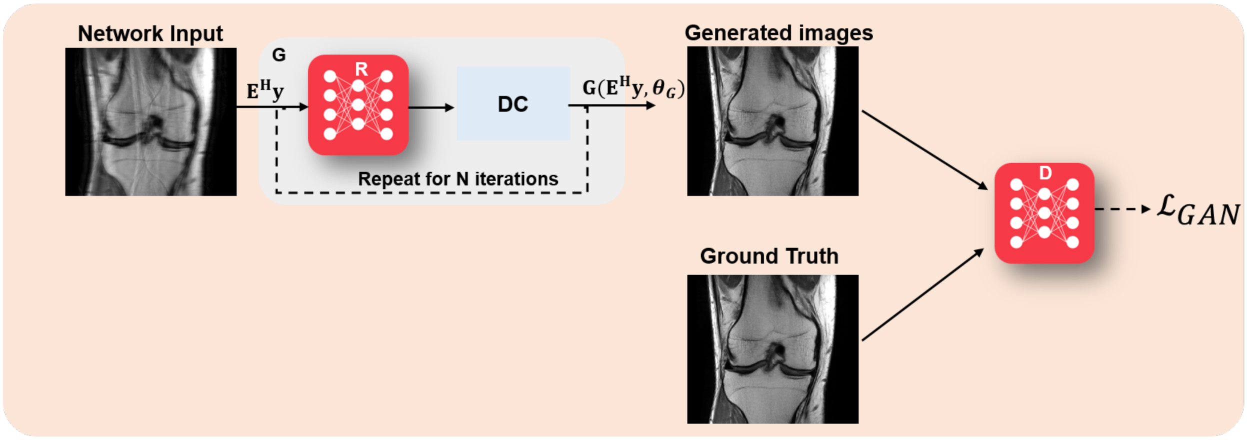

Fig. 2: Overview of GAN methods in physics-driven deep learning methods for computational MRI. A generator network (G), typically followed by a data consistency layer, implemented using a gradient descent step as in Eq. (8a), is used to generate an image. In the supervised setting, this generator is jointly trained with a discriminator network (D) that implements an adversarial loss to aid in the recovery of fine details of the image. In the unsupervised setting, such as cycleGANs, physics information is further enforced in the loss function both in image and k-space domains. Note that the figure shows the training phase, and at inference time, only the G network is used.

training. The physics-based information and the trained generator network $G_{\boldsymbol{\theta}_G}$ are then included in the optimization problem

$$\min_{\mathbf{z}} \frac{1}{2} \|\mathbf{E} G_{\boldsymbol{\theta}_G}(\mathbf{z}) - \mathbf{y}\|_2^2. \tag{16}$$

This solves for the latent vector $\mathbf{z}$ which is bounded from above by a hypersphere constraint, generating an image that lies in the range space of the generator. In a final step, both the generator parameters and the latent vector are optimized following:

$$\min_{\boldsymbol{\theta}_G, \mathbf{z}} \frac{1}{2} \|\mathbf{E} G_{\boldsymbol{\theta}_G}(\mathbf{z}) - \mathbf{y}\|_2^2, \tag{17}$$

This allows for adaptation of the generator to the undersampled k-space data at test time, and is not restricted to any sampling pattern. Initialization of (17) by the optimal latent vector found in (16) and early stopping (before reaching the minimum), allow the generator parameters $\boldsymbol{\theta}_G$ to not deviate too far from the original generator parameters. The reconstructed image $\mathbf{x}$ is obtained by using the optimized values $\boldsymbol{\theta}_G^*, \mathbf{z}^*$ for the generator, i.e., $\mathbf{x} = G_{\boldsymbol{\theta}\_G^*}(\mathbf{z}^*)$.

In another line of work, cycle consistent GANs (cycleGAN) that enable unpaired image-to-image translation, have been analyzed using optimal transport, which provides a means to transport probability measures by minimizing average transport between measures [20]. While traditional GANs and DIP-like networks are trained to minimize a distance measure in either image space or k-space, CycleGAN aims to minimize this in both k-space and image domain. In essence, this is achieved by minimizing two forms of losses, one for cyclic consistency and one for GAN training. The former is given by

$$\mathcal{L}_{\text{cycle}}(\boldsymbol{\theta}_G) = \mathbb{E}_{\mathbf{x}}[||\mathbf{x} - G_{\boldsymbol{\theta}_G}(\mathbf{E}\mathbf{x})||_2^2] + \mathbb{E}_{\mathbf{y}}[||\mathbf{y} - \mathbf{E}G_{\boldsymbol{\theta}_G}(\mathbf{y})||_2^2], \tag{18}$$

where the generator uses k-space measurements $\mathbf{y}$ as input. Here, the first term ensures consistency in the image domain, while the latter enforces consistency in the k-space domain. The second part of the training loss is a Wasserstein GAN loss,

$$\mathcal{L}_{\text{WGAN}}(\boldsymbol{\theta}_G, \boldsymbol{\theta}_D) = \max_{||D_{\boldsymbol{\theta}_D}||_L \leq 1} \mathbb{E}_{\mathbf{x}}[D_{\boldsymbol{\theta}_D}(\mathbf{x})] - \mathbb{E}_{\mathbf{z}}[D_{\boldsymbol{\theta}_D}(G_{\boldsymbol{\theta}_G}(\mathbf{z}))]. \tag{19}$$

This equation is a generalization of Eq. (14) with improved training stability, where the $D_{\boldsymbol{\theta}_D}$ now outputs a scalar value instead of a probability and as such is referred to as a critic instead of a discriminator, and $||D_{\boldsymbol{\theta}_D}||_L \leq 1$ indicates that it is restricted to be a Lipschitz-1 function. The final training loss is given as a weighted combination of these

$$\mathcal{L}_{\text{cycleGAN}} = \gamma \mathcal{L}_{\text{cycle}}(\boldsymbol{\theta}_G) + \mathcal{L}_{\text{WGAN}}(\boldsymbol{\theta}_G, \boldsymbol{\theta}_D), \tag{20}$$

where γ is a weighting hyperparameter. This approach was applied to unsupervised training of generative models for MRI reconstruction [20].

Variational Autoencoders (VAEs) build on the dimensionality-reducing encoder-decoder structure of autoencoders. Different from autoencoders, the encoder in a VAE learns a conditional distribution on the latent space, conditioned on the input distribution. Then, a vector is sampled from this probability distribution and fed to a decoder, which approximates the original data distribution conditioned on the latent space distribution. Hence, the latent code is learned in VAEs for a class of input images, while for conventional GANs, the latent vector amounts to random noise. Another application of VAEs in the field of MRI reconstruction is uncertainty quantification [18], where the VAE encodes the acquisition uncertainty and a Monte Carlo sampling approach is used to sample from the learned distribution and generate uncertainty maps along with the reconstructed image.

D. Algorithm unrolling and unrolled networks

Algorithm unrolling considers the traditional iterative approaches considered in Section II-A and adapts them in a manner that is amenable to learning the optimal parameters for image reconstruction [6]. Traditional approaches require numerous iterations during optimization to solve the MRI reconstruction problem. Additionally, only a fixed, handcrafted regularizer is used, which do not necessarily model MR images accurately. Instead of solving a new optimization problem for each task, the whole iterative reconstruction procedure, including the image regularizer, can be learned. The original idea was proposed in the context of sparse coding, but has found great use in computational imaging applications, including computational MRI. In this line of work, a conventional iterative algorithm for solving Eq. (6) is unrolled and solved for a fixed number of iterations, as overviewed in Figure 3. The concept of algorithm unrolling will be introduced throughout this section. In practice, any iterative optimization algorithm can be unrolled for solving Eq. (6). In the context of MRI, algorithm unrolling is based on ADMM as described in Eq. (9a)-(9c), gradient descent schemes [4], proximal gradient schemes [21], primal-dual methods [22], or variable splitting methods [5, 23]. Note that these algorithms contain a processing step associated with the regularization, such as the proximal operator as in Eq. (8b) or (9b), and a data consistency step that ensures the image estimate is consistent with the acquired k-space data, such as the gradient descent step in Eq. (8a) or the ℓ_2 minimization step in (9a). We refer to this latter step that controls fidelity with the raw k-space data as *data consistency layer* (or block).

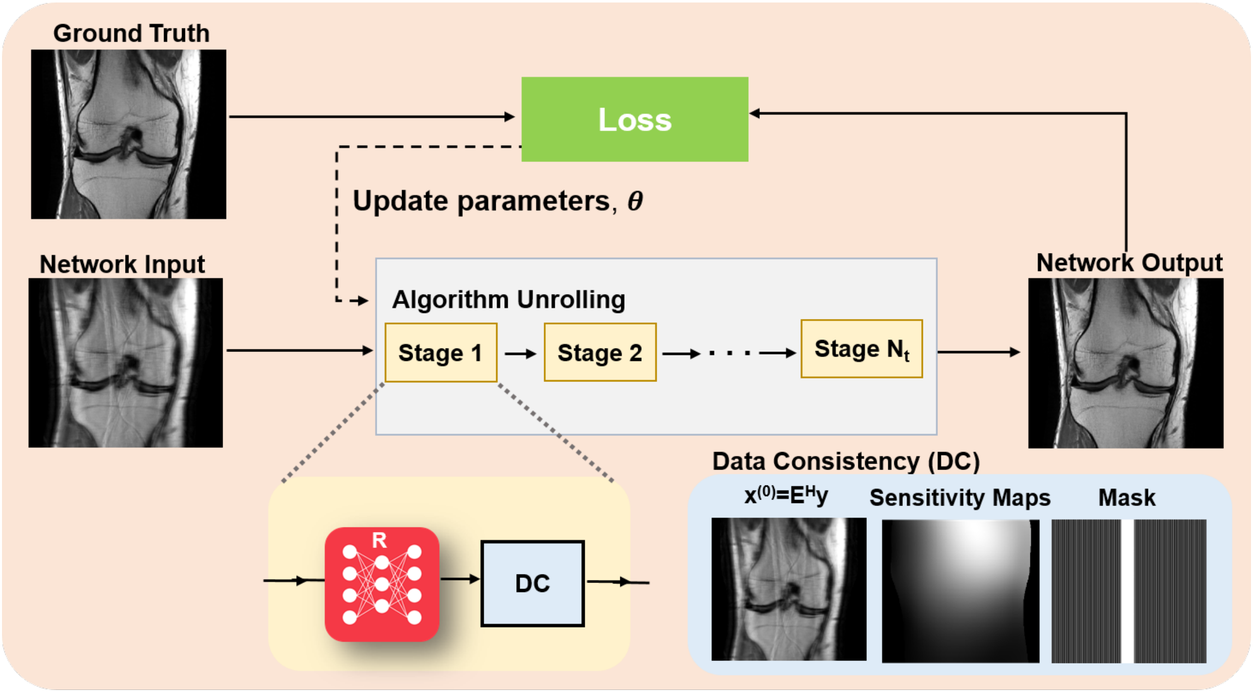

Fig. 3: Overview of algorithm unrolling in physics-driven deep learning methods for computational MRI. An iterative algorithm for solving Eq. (6) is unrolled for a fixed number of iterations, and trained end-to-end using corresponding fully-sampled/reference data. The red network block R denotes a regularization network. This is followed by a data consistency (DC) layer. The implementation of the DC layer depends on which algorithm is used for unrolling. If a gradient descent scheme is used, the DC layer implements a gradient descent update involving the raw k-space data, and other constants that are involved in the forward model. If variable splitting based methods are used, this involves solving a problem similar to Eq. (9a). The parameters can be either shared or vary over the single iterations, also termed stages.

We introduce the concept of unrolled networks on Variational Networks (VNs), which are an example for an unrolled gradient descent scheme. In this method, the gradient descent approach in Eq. (7) is unrolled for a fixed number of N_t steps. In VNs, the gradient of the regularizer $\nabla_{\mathbf{x}}\mathcal{R}(\mathbf{x})|_{\mathbf{x}=\mathbf{x}^{(i-1)}}$ is derived from the Fields-of-Experts (FoE) regularizer [4], i.e.,

$$\mathcal{R}(\mathbf{x}) = \sum_{j=1}^{N_k} \mathbf{\Phi}_j(\mathbf{K}_j\mathbf{x}).$$

This can be seen as a generalization of the Total Variation semi-norm for a number of N_k convolution operators $\mathbf{K}_j : \mathbb{C}^{N_x} \to \mathbb{C}^{N_x}$ and non-linear potential functions $\mathbf{\Phi}_j : \mathbb{C}^{N_x} \to \mathbb{R}$. Calculating the gradient with respect to $\mathbf{x}$ yields:

$$\nabla_{\mathbf{x}}\mathcal{R}(\mathbf{x}) = \sum_{j=1}^{N_k} \mathbf{K}_j^H \mathbf{\Phi}_j'(\mathbf{K}_j\mathbf{x}), \tag{21}$$

where $\mathbf{\Phi'}_j : \mathbb{C}^{N_x} \to \mathbb{C}^{N_x}$ denotes the gradient vector of $\mathbf{\Phi}_j$ with respect to $\mathbf{x}$. Plugging Eq. (21) in Eq. (7) yields

$$\mathbf{x}^{(i)} = \mathbf{x}^{(i-1)} + \eta\mathbf{E}^H(\mathbf{y} - \mathbf{E}\mathbf{x}^{(i-1)}) - \sum_{j=1}^{N_k} \mathbf{K}_j^H \mathbf{\Phi}_j'(\mathbf{K}_j\mathbf{x}^{(i-1)}) \quad i = 0, \dots N_t - 1, \tag{22}$$

where N_t denotes the number of cascaded stages, and one updated (i) denotes a single stage. The network is said to be *unrolled* for a fixed number of stages N_i for training. In Eq. (23) the trainable network parameters are the convolution operators $\mathbf{K}_j$, the activation functions $\mathbf{\Phi'}_j$ and the weight η. The parameters can be shared over stages or varied over stages. The activation functions $\mathbf{\Phi}'$ are modelled by a weighted combination of Gaussian radial basis functions, whose weights are learned, allowing us to approximate arbitrary activation functions.

$$\mathbf{x}^{(i)} = \mathbf{x}^{(i-1)} + \eta\mathbf{E}^H(\mathbf{y} - \mathbf{E}\mathbf{x}^{(i-1)}) - \sum_{j=1}^{N_k} \mathbf{K}_j^H \mathbf{\Phi}_j'(\mathbf{K}_j\mathbf{x}^{(i-1)}) \quad i = 0, \dots N_t - 1, \tag{23}$$

VNs are characterized by the energy-based formulation of the regularizer such as the FoE regularizer [4]. In other approaches, this energy-based formulation is discarded and the gradient with respect to $\mathbf{x}$ is replaced by a CNN with trainable parameters $\boldsymbol{\theta}$:

$$\nabla\mathcal{R}(\mathbf{x})|_{\mathbf{x}=\mathbf{x}^{(i-1)}} = \text{CNN}_{\boldsymbol{\theta}}(\mathbf{x}^{(i-1)}). \tag{24}$$

Another line of work considers the variable splitting approach in Eq. (10a)-(10b), used in data consistent CNNs [21] and MoDL [5], which again replace the gradient with respect to $\mathbf{x}$ by a CNN with trainable parameters $\boldsymbol{\theta}$ as in Eq. (24). This leads to the following scheme:

$$\mathbf{x}^{(i)} = \left(\mathbf{E}^H\mathbf{E} + \eta\mathbf{I}\right)^{-1}\left(\mathbf{E}^H\mathbf{y} + \eta\mathbf{z}^{(i)}\right), \tag{25a}$$

$$\mathbf{z}^{(i)} = \text{CNN}_{\boldsymbol{\theta}}(\mathbf{x}^{(i-1)}) \tag{25b}$$

where η is an additional learnable parameter. Eq. (25a) can be solved directly via matrix inversion for single-coil datasets [21], or using an iterative optimization approach based on conjugate gradient (CG) for the more commonly used multi-coil setup [5], where matrix inversion is computationally infeasible. Note in this case, the CG algorithm itself has to be unrolled for a fixed number of iterations for easy back-propagation through the whole network. Once again, the CNN in Eq. (25b) can be any kind of regularization network, as the idea is agnostic to the particulars of the CNN that is used in this step.

Proximal gradient descent unrolling, which utilizes Eq. (8a)-(8b), leads to the replacement of the proximal operator of $\mathcal{R}(\cdot)$ by a CNN with trainable parameters $\boldsymbol{\theta}$, leading to:

$$\mathbf{z}^{(i)} = \mathbf{x}^{(i-i)} + \eta \mathbf{E}^H(\mathbf{y} - \mathbf{E}\mathbf{x}^{(i-i)}). \tag{26a}$$

$$\mathbf{x}^{(i)} = \text{CNN}_{\boldsymbol{\theta}}(\mathbf{z}^{(i)}) \tag{26b}$$

This method was utilized in [21].

In the context of learned unrolled schemes, classical multi-layer CNNs [5, 21] or multi-scale regularizers such as UNET, Down-Up Networks [24], multi-level wavelet CNNs [22] are commonly used. Also, the parameters of these networks can be either shared, e.g. [5], or varied, e.g. [4], over the stages. However, similar performance has been achieved with both gradient descent and variable splitting-type algorithms as reported in the first fastMRI reconstruction challenge [25]. The differences reported in the context of the second fastMRI reconstruction challenge focus more on managing different coil sensitivities and regularization networks [26].

1) Training unrolled networks: The output of the unrolled network depends on the variables in both the regularization network and data consistency layers, and can be represented with a function $f_{\text{unroll}}(\mathbf{y}, \mathbf{E}; \{\boldsymbol{\theta}_i, \eta_i\}_{i=1}^{N_t})$. For the most generalized representation, we allow the regularizer CNN parameters $\boldsymbol{\theta}$ and the data consistency parameters η to vary across the unrolled iterations (cascaded stages). However, as noted earlier, the parameters can also be shared between stages. While for ease of notation, we have referred to the multi-coil operator as $\mathbf{E}$, this operator implicitly includes the sub-sampling mask Ω. For the following, we will make this dependence explicit, and use $\mathbf{E}_\Omega$ and $\mathbf{y}_\Omega$ for the multi-coil operator and the measured k-space data, respectively.

The standard learning strategy for unrolled networks is to train them end-to-end, using the full network that has been unrolled for N_t steps. For end-to-end training of unrolled networks, the most commonly used paradigm relies on supervised learning, where a database of fully-sampled measurements/ground-truth images as a reference. Given a database of pairs of input and reference data, the supervised learning loss function can be written as

$$\min_{\{\boldsymbol{\theta}_i,\eta_i\}_{i=1}^{N_t}} \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}(\mathbf{x}_{\text{ref}}^n,\ f(\mathbf{y}_{\Omega}^n,\mathbf{E}_{\Omega}^n;\{\boldsymbol{\theta}_i,\eta_i\}_{i=1}^{N_t})), \tag{27}$$

where $\boldsymbol{\theta}$ represents the network parameters, N is the number of samples in the training database, $\mathcal{L}(\cdot,\cdot)$ is a loss function characterizing the difference between network output and referenced data, $\mathbf{x}_{\text{ref}}^n$ denotes the ground-truth image for subject n. The domain for the loss function can be image, k-space or a mixture of them. Numerous loss functions such as ℓ_1, ℓ_2, adversarial and perceptual losses have been used in supervised deep learning approaches [6].

However, in many applications, fully-sampled reference data may be impossible to acquire, for instance due to organ motion or signal decay, or may be impractical due to excessively long scan times. In these cases, self-supervised learning enables training of neural networks without fully-sampled data by generating training data from the sub-sampled measurements themselves. One of the first works in this area, self-supervised learning via data undersampling (SSDU) [23], partitions the acquired measurements Ω, for each scan into two disjoint sets, Θ and Λ. One of these sets, Θ, is used during training to enforce data consistency within the network, while the other set, Λ, remains unseen by the unrolled network and is used to define the loss function in k-space. Hence, SSDU performs end-to-end training by minimizing the following self-supervised loss:

$$\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\Big(\mathbf{y}_{\Lambda}^n,\ \mathbf{E}_{\Lambda}^n\big(f(\mathbf{y}_{\Theta}^n,\mathbf{E}_{\Theta}^n;\boldsymbol{\theta})\big)\Big), \tag{28}$$

where the network output is transformed back to k-space by applying the encoding operator $\mathbf{E}_{\Lambda}^n$ at unseen locations in training. Thus, the self-supervised loss function measures the reconstruction quality of the model by characterizing the discrepancy between the unseen acquired measurements and network output measurements at the corresponding locations. Once the network is trained, the reconstruction for unseen test data is performed by using all acquired measurements Ω. In another line of work [27], Stein's unbiased risk estimate of mean square error (MSE) is leveraged to enable unsupervised training of neural networks for MRI reconstruction. In particular, the loss function obtained from an ensemble of images, each acquired by employing different undersampling operator, has been shown to be an unbiased estimator for MSE.

Finally, like generative models based on DIP, there has been interest in training unrolled networks on single datasets without a database. In this setting, the number of trainable parameters is usually larger than the number of pixels/k-space measurements, and training may lead to overfitting. Recent work in this area has tackled this challenge by developing a zero-shot self-supervised learning [28] approach that includes a third additional partition, which is used to monitor a self-validation loss in addition to the previous self-supervision setup. This self-validation loss starts to increase once overfitting is observed.

Once the model training is stopped, the final reconstruction is calculated by using the network parameters from the stopping epoch, while using all acquired measurements.

2) Memory challenges of unrolled networks and deep equilibrium networks: A major challenge for training unrolled networks is their large memory footprint. When an algorithm is unrolled for N_t iterations, a straightforward implementation involves the storage of N_t CNNs, along with N_t DC operations in GPU memory. The latter itself can have a large footprint, when a CG-type approach is used [5]. This creates challenges for training unrolled networks for large-scale or multi-dimensional datasets, especially since deeper networks tend to lead to better performance [25, 26]. Recently, this was tackled with the development of memory-efficient learning schemes [29]. In memory-efficient learning, intermediate outputs from each unrolled iteration are stored on host memory during forward pass, and backpropagation gradients are computed using this intermediate data and gradients from the preceeding step. Thus, this approach conceptually supports as many unrolling steps as desired, with the drawback of additional data transfer between GPU and the host memory.

Another alternative for handling the large memory footprint of unrolled networks is deep equilibrium networks [30]. Unrolled networks that share learnable weights across stages show competitive performance [5], while each stage can be represented with a single function leading to a compact representation. Unrolled networks execute this function for a finite number of steps N_t, whereas deep equilibrium networks characterize the limit as the $N_t \to \infty$. Provided this limit exists, it corresponds to the solution of the fixed point equation for an operator corresponding to a single stage. This approach leads to two advantages for training. First, only one stage has to be stored during training, leading to a smaller memory usage. Second, the convergence behavior for different values of N_t during inference is more well-behaved compared to unrolled networks, which are designed to achieve maximal performance for a specific value of N_t. On the other hand, deep equilibrium networks are run until convergence and do not have fixed inference time unlike unrolled networks, which may not be ideal in clinical applications. Furthermore, these approaches require large training time due to the Jacobian inversion during gradient computation.

IV. State-of-the-art in MRI practice and domain-specific challenges

A. *Real vs complex building blocks*

As complex-valued data is used in computational MRI, this has to be considered in the network processing pipeline, not only during data consistency, but also in the network blocks itself. Two processing modes are possible: 1) Real/Imaginary or magnitude/phase are considered in two input channels stacked via the feature dimension, 2) Complex-valued operations are performed on complex-valued tensors. While the former allows us to use real-valued operations, the complex-valued relationship between real and imaginary parts is lost. Complex-valued operations maintain the complex nature of the data, but some operations require twice the amount of trainable parameters. For example in complex-valued convolutions, a real and imaginary filter kernel needs to be learned. Additionally, the number of multiplications doubles compared to real-valued processing. If complex-valued layers and tensors are involved, complex back-propagation following Wirtinger calculus has to be considered [31] which is supported in most recent frameworks (Tensorflow $\geq$ v1.0, PyTorch $\geq$ v1.10). An overview of the most common layer operations together with their complex-valued Wirtinger derivatives is shown in Table II. In the context of MRI reconstruction, complex-valued processing is conducted in both ways.

TABLE II: Overview of important functions along with their pair of Wirtinger derivatives.

Function	$f(\mathbf{x})$	$\frac{\partial f}{\partial \mathbf{x}}$	$\frac{\partial f}{\partial \mathbf{x}^H}$
Magnitude	$\lvert\mathbf{x}\rvert = \left(\mathbf{x}^H\mathbf{x}\right)^{0.5}$	$\frac{\mathbf{x}^H}{2f(\mathbf{x})}$	$\frac{\mathbf{x}}{2f(\mathbf{x})}$
Phase	$-\mathrm{i}\log\frac{\mathbf{x}}{\lvert\mathbf{x}\rvert}$	$-\frac{\mathrm{i}}{2\mathbf{x}}$	$\frac{\mathrm{i}}{2\mathbf{x}^H}$
Real Component	$\frac{1}{2}\left(\mathbf{x}+\mathbf{x}^H\right)$	$\frac{1}{2}$	$\frac{1}{2}$
Imaginary Component	$\frac{1}{2\mathrm{i}}\left(\mathbf{x}+\mathbf{x}^H\right)$	$\frac{1}{2\mathrm{i}}$	$\frac{\mathrm{i}}{2}$
Normalization	$\frac{\mathbf{x}}{\left(\mathbf{x}^H\mathbf{x}\right)^{0.5}}$	$\frac{1}{2\left(\mathbf{x}^H\mathbf{x}\right)^{0.5}}$	$-\frac{z^2}{2\left(\mathbf{x}^H\mathbf{x}\right)^{1.5}}$
Scalar product	$\mathbf{w}^H\mathbf{x}$	$\mathbf{w}^H$	0
Max Pooling	$\mathbf{x}_n,\, n = \arg\max_k \lvert\mathbf{x}_k\rvert$	$\begin{cases} 1 & \text{if } n = \arg\max_k \lvert\mathbf{x}_k\rvert \\ 0 & \text{else} \end{cases}$	0
Dropout	$\begin{cases} \frac{1}{p}\mathbf{x}_n & \text{if } n \in \Omega \\ 0 & \text{else} \end{cases}$	$\begin{cases} \frac{1}{p} & \text{if } n \in \Omega \\ 0 & \text{else} \end{cases}$	0
Separable activation (ReLU, Sigmoid, ...)	$f(Re(\mathbf{x})) + \mathrm{i}f(Im(\mathbf{x}))$	$\frac{1}{2}\left(\frac{\partial f}{\partial \mathbf{x}}(Re(\mathbf{x})) + \frac{\partial f}{\partial \mathbf{x}}(Im(\mathbf{x}))\right)$	$\frac{1}{2}\left(\frac{\partial f}{\partial \mathbf{x}}(Re(\mathbf{x})) - \frac{\partial f}{\partial \mathbf{x}}(Im(\mathbf{x}))\right)$
Cardioid [31]	$\frac{1}{2}\left(1+\cos(\angle\mathbf{x})\right)\mathbf{x}$	$\frac{1}{2} + \frac{1}{2}\cos(\angle\mathbf{x}) + \frac{\mathrm{i}}{4}\sin(\angle\mathbf{x})$	$-\frac{\mathrm{i}}{4}\sin(\angle\mathbf{x})\frac{\mathbf{x}}{\mathbf{x}^H}$
Complex sigmoid	$\frac{1}{1+e^{-\mathbf{x}}}$	$\frac{e^{-\mathbf{x}}}{\left(1+e^{-\mathbf{x}}\right)^2}$	0

1) Convolution: The discrete convolution maps the $N_{f,\text{in}}$ input feature channels to $N_{f,\text{out}}$ output feature channels of an image $\mathbf{x} \in \mathbb{K}^n$ with filter kernels $\mathbf{k}_{i,j} \in \mathbb{K}^k$ of kernel size k via

$$\hat{\mathbf{x}}_j = \sum_{i=1}^{N_{f,\text{in}}} \mathbf{x}_i * \mathbf{k}_{i,j} \quad j = 1, \ldots, N_{f,\text{out}}, \tag{29}$$

where the subscripts denote the feature channels. Convolutions can be performed along multiple dimensions. For real-valued convolutions it is $\mathbb{K} = \mathbb{R}$. For complex convolutions, $\mathbb{K} = \mathbb{C}$, the convolution operation is extended to

$$\mathbf{x}_i * \mathbf{k}_{i,j} = (Re(\mathbf{x}_i) * Re(\mathbf{k}_{i,j}) - Im(\mathbf{x}_i) * Im(\mathbf{k}_{i,j})) \\ + \mathbf{i} \cdot (Im(\mathbf{x}_i) * Re(\mathbf{k}_{i,j}) + Re(\mathbf{x}_i) * Im(\mathbf{k}_{i,j})) \tag{30}$$

2) Activation: While convolution functions operate in a local neighborhood of a pixel, activation functions operate in a pixel-wise way. When applying non-linear activation functions ϕ to complex values, the impact on magnitude and phase information needs to be considered. The reader is referred to [31] for some comparative work. One possibility is to apply the activation function to the real and imaginary part separately as separable activations, however, the natural correlation between real and imaginary channels are not considered in this case. When using separable ReLUs, phase information is mapped to the first quadrant, i.e., the interval $[0, \frac{\pi}{2}]$, as all negative real and imaginary parts are set to zero and only the positive parts are kept. Alternative approaches have been proposed that retain phase information, for example siglog, defined as

$$\phi_{\text{siglog}} = \frac{\mathbf{x}}{1 + |\mathbf{x}|}.$$

As another option, the phase information can be fixed and only the magnitude information is altered by the activation. An example therefore is the *ModReLU*

$$\phi_{\text{ModReLU}}(\mathbf{x}) = \max(0, |\mathbf{x}| + \beta)\frac{\mathbf{x}}{|\mathbf{x}|} \tag{31}$$

where β is the bias that is trainable. A new complex activation function called *Cardioid* was also proposed for MRI processing [31]

$$\phi_{\text{Cardioid}}(\mathbf{x}) = \frac{1}{2}\left(1 + \cos(\angle\mathbf{x} + \beta)\right)\mathbf{x} \tag{32}$$

The complex cardioid can be seen as a generalization of ReLU activation functions to the complex plane. Compared to other complex activation functions, the complex cardioid acts on the input phase rather than the input magnitude. A bias β can be additionally learned.

The specific choice among these complex activation functions is application-dependent. For phase sensitive applications, such as water-fat imaging and phase contrast imaging, it was shown that complex networks outperformed the real-valued networks, with the separable ReLUs performing best, whereas for MR fingerprinting, the complex cardioid outperformed other activations functions [31].

3) Normalization: Adding normalization layers (batch, instance or layer/group normalization) directly after convolution layers are a common way to enable faster and more stable training of networks. Statistics are estimated from the input and used to re-parametrize the input. Complex-valued normalization layers require to estimate the normalization via the covariance matrix and are straight-forward to implement.

The subsequent layers are less tolerant to changes in previous layers. The selection of the normalization layer is task dependent. Although normalization layers are often important to train a network, they might lead to unwanted artifacts for image restoration tasks.

4) Pooling: Pooling layers are used as down-sampling operation to reduce the spatial resolution in the image and to introduce approximate invariance to small translations. Small patches are analyzed in the individual features maps to keep important information about extracted features. Common pooling layers are *Average Pooling* and *Max Pooling*. For complex-valued images, the maximum operation does not exist. Instead, the pooling layer is modified such that it keeps values with, e.g., the maximum magnitude response.

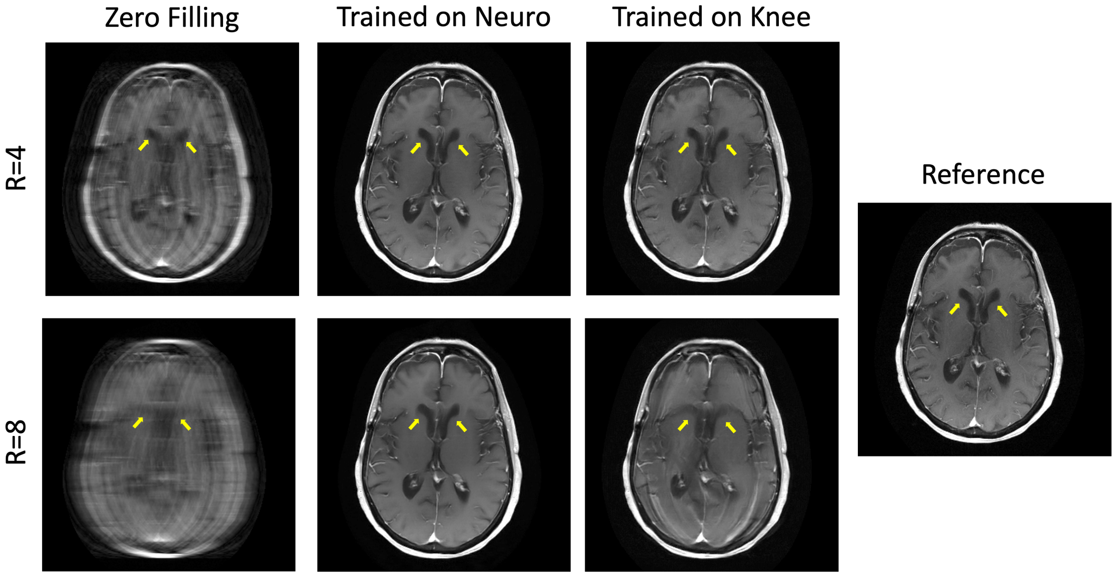

Fig. 4: Down-Up Networks combined with a proximal mapping layer for data consistency [24], trained with different data configuration. While the reconstruction performance generalizes well independent of the type of training data for R=4, the ventricles of the brain change here when trained with the wrong data, i.e., knee data for R=8.

B. Canonical MRI reconstruction with the linear forward model

Physics-driven MRI deep learning methods have become the most popular approach in computational MRI due to their improved robustness, especially for the accelerated MRI problem that relies on the linear forward model in Eq. (2). Such methods have been the top performers in community-wide reconstruction challenges, such as the fastMRI challenge [25, 26], for Cartesian sampling. The success for physics-driven learning for MRI reconstruction is not limited to the Cartesian sampling pattern. Promising results were also shown for non-Cartesian sampling schemes [15, 22]. These algorithms have in common that data consistency is included, and expressive regularization networks are used. Imaged pathologies do not need to be included in the training dataset as long as enough k-space data is available to guide the reconstruction to recover the pathology encoded in k-space. Additionally it was shown that the pathologies may appear or disappear depending on the selection of the undersampling pattern for a given number of sampled k-space lines [25]. Theoretical analysis, reader studies and uncertainty quantification are tools that might support us to identify the clinically possible acceleration limit.

However, even physics-driven deep learning methods face some challenges for accelerated MRI. The impact of domain shift, i.e., training and testing on different data was studied in [24], for different acceleration factors. All training and evaluation is based on the fastMRI knee and neuro datasets [25]. While for acceleration 4, the proposed Down-Up networks with varying data consistency layers generalize well for both anatomies, the type and amount of training data becomes more critical for acceleration factor 8. Since fewer data is available for data consistency at this acceleration, the networks start to reconstruct anatomical structures that are not real. When trained on a subset of knee data and applied to neuro data, the ventricles start resembling knee structures, for acceleration 8 as depicted in Figure 4.

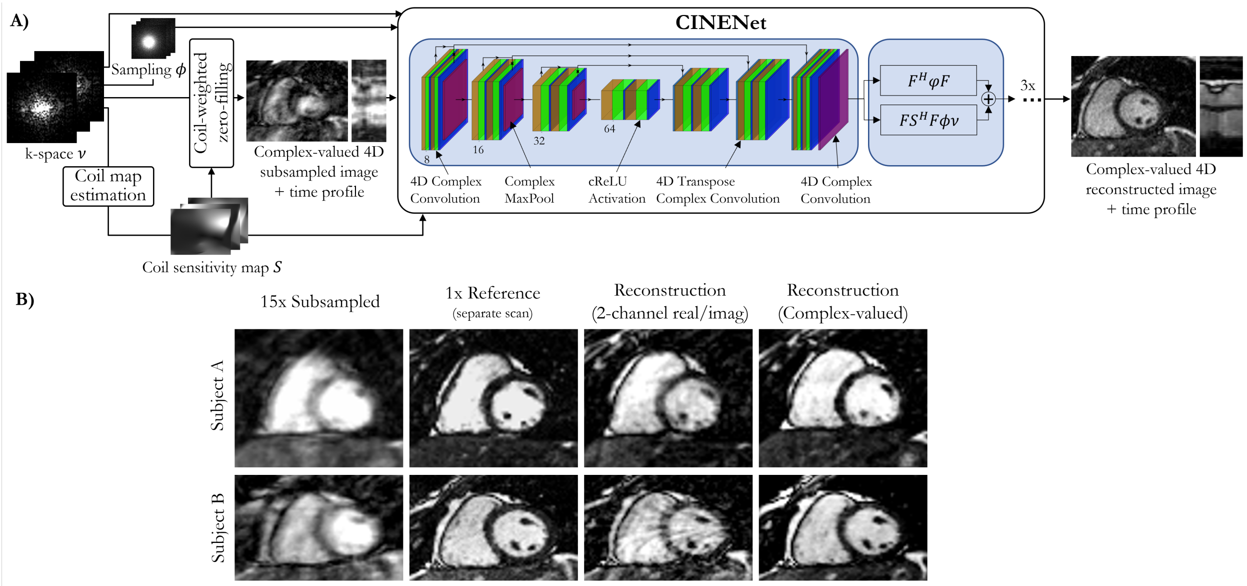

Fig. 5: A) CINENet combines data consistency layers and a UNet architecture with complex-valued building blocks for convolution, activation, normalization and pooling layers. To process the 3D+t data, convolutions are split into 3D spatial convolutions and 1D temporal convolutions. B) Impact of complex-valued operations over 2-channel (real/imaginary) processing in two subjects for a prospectively undersampled 3D cardiac CINE (3D+t) acquisition with R=15. A separate fully-sampled (R=1) reference scan is obtained for comparison.

All previously mentioned approaches consider the complex-valued MR images as images with two real-valued feature channels. CINENet [32] combined both data consistency layers with complex-valued building blocks as depicted in Figure 5, for dynamic 3D (3D+t) data. These complex-valued building blocks include convolutions, activations, pooling, and normalization layers. To process the 3D+t in the regularization network, convolution operations are split into 3D spatial convolutions, followed by 1D temporal convolutions.

Finally, as aforementioned the need for fully-sampled data for training had hindered the use of deep learning reconstructions for certain applications. Thus, alternative methods have been explored. Dynamic contrast-enhanced MRI (DCE-MRI) represents one such challenging acquisition, where k-space data is acquired continuously while contrast agent been injected to the patient. The dynamic distribution of the contrast agent causes the image contrast dynamics, hence, both the k-space and image are time-series. With the recent advances outlined in Section III-D1, training in such scenarios can also be done in with more realistic datasets without resorting to simulations. For instance, in another contrast-based cardiac acquisition, called late gadolinium enhancement imaging, unrolled networks have been trained using prospectively accelerated acquisitions without fully-sampled reference data [33], and were shown to improve on clinically used compressed sensing methods, doubling the achievable acceleration rates, as depicted in Figure 6.

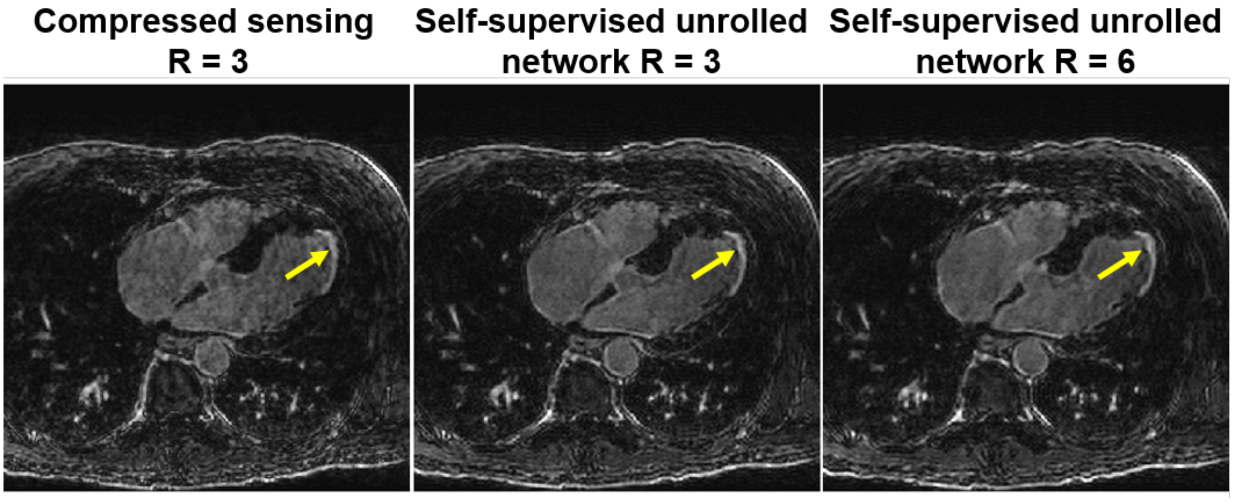

Fig. 6: Reconstruction results from a high-resolution late gadolinium enhancement acquisition on a cardiac patient (arrows: scar areas). This scan cannot be fully-sampled due to contrast-related scan time constraints. Unrolled networks can be trained in a self-supervised manner [23], leading to reconstructions that outperform current clinically used approaches, such as compressed sensing, and allowing acceleration rates twice as fast [33].

C. Inverse problems in MRI with non-linear forward models

Recently, deep learning models have been proposed to address the computationally demanding task of non-linear inverse problems in MRI. A neural network $f_{\text{NM},\boldsymbol{\theta}} : \mathbb{C}^{m \cdot n_c} \rightarrow \mathbb{C}^{n_v}$, parametrized by $\boldsymbol{\theta}$, which maps the acquired data $\mathbf{y}$ to the unknown parameters $\mathbf{v}$ (e.g. magnetization and relaxation maps) is learned either in a supervised setup [7]:

$$\arg\min_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{y}} \, \|\mathbf{E}\mathcal{M}(f_{\text{NM},\boldsymbol{\theta}}(\mathbf{y})) - \mathbf{y}\|_2^2 + \lambda || f_{\text{NM},\boldsymbol{\theta}}(\mathbf{y}) - \mathbf{v}_{\text{ref}}||_2^2 \tag{33}$$

or in a self-supervised setup

$$\arg\min_{\boldsymbol{\theta}} \mathbb{E}_{\mathbf{y}} \, \|\mathbf{E}\mathcal{M}(f_{\text{NM},\boldsymbol{\theta}}(\mathbf{y})) - \mathbf{y}\|_2^2 + \mathcal{R}(f_{\text{NM},\boldsymbol{\theta}}(\mathbf{y})), \tag{34}$$

where figure$\mathcal{R}(\cdot)$ is a conventional regularizer that is not based on reference data, such as spatial total variation. In the following, we will expand on some applications for which non-linear forward models are beneficial.

1) Relaxivity mapping: MRI allows for quantitative measurements of inherent tissue parameters (T_1, T_2, T_2^*, $\text{T}_{1\rho}$), which is often referred to as relaxivity or quantitative mapping. In recent years, research developments have contributed towards the goal of retrieving multiple parametric maps from a single scan [1]. A model-based reconstruction in these cases eliminates the need for reconstructing individual images along the relaxivity curve (data sampled at different time-points along the T_1/T_2/T_2^*/$\text{T}_{1\rho}$ relaxation based on Eq. (1) after excitation with appropriate preparatory pulses). However, model-based reconstruction methods have prolonged reconstruction times compared to reconstruction of individual images followed by a parametric fitting, which hinders their clinical translation. Deep learning models have been proposed to enable fast inference and shifting time-demanding workloads to the offline training procedure, showing feasibility in a number of quantitative mapping applications [7, 31]. In this setting, physics information, arising from the underlying known forward model (Eq. (2) or (3)), has primarily been incorporated to the loss function during training, similar to Eqs. (33) and (34) [7].

2) Susceptibility mapping: Physics-driven deep learning methods have also been studied in the context of quantitative susceptibility mapping. First works incorporated the physical principles of the dipole inversion model that describes the susceptibility-phase relationship into the loss function during neural network training [9]. More recently, the idea of fine-tuning pre-trained network weights on a scan-specific basis using the physics model was proposed [34], similar to the loss function in Eq. (34) without the additional regularizers.

3) Motion: Acquisitions under physiological and patient motion require methods for handling motion in order to avoid aliasing or blurring of the imaged anatomy. In addition to various prospective motion triggering, gating or correction methods, motion can be retrospectively modeled into the forward model and can thus be considered inside a motion-compensated/corrected reconstruction [2]. These methods perform two fundamental operations: image registration and image reconstruction. Hence, they require reliable motion-resolved images from which the motion can be estimated. Motion field estimation can be controlled or supported by external motion surrogate signals or initial motion field estimates [2].

While deep learning allows for efficient motion estimation, only few works embed motion estimation in image reconstruction. Among these, LAPNet formulates non-rigid registration directly in k-space [35], inspired by the optical flow formulation. The estimated motion fields are then used to enhance the data consistency and exploit the information of all motion resolved states to reconstruct images of the body trunk. In the context of coronary MRI, a motion-informed MoDL network was proposed [8], using diffeomorphic motion fields estimated from the zero-filled images using a UNet and subsequent scaling and squaring layer. These motion fields are then embedded into the data consistency layer, solved via the conjugate gradient algorithm as in MoDL. The network is unrolled for 3 iterations, with intermittent denoising networks. The full model is trained using a reconstruction loss and a motion estimation loss. Hence, both reconstruction and motion estimation improve as the motion-estimation networks rely on the reconstructions of the previous unrolled iteration. Another approach achieved motion correction by rejecting motion-affected k-space lines [36]. Inspired by [2], warping with a motion field is embedded in the forward operator for Cartesian cine imaging, where the motion fields are estimated by a neural network [37]. An example reconstruction results of the systole and diastole for accelerations R=4 and R=8 is depicted in Figure 7.

V. Discussion

A. Issues and open problems

Deep learning has dominated research in computational MRI during the last few years, and while there are still a number of open questions and issues, both on the basic science and on the translational front, they evolved as the developments are going on in the field. During the early stages of the developments, access to raw MRI k-space training data was a major limiting factor hat held the field back. The availability of open databases has largely removed this obstacle and public research challenges have also helped to compare developed approaches on standardized datasets [25, 26]. However, they have also highlighted new issues. While it was demonstrated that deep learning models generally outperform handcrafted regularizers in iterative image reconstruction in terms of quantitative metrics like SSIM, PSNR and RMSE, their performance in the regime of over-regularization (when the influence of the prior becomes dominant because of the low-quality of the data) is challenging to assess. The results of classic regularizers

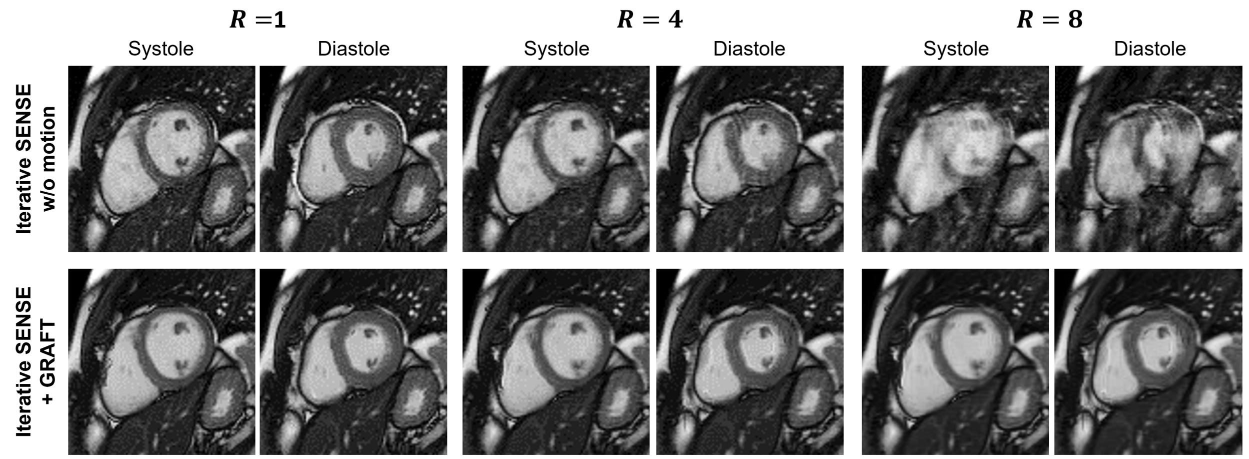

Fig. 7: Motion-Compensated reconstruction. A motion estimation network (GRAFT) is embedded in the reconstruction procedure [37]. The motion-compensated reconstruction outperforms iterative SENSE without motion compensation. If motion compensation is not performed, undersampling artifacts are substantially present. Systole and diastole frames are depicted for R=1, R=4 and R=8.

like Tikhonov, total variation or ℓ_1-wavelets in this scenario can be interpreted much easier by end-users. They lead to very characteristic artifact patterns that are easy to spot as being technical artifacts. Deep learning models have the computational capacity to generate realistic-looking images with either missing or artificially hallucinated image features [24] that are inconsistent with the measurement data if they are used at acceleration levels that are too high with respect to the encoding capabilities of the multi-element receive coil.

A solution is to move from qualitative image assessment towards the assessment of clinical outcomes. In particular, does the diagnostic quality improve for patients when deep learning methods are used instead of handcrafted regularizers? However, conducting such clinical studies is slow and costly, and in many cases an imaging exam cannot even be considered to be a true ground truth, which requires follow-ups with pathology or surgery departments.

Research challenges are also limited in terms of their ability to assess model generalization. The 2020 fastMRI challenge [26] included a track that specifically evaluated generalization with respect to deploying a trained model at a scanner from a different manufacturer. While the winning models performed well in this test, the performance of some approaches was impacted negatively by trivial modifications in the data, for example whether raw data is saved with oversampling in the readout direction or not. In light of the substantial range of imaging parameters that can be changed during an MRI acquisition, it is still an open question if deep learning models should serve a general purpose role, or if specialized models should be tailored to a more narrow range of imaging settings for dedicated exams.

Another open issue of almost all developments is that they are performed with retrospectively accelerated acquisitions, i.e., the accelerated acquisition is obtained by applying a simulated undersampling on the fully sampled dataset. While this is acceptable if true k-space raw data is used and no subtle data crime is performed [38], not all MR-signal-acquisition effects are captured with retrospective undersampling. In particular, spin-history, gradient and RF-hardware related effects seen in prospectively accelerated acquisitions, i.e. when data is acquired with true undersampling on the scanner, are usually not captured in retrospective acceleration. This can cause issues when moving to prospectively accelerated acquisitions on real MR-scanners. However, it should be pointed out that this is a general issue of all computational imaging methods that are developed retrospectively, and not a unique issue of deep learning techniques.

While physics-based learning for MRI reconstruction has been successfully established over the past years, there are some potential pitfalls and limitations of these approaches in practice. or instance, when DICOM images are used for experiments instead of raw k-space data, learning-based approaches may lead to overall optimistic results, while real-world unprocessed data performs much worse, resulting in biased state-of-the-art results [38]. In another line of work, the stability of various single-coil networks to small adversarial perturbations at their input was studied [39], and it was found that networks may exhibit large perturbations at the output. Furthermore, the definition of acceleration factor might also often be misleading. As shown in [25], different sampling patterns yield different results, which opens the question how potential mis-reconstructions can be identified and how the uncertainty of the reconstructions regarding the sampling pattern can be quantified. Finally, we would like to note that the reported undersampling factor and acceleration rates have to be carefully investigated. In the Cartesian setting, undersampling factor and acceleration rates are equivalent and defined by the number of sampled lines divided by the number of total lines in k-space. However, in the non-Cartesian case, the overall effective undersampling depends on the size of oversampling performed along the readout trajectories, without affecting the acceleration factor between the readouts.

B. Domain-specific knowledge in post-processing and multi-task imaging

The medical imaging pipeline consists of many tasks that are mostly viewed separately. The imaging pipeline starts with data acquisition, followed by image reconstruction. The reconstructed image is then further analyzed using post-processing tasks, image segmentation, and quantitative evaluation, and/or methods for diagnosis and treatment planning are applied to facilitate medical decisions.

Thus, there have been efforts to combine several of these tasks into a multi-task imaging framework. Most work on solving multiple tasks jointly has been conducted in the field of motion-corrected image reconstruction, as summarized in Section IV-C3. In [36], joint motion detection, correction and segmentation was proposed. In contrast to the previously mentioned approaches, the motion was detected directly in k-space and, hence, influence the data consistency layer. Additionally, a bidirectional recurrent CNN (BCRNN) was used to account for spatio-temporal redundancies. The motion-corrected image was obtained by cascading 10 data consistency layers and BCRNNs. Afterwards, a UNet was applied for cardiac segmentation. Evaluated on the UK Biobank data, this work showed that training a joint network for reconstruction and segmentation outperforms sequential training of these networks.

A unified network for joint MRI reconstruction and segmentation was also proposed [40]. For image reconstruction, an unrolled network with alternating data consistency layers and denoising networks are used. The denoising networks are based on an encoder-decoder structure, where the encoder is shared with the image segmentation network. Hence, common features are extracted using the encoder, while the decoder adapts to the underlying task. Evaluation and simulation of k-space is performed on the MRBrainS segmentation challenge dataset. Their results suggests that high-quality segmentation benefits from this multi-task architecture. While this method both optimizes for image reconstruction and another downstream task such as segmentation, it is still an open question if intermediately reconstructed images are needed, or if one could directly obtain, e.g., segmentation in k-space.

VI. Conclusion

Physics-driven deep learning techniques are the current state-of-the-art methods for computational MRI. Spanning methods that incorporate physics of MRI acquisitions into loss functions to plug-and-play techniques, and generative models to unrolled networks, a large number of approaches have been proposed to improve the solution of linear and non-linear inverse problems that arise in MRI. These methods are starting to make their way into translational and clinical settings, while also potentially altering the downstream tasks in the medical imaging pipeline. Thus, there are numerous opportunities for new technical developments and applications in physics-driven computational MRI from the signal processing community.

References

[1] M. Doneva, “Mathematical models for magnetic resonance imaging reconstruction: An overview of the approaches, problems, and future research areas,” *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 24–32, 2020.

[2] P. Batchelor, D. Atkinson, et al., “Matrix description of general motion correction applied to multishot images,” *Magn Reson Med*, vol. 54, no. 5, pp. 1273–1280, 2005.

[3] K. P. Pruessmann, M. Weiger, M. B. Scheidegger, and P. Boesiger, “SENSE: Sensitivity encoding for fast MRI,” *Magn Reson Med*, vol. 42, no. 5, pp. 952–962, 1999.

[4] K. Hammernik, T. Klatzer, et al., “Learning a variational network for reconstruction of accelerated MRI data,” *Mag Reson Med*, vol. 79, pp. 3055–71, 2018.

[5] H. K. Aggarwal, M. P. Mani, and M. Jacob, “MoDL: Model-based deep learning architecture for inverse problems,” *IEEE Trans Med Imaging*, 2019.

[6] F. Knoll, K. Hammernik, et al., “Deep-learning methods for parallel magnetic resonance imaging reconstruction,” *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 128–140, 2020.

[7] F. Liu, L. Feng, and R. Kijowski, “MANTIS: Model-Augmented Neural neTwork with Incoherent k-space Sampling for efficient MR parameter mapping,” *Magn Reson Med*, vol. 82, no. 1, pp. 174–188, 2019.

[8] H. Qi, R. Hajhosseiny, et al., “End-to-end deep learning nonrigid motion-corrected reconstruction for highly accelerated free-breathing coronary MRA,” *Magn Reson Med*, vol. 86, no. 4, pp. 1983–1996, 2021.

[9] J. Yoon, E. Gong, et al., “Quantitative susceptibility mapping using deep neural network: QSMnet,” *Neuroimage*, vol. 179, pp. 199–206, 2018.

[10] J. A. Fessler, “Optimization methods for magnetic resonance image reconstruction: Key models and optimization algorithms,” *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 33–40, 2020.

[11] C. M. Hyun, H. P. Kim, S. M. Lee, S. M. Lee, and J. K. Seo, “Deep learning for undersampled MRI reconstruction,” *Physics in Medicine and Biology*, vol. 63, no. 13, pp. 135007, 2018.

[12] G. Yang, S. Yu, et al., “DAGAN: Deep de-aliasing generative adversarial networks for fast compressed sensing mri reconstruction,” *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1310–1321, 2017.

[13] M. Akçakaya, S. Moeller, S. Weingartner, and K. Ugurbil, “Scan-specific robust artificial-neural-networks for k-space interpolation (RAKI) reconstruction: Database-free deep learning for fast imaging,” *Magn Reson Med*, vol. 81, no. 1, pp. 439–453, Jan 2019.

[14] R. Ahmad, C. A. Bouman, et al., “Plug-and-play methods for magnetic resonance imaging: Using denoisers for image recovery,” *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 105–116, 2020.

[15] J. Liu, Y. Sun, et al., “RARE: Image reconstruction using deep priors learned without ground truth,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 6, pp. 1088–1099, 2020.

[16] J. Yoo, K. H. Jin, et al., “Time-dependent deep image prior for dynamic MRI,” *IEEE Transactions on Medical Imaging*, vol. 40, no. 12, pp. 3337–3348, 2021.

[17] M. Mardani, E. Gong, et al., "Deep generative adversarial neural networks for compressive sensing MRI," *IEEE Transactions on Medical Imaging*, vol. 38, no. 1, pp. 167–179, 2019.

[18] V. Edupuganti, M. Mardani, S. Vasanawala, and J. Pauly, "Uncertainty quantification in deep MRI reconstruction," *IEEE Transactions on Medical Imaging*, vol. 40, no. 1, pp. 239–250, 2021.

[19] D. Narnhofer, K. Hammernik, F. Knoll, and T. Pock, "Inverse GANs for accelerated MRI reconstruction," in *Wavelets and Sparsity XVIII*, D. V. D. Ville, M. Papadakis, and Y. M. Lu, Eds. International Society for Optics and Photonics, 2019, vol. 11138, pp. 381 – 392, SPIE.

[20] G. Oh, B. Sim, H. Chung, L. Sunwoo, and J. C. Ye, "Unpaired deep learning for accelerated MRI using optimal transport driven cyclegan," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 1285–1296, 2020.

[21] J. Schlemper, J. Caballero, J. V. Hajnal, A. N. Price, and D. Rueckert, "A deep cascade of convolutional neural networks for dynamic MR image reconstruction," *IEEE Trans Med Imaging*, vol. 37, no. 2, pp. 491–503, 2018.

[22] Z. Ramzi, G. R. Chaithya, J.-L. Starck, and P. Ciuciu, "NC-PDNet: A density-compensated unrolled network for 2D and 3D non-Cartesian MRI reconstruction," *IEEE Transactions on Medical Imaging*, pp. 1–1, 2022.

[23] B. Yaman, S. A. H. Hosseini, et al., "Self-supervised learning of physics-guided reconstruction neural networks without fully-sampled reference data," *Magn Reson Med*, vol. 84, no. 6, pp. 3172–3191, Dec 2020.

[24] K. Hammernik, J. Schlemper, et al., "Systematic evaluation of iterative deep neural networks for fast parallel MRI reconstruction with sensitivity-weighted coil combination," *Magn Reson Med*, vol. 86, no. 4, pp. 1859–1872, 2021.

[25] P. M. Johnson, G. Jeong, et al., "Evaluation of the robustness of learned MR image reconstruction to systematic deviations between training and test data for the models from the fastMRI challenge," in *Machine Learning for Medical Image Reconstruction*, pp. 25–34. 2021.

[26] M. J. Muckley, B. Riemenschneider, et al., "Results of the 2020 fastMRI challenge for machine learning MR image reconstruction," *IEEE Trans. Medical Imaging*, vol. 40, no. 9, pp. 2306–2317, 2021.

[27] H. K. Aggarwal, A. Pramanik, and M. Jacob, "ENSURE: Ensemble Stein's unbiased risk estimator for unsupervised learning," in *ICASSP*. 2021, pp. 1160–1164, IEEE.

[28] B. Yaman, S. A. H. Hosseini, and M. Akcakaya, "Zero-shot self-supervised learning for MRI reconstruction," in *International Conference on Learning Representations*, 2022.

[29] M. Kellman, K. Zhang, et al., "Memory-efficient learning for large-scale computational imaging,"

IEEE Trans Comp Imaging, vol. 6, pp. 1403–1414, 2020.

[30] D. Gilton, G. Ongie, and R. Willett, "Deep equilibrium architectures for inverse problems in imaging," *IEEE Transactions on Computational Imaging*, vol. 7, pp. 1123–1133, 2021.

[31] P. Virtue, S. X. Yu, and M. Lustig, "Better than real: Complex-valued neural nets for MRI fingerprinting," *IEEE International Conference on Image Processing*, vol. 2017-Septe, pp. 3953–3957, 2018.

[32] T. Küstner, N. Fuin, et al., "CINENet: deep learning-based 3D cardiac CINE MRI reconstruction with multi-coil complex-valued 4D spatio-temporal convolutions," *Sci Rep*, vol. 10, no. 13710, 2020.

[33] B. Yaman, C. Shenoy, et al., "Self-supervised physics-guided deep learning reconstruction for high-resolution 3D LGE CMR," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 2021, pp. 100–104.

[34] J. Zhang, Z. Liu, et al., "Fidelity imposed network edit (FINE) for solving ill-posed image reconstruction," *NeuroImage*, vol. 211, pp. 116579, 2020.

[35] T. Küstner, J. Pan, et al., "LAPNet: Non-rigid registration derived in k-space for magnetic resonance imaging," *IEEE Transactions on Medical Imaging*, vol. 40, no. 12, pp. 3686–3697, 2021.

[36] I. Oksuz, J. R. Clough, et al., "Deep learning-based detection and correction of cardiac MR motion artefacts during reconstruction for high-quality segmentation," *IEEE Transactions on Medical Imaging*, vol. 39, pp. 4001–4010, 12 2020.

[37] K. Hammernik, J. Pan, D. Rueckert, and T. Küstner, "Motion-guided physics-based learning for cardiac MRI reconstruction," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, pp. 900–907.

[38] E. Shimron, J. I. Tamir, K. Wang, and M. Lustig, "Implicit data crimes: Machine learning bias arising from misuse of public data," *Proc Natl Acad Sci U S A*, vol. 119, no. 13, pp. e2117203119, 03 2022.

[39] V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen, "On instabilities of deep learning in image reconstruction and the potential costs of ai," *Proc Natl Acad Sci U S A*, vol. 117, no. 48, pp. 30088–30095, 2020.

[40] L. Sun, Z. Fan, X. Ding, Y. Huang, and J. Paisley, "Joint CS-MRI reconstruction and segmentation with a unified deep network," *Information Processing in Medical Imaging*, vol. 11492 LNCS, pp. 492–504, 6 2019.