bioengineering

MDPI

Article

Clean Self-Supervised MRI Reconstruction from Noisy, Sub-Sampled Training Data with Robust SSDU

Charles Millard [1] and Mark Chiew [2,3,*]

[1] Wellcome Centre for Integrative Neuroimaging, FMRIB, University of Oxford, Oxford OX3 9DU, UK
[2] Department of Medical Biophysics, University of Toronto, Toronto, ON M4N 3M5, Canada
[3] Physical Sciences, Sunnybrook Research Institute, Toronto, ON M4N 3M5 Canada
* Correspondence: mark.chiew@utoronto.ca

Abstract: Most existing methods for magnetic resonance imaging (MRI) reconstruction with deep learning use fully supervised training, which assumes that a fully sampled dataset with a high signal-to-noise ratio (SNR) is available for training. In many circumstances, however, such a dataset is highly impractical or even technically infeasible to acquire. Recently, a number of self-supervised methods for MRI reconstruction have been proposed, which use sub-sampled data only. However, the majority of such methods, such as Self-Supervised Learning via Data Undersampling (SSDU), are susceptible to reconstruction errors arising from noise in the measured data. In response, we propose Robust SSDU, which provably recovers clean images from noisy, sub-sampled training data by simultaneously estimating missing k-space samples and denoising the available samples. Robust SSDU trains the reconstruction network to map from a further noisy and sub-sampled version of the data to the original, singly noisy, and sub-sampled data and applies an additive Noisier2Noise correction term upon inference. We also present a related method, Noiser2Full, that recovers clean images when noisy, fully sampled data are available for training. Both proposed methods are applicable to any network architecture, are straightforward to implement, and have a similar computational cost to standard training. We evaluate our methods on the multi-coil fastMRI brain dataset with novel denoising-specific architecture and find that it performs competitively with a benchmark trained on clean, fully sampled data.

Keywords: deep learning; image reconstruction; magnetic resonance imaging

check for updates

Citation: Millard, C.; Chiew, M. Clean Self-Supervised MRI Reconstruction from Noisy, Sub-Sampled Training Data with Robust SSDU. *Bioengineering* **2024**, *11*, 1305. https://doi.org/10.3390/bioengineering11121305

Academic Editor: Yu Tian

Received: 30 November 2024
Revised: 14 December 2024
Accepted: 16 December 2024
Published: 23 December 2024

1. Introduction

Magnetic resonance imaging (MRI) has excellent soft tissue contrast and is the gold standard modality for a number of clinical applications. A hindrance of MRI, however, is its lengthy acquisition time, which is especially challenging when high spatio-temporal resolution is required, such as for dynamic imaging [1]. To address this, there has been substantial research attention on methods that reduce the acquisition time without significantly sacrificing the diagnostic quality [2–4]. In MRI, measurements are acquired in the Fourier representation of the image, referred to in the MRI literature as "k-space". Since the acquisition time is roughly proportional to the number of k-space samples, acquisitions can be accelerated by sub-sampling. A reconstruction algorithm is then employed to estimate the image from the sub-sampled data.

In recent years, reconstructing sub-sampled MRI data with neural networks has emerged as a state-of-the-art method [5–7]. The majority of existing methods assume that a fully sampled dataset is available for fully supervised training. However, for many applications, no such dataset is available and may be difficult or even infeasible to acquire in practice [8–10]. In response, there have been a number of self-supervised methods proposed, which train on sub-sampled data only [11–14]. Many such methods have shown promise in a broad range of clinical applications where fully sampled data are challenging to acquire, including dynamic imaging [15], late gadolinium enhancement cardiac imaging [16], simultaneous multi-slice functional imaging [17], and multi-contrast imaging [18].

Most existing training methods assume that the measurement noise is small and does not explicitly denoise sampled data. Section 3 shows theoretically that without explicit denoising, the reconstruction quality degrades when the measurement noise increases. This is a particular concern for low SNR measurements, where the SNR is the ratio of the signal and noise amplitudes, and the SNR is considered “low” when the measurement noise contributes substantially to the difference between the noisy, sub-sampled data and the ground truth. For instance, the data acquired from low-cost, low-field scanners are considered having a low SNR [19–21].

The goal of this paper is to develop a theoretically rigorous, computationally efficient approach for simultaneous self-supervised reconstruction and denoising that performs comparably to fully supervised training. The primary challenge of this goal is that many existing self-supervised denoising methods are not applicable to data that are also sub-sampled [22], depend on paired instances of noisy data [23], or are substantially computationally more expensive than fully supervised learning in training time [24].

This paper proposes a modification of Self-Supervised Learning via Data Undersampling (SSDU) [13] that also removes measurement noise, building on the present authors’ recent work [25] on the connection between SSDU and the multiplicative version of the self-supervised denoising method Noisier2Noise [26]. Our method, which we term “Robust SSDU”, combines SSDU with the *additive* Noisier2Noise. In brief, Robust SSDU trains a network to map from a further sub-sampled and further noisy version of the training data to the original sub-sampled, noisy data. Then, upon inference, a correction is applied to the network output that ensures that the clean (i.e., noise-free) image is recovered as expected.

We find that Robust SSDU performs competitively with a fully supervised benchmark where the network is trained on clean, fully sampled data, despite training on noisy, sub-sampled data only. We also propose a related method that recovers clean images for the simpler task of noisy data being available for training when fully sampled, which we term “Noisier2Full”. Both Noisier2Full and Robust SSDU are fully mathematically justified and have minimal additional computational expenses compared to standard training.

The existing method most similar to Robust SSDU is Noise2Recon-Self-Supervised (Noise2Recon-SS) [27]. The proposed method, Robust SSDU, has a number of key difference to Noise2Recon-SS, including a loss weighting and an additive Noisier2Noise correction term upon inference that statistically guarantees the recovery of the ground truth; see Section 4.3 for a detailed comparison. To our knowledge, Robust SSDU is the first method that provably recovers clean images when only noisy, randomly sub-sampled data are available for training. In practice, we find that Robust SSDU offers substantial image quality improvements over Noise2Recon-SS and a two-fold reduction in computational cost during training; see Section 5.

Notation

This paper uses notation consistent with [25]. We use the subscripts t and s to index the training set $\mathcal{T}$ and test set $\mathcal{S}$, respectively. For instance, data in the training and test set are denoted by y_t and y_s, respectively. Random variables are represented as their instances without indices and are capitalized if they are vectors. For instance, $y_t, y_s \backsim Y$ for vectors and $M_{\Omega_t}, M_{\Omega_s} \backsim M_\Omega$ for matrices, where $\backsim$ denotes that the left-hand side is an instance of the random variable on the right-hand side.

We use Y_0 to refer to the ground truth, Y to refer to the data, $\widetilde{Y}$ to refer to the further corrupted data, and $\hat{Y}$ to refer to an estimate of the ground truth. We note that Sections 2.1, 2.2, and 3 onward discuss different recovery tasks, so the definitions of, for instance, the data, Y, and their instances are section-specific.

2. Theory: Background

Image recovery with deep learning is a regression problem, so it is centered around the conditional distribution $Y_0|Y$, where Y_0 and Y are the random variables associated with the ground truth and data, respectively [28]. If the ground truth data $y_{0,t} \backsim Y_0$ are available for training, fully supervised learning can be employed to characterize $Y_0|Y$ directly [29]. This paper focuses on self-supervised learning, which concerns the task of training a network to estimate the ground truth when the training data are $y_t \backsim Y$ so are themselves corrupted [23,24,30,31].

The remainder of this section reviews key works from the self-supervised learning literature that form the bases of the methods proposed in this paper. Section 2.1 presents the case where the data corruption is Gaussian noise, and Section 2.2 presents the case where the data corruption is sub-sampling.

2.1. Self-Supervised Denoising with Noisier2Noise

Denoising with deep learning aims to recover a clean q-dimensional vector from noisy data:

$$y_s = y_{0,s} + n_s, \tag{1}$$

where n_s is noise and $s \in \mathcal{S}$ indexes the test set. In MRI, noise in k-space is modeled as a complex zero-mean Gaussian, $n_s \backsim \mathcal{CN}(0, \Sigma_n^2)$, where Σ_n^2 is a covariance matrix that can be estimated, for instance, with an empty pre-scan [32]. We treat the noise as white, $\Sigma_n^2 = \sigma_n^2 \mathbb{1}$, noting that noise with non-trivial covariance can be whitened by left-multiplying y_s with the square root inverse of Σ_n^2, denoted by Σ_n^{-1}. Other noise distributions are discussed in Section 6.

This paper focuses on the additive Noisier2Noise [26] because we find that it offers a natural way to extend image reconstruction to low-SNR data; see Section 3. Noisier2Noise's training procedure consists of corrupting noisy training data with further noise and training a network to recover a singly noisy image from a noisier image. Concretely, for each y_t, further noise is introduced:

$$\widetilde{y}_t = y_t + \widetilde{n}_t = y_{0,t} + n_t + \widetilde{n}_t, \tag{2}$$

where $\widetilde{n}_t \backsim \mathcal{CN}(0, \alpha^2 \sigma_n^2 \mathbb{1})$ for a constant α. Then, a network f_θ with parameters θ is trained to minimize the sum

$$\hat{\theta} = \arg\min_\theta \sum_{t \in \mathcal{T}} \|f_\theta(\widetilde{y}_t) - y_t\|_2^2 \tag{3}$$

where the symbol $\sum$ is used exclusively for summation herein. The following result states that a simple transform of the trained network yields the ground truth in expectation despite never seeing the ground truth during training. Here, and throughout this paper, expectations are taken over all random variables.

Result 1. *Consider the random variables $Y = Y_0 + N$ and $\widetilde{Y} = Y + \widetilde{N}$, where N and $\widetilde{N}$ are zero-mean Gaussians distributed with variances of σ_n^2 and $\alpha^2\sigma_n^2$, respectively. Minimizing*

$$\theta^* = \arg\min_\theta \mathbb{E}[\|f_\theta(\widetilde{Y}) - Y\|_2^2 | \widetilde{Y}] \tag{4}$$

yields a network that satisfies

$$\mathbb{E}[Y_0|\widetilde{Y}] = \frac{(1+\alpha^2) f_{\theta^*}(\widetilde{Y}) - \widetilde{Y}}{\alpha^2}. \tag{5}$$

Proof. See Section 3.3 of [26]. □

Here, Equation (4) can be thought of as Equation (3) in the limit of an infinite number of samples and $\hat{\theta}$ as a finite sample approximation of θ^*. Result 1 states that a clean image can be estimated in a conditional expectation by employing a correction term based on α. It suggests the following procedure for estimating $y_{0,s}$ upon inference: corrupt the test data y_s with further noise, $\widetilde{y}_s = y_s + \widetilde{n}_s$; apply the trained network to the further noisy data, $f_{\hat{\theta}}(\widetilde{y}_s)$; and correct the output using the right-hand side of Equation (5).

2.2. Self-Supervised Reconstruction with SSDU

This section focuses on the case where the data consist of noise-free, sub-sampled data:

$$y_s = M_{\Omega_s} y_{0,s}. \tag{6}$$

Here, M_{Ω_s} is a sampling mask, a diagonal matrix with a jth diagonal of 1 when $j \in \Omega_s$ and otherwise 0 for the sampling set $\Omega_s \subseteq \{1, 2, \ldots, q\}$.

Self-supervised reconstruction consists of training a network to recover images when only sub-sampled data is available for training: $y_t = M_{\Omega_t} y_{0,t}$ [33]. This work focuses on the popular method SSDU [13], which was theoretically justified in [25] via the multiplicative noise version of Noiser2Noise [26]. In this framework, analogous to the further noise used in Equation (2), the training data y_t are *further sub-sampled* by applying a second mask with the sampling set $\Lambda_t \subseteq \{1, 2, \ldots, q\}$ to y_t:

$$\widetilde{y}_t = M_{\Lambda_t} y_t = M_{\Lambda_t \cap \Omega_t} y_{0,t}, \tag{7}$$

where $M_{\Lambda_t \cap \Omega_t} = M_{\Lambda_t} M_{\Omega_t}$. Training consists of minimizing a loss function on indices in $\Omega_t \setminus \Lambda_t$, such as

$$\hat{\theta} = \arg\min_{\theta} \sum_{t \in \mathcal{T}} \| M_{\Omega_t \setminus \Lambda_t} (f_\theta(\widetilde{y}_t) - y_t) \|_2^2, \tag{8}$$

where $M_{\Omega_t \setminus \Lambda_t} = (\mathbb{1} - M_{\Lambda_t}) M_{\Omega_t}$. Although for theoretical ease we state SSDU with an ℓ_2 loss here, it is known that other losses are possible [13].

Let $p_j = \mathbb{P}[j \in \Omega]$ and $\widetilde{p}_j = \mathbb{P}[j \in \Lambda]$. Assuming that

$$p_j > 0 \ \ \forall\, j, \tag{9}$$

$$\widetilde{p}_j < 1 \ \ \forall\, \{j : p_j < 1\}, \tag{10}$$

the following result from [25] proves that SSDU recovers the clean image as expected.

Result 2. *Consider the random variables* $Y = M_\Omega Y_0$ *and* $\widetilde{Y} = M_\Lambda Y$. *When Equations* (9) *and* (10) *hold, minimizing*

$$\theta^* = \arg\min_{\theta} \mathbb{E}[\| M_{\Omega \setminus \Lambda} (f_\theta(\widetilde{Y}) - Y) \|_2^2 | \widetilde{Y}] \tag{11}$$

yields a network with parameters that satisfies

$$M_{(\Lambda \cap \Omega)^c} \mathbb{E}[Y_0 | \widetilde{Y}] = M_{(\Lambda \cap \Omega)^c} f_{\theta^*}(\widetilde{Y}). \tag{12}$$

Proof. See Appendix B of [25] (where [25] uses $\mathbb{1} - M_\Lambda M_\Omega$, this uses paper the more compact notation $M_{(\Lambda \cap \Omega)^c}$, where superscript c denotes the complement of a set). □

Result 2 states that the network correctly estimates Y_0 in a conditional expectation for indices not in $\Lambda \cap \Omega$. To estimate everywhere in k-space, one can overwrite sampled indices or use data-consistent architecture; see [25] for details.

3. Theory: Proposed Methods

The remainder of this paper considers the task of training a network to recover images from data that are both noisy *and* sub-sampled:

$$y_s = M_{\Omega_s}(y_{0,s} + n_s). \tag{13}$$

It has been stated that when a network reconstructs noisy MRI data with a standard training method, there is a denoising effect [20]. In the following, we are motivated by the need for methods that explicitly remove noise by showing that the apparent noise removal is in fact a "pseudo-denoising" effect due to the correct estimation of the ground truth in an expectation only for indices in Ω^c.

Consider the standard approach of training a network to map from noisy, sub-sampled y_t to noisy, fully sampled $y_{0,t} + n_t$. In terms of random variables, training consists of minimizing

$$\theta^* = \arg\min_{\theta} \mathbb{E}[\|f_\theta(Y) - (Y_0 + N)\|_2^2 | Y], \tag{14}$$

which gives a network that satisfies

$$f_{\theta^*}(Y) = \mathbb{E}[Y_0 + N|Y]. \tag{15}$$

Equation (15) does not hold for completely arbitrary network architecture. The conditions on f_θ (which are also required for Results 1 and 2) are detailed in Section II-A of [25]. In brief, the Jacobian matrix J with the entries $J_{ij} = \partial f_\theta(Y)_j / \partial \theta_i$ must have maximally linearly independent rows, which is expected for well-constructed architectures when the number of parameters exceeds q. Throughout the remainder of this paper, we assume that f_θ satisfies this condition. We also assume that the optimizer is not stuck in a poor local minimum so that the network is a good approximation of Equation (15) in practice.

It is instructive to examine how $\mathbb{E}[Y_0 + N|Y]$ depends on the sampling mask Ω. Firstly, for $j \notin \Omega$,

$$\begin{aligned} \mathbb{E}[Y_{0,j} + N_j | Y, j \notin \Omega] &= \mathbb{E}[Y_{0,j}|Y] + \mathbb{E}[N_j] \\ &= \mathbb{E}[Y_{0,j}|Y], \end{aligned} \tag{16}$$

where we use the independence of N_j from Y when $j \notin \Omega$ and $\mathbb{E}[N_j] = 0$ by assumption. For the alternative, $j \in \Omega$,

$$\mathbb{E}[Y_{0,j} + N_j | Y, j \in \Omega] = \mathbb{E}[Y_j|Y] = Y_j \tag{17}$$

where $Y_{0,j} + N_j = Y_j$ for $j \in \Omega$ is used. The trained network therefore satisfies

$$f_{\theta^*}(Y) = \mathbb{E}[Y_0 + N|Y] = M_{\Omega^c}\mathbb{E}[Y_0|Y] + M_\Omega Y. \tag{18}$$

Therefore, the network targets the noise-free Y_0 in regions in Ω^c but recovers the noisy Y otherwise. As there is less total measurement noise present than $Y_0 + N$, this gives the impression of noise removal; however, we emphasize that the network does not remove the noise in Y. Since the term "denoising" typically refers to the removal of noise *from the input data*, we use the term "pseudo-denoising" to refer to the behavior stated in Equation (18). Other than the conditions on f_θ described above, this result is agnostic to the network architecture so includes "unrolled" approaches that may have a regularization parameter which is designed to trade off the model and consistency with the data.

We refer to this method described in this section as "Supervised w/o denoising" throughout this paper. In the following, we propose methods that explicitly recover Y_0 in a conditional expectation from noisy, sub-sampled inputs in two cases: (A) the training data is noisy and fully sampled; (B) the training data is noisy and sub-sampled. For tasks A and B, we propose "Noisier2Full" and "Robust SSDU", respectively.

3.1. Noisier2Full for Fully Sampled, Noisy Training Data

This section proposes Noisier2Full, which extends the additive Noisier2Noise to reconstruction tasks for noisy, fully sampled training data. Based on Equation (2), we propose corrupting the measurements y_t with further noise on the sampled indices:

$$\widetilde{y}_t = y_t + M_{\Omega_t}\widetilde{n}_t. \tag{19}$$

Then, we minimize the loss between $\widetilde{y}_t$ and the noisy, fully sampled training data $y_{0,t} + n_t$. In terms of random variables,

$$\theta^* = \arg\min_{\theta} \mathbb{E}[\|f_\theta(\widetilde{Y}) - (Y_0 + N)\|_2^2|\widetilde{Y}]. \tag{20}$$

Minimizing the ℓ_2 norm gives a network that satisfies

$$f_{\theta^*}(\widetilde{Y}) = \mathbb{E}[Y_0 + N|\widetilde{Y}], \tag{21}$$

which is recognizable as Equation (15) with Y replaced by $\widetilde{Y}$. Similarly to Equation (16), N_j is independent of $\widetilde{Y}$ when $j \notin \Omega$, so the ground truth is estimated in such regions:

$$\mathbb{E}[Y_{0,j}|\widetilde{Y}, j \notin \Omega] = \mathbb{E}[Y_{0,j}|\widetilde{Y}]. \tag{22}$$

However, crucially, the expectation is conditional on $\widetilde{Y}$, not Y, so the additive Noisier2Noise correction stated in Result 1 is applicable when $j \in \Omega$:

$$\mathbb{E}[Y_{0,j}|\widetilde{Y}, j \in \Omega] = \frac{(1+\alpha^2)f_{\theta^*}(\widetilde{Y})_j - \widetilde{Y}_j}{\alpha^2} \tag{23}$$

Although Result 1 is not specifically constructed for sub-sampled data, it is applicable here because it is an entry-wise statistical relationship so can be applied to each index that has the proper noise statistics. Therefore, Y_0 can be estimated with

$$\mathbb{E}[Y_0|\widetilde{Y}] = M_\Omega\left(\frac{(1+\alpha^2)f_{\theta^*}(\widetilde{Y}) - \widetilde{Y}}{\alpha^2}\right) + M_{\Omega^c}f_{\theta^*}(\widetilde{Y}). \tag{24}$$

In summary, Noisier2Full recovers Y_0 in a conditional expectation by introducing further noise to the sampled indices during training and correcting those indices upon inference via additive Noisier2Noise. In the subsequent section, we show how this approach can be extended to the more challenging case where the training data are also sub-sampled.

3.2. Robust SSDU for Sub-Sampled, Noisy Training Data

This section proposes Robust SSDU, which recovers clean images in a conditional expectation when the training data are both noisy and sub-sampled. Robust SSDU combines the approaches from Sections 2.1 and 2.2 to simultaneously reconstruct and denoise data; see Figure 1 for a schematic. We propose combining Equations (2) and (7) to form a vector that is further sub-sampled *and* additionally noisy:

$$\widetilde{y}_t = M_{\Lambda_t \cap \Omega_t}(y_t + \widetilde{n}_t). \tag{25}$$

Recall that SSDU employs $M_{\Omega\setminus\Lambda}$ in the loss, which yields a network that estimates indices in $(\Lambda \cap \Omega)^c$; see Result 2. For Robust SSDU, we replace $M_{\Omega\setminus\Lambda}$ with M_Ω so that the loss is

$$\hat{\theta} = \arg\min_{\theta} \sum_{t \in \mathcal{T}} \|M_{\Omega_t}(f_\theta(\widetilde{y}_t) - y_t)\|_2^2. \tag{26}$$

In the following, we show that this change leads to estimation everywhere in k-space, not just indices in $(\Lambda \cap \Omega)^c$.

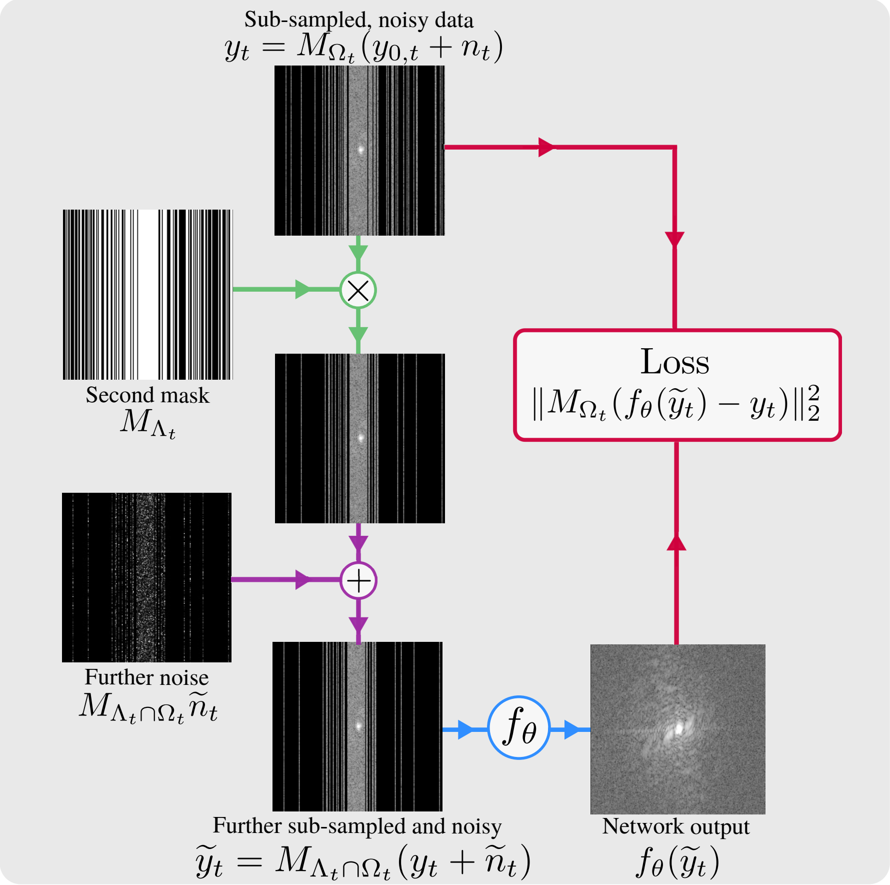
Sub-sampled, noisy data
$y_t = M_{\Omega_t}(y_{0,t} + n_t)$
Second mask
M_{Λ_t}
Loss
$\|M_{\Omega_t}(f_\theta(\widetilde{y}_t) - y_t)\|_2^2$
Further noise
$M_{\Lambda_t \cap \Omega_t}\widetilde{n}_t$
f_θ
Further sub-sampled and noisy
$\widetilde{y}_t = M_{\Lambda_t \cap \Omega_t}(y_t + \widetilde{n}_t)$
Network output
$f_\theta(\widetilde{y}_t)$

Figure 1. The proposed self-supervised reconstruction and denoising method, Robust SSDU, which extends the training procedure illustrated in Figure 1 of [25] to low-SNR data. The sub-sampled, noisy training data y_t are further sub-sampled by a mask M_{Λ_t} and corrupted by further noise $\widetilde{n}_t$, yielding $\widetilde{y}_t$. The loss is computed between y_t and $f_\theta(\widetilde{y}_t)$ on Ω_t.

Claim 1. *Consider the random variables $Y = M_\Omega(Y_0 + N)$ and $\widetilde{Y} = M_{\Lambda\cap\Omega}(Y + \widetilde{N})$, where N and $\widetilde{N}$ are zero-mean Gaussians distributed with variances of σ_n^2 and $\alpha^2\sigma_n^2$, respectively. When Equations (9) and (10) hold, minimizing*

$$\theta^* = \arg\min_\theta \mathbb{E}[\|M_\Omega(f_\theta(\widetilde{Y}) - Y)\|_2^2 | \widetilde{Y}] \tag{27}$$

yields a network with parameters that satisfies

$$f_{\theta^*}(\widetilde{Y}) = \mathbb{E}[Y_0 + N | \widetilde{Y}]. \tag{28}$$

Proof. See Appendix A. □

The differences between Equation (27) and the standard SSDU loss Equation (11) are the change from $M_{\Omega\setminus\Lambda}$ to M_Ω and the inclusion of noise in the data, Y. Intuitively, since $M_\Omega = M_{\Omega\setminus\Lambda} + M_{\Lambda\cap\Omega}$, the mask change extends Equation (11) to include entries in $M_{\Lambda\cap\Omega}$. Therefore, upon inference, the network learns to map to entries in $M_{(\Lambda\cap\Omega)^c}$, as stated in Result 2, *and* $M_{\Lambda\cap\Omega}$, which comes from the additional indices in the loss. In other words, it learns to map to everywhere in k-space. The inclusion of noise in the target simply implies that the network will learn to map to the noisy $Y_0 + N$, as in Equation (15).

Upon inference, we can use a similar approach to Section 3.1, applying the additive Noisier2Noise correction on indices sampled in $\widetilde{Y}$. Since the indices sampled in $\widetilde{Y}$ are $\Lambda \cap \Omega$, the clean image Y_0 is estimable with

$$\mathbb{E}[Y_0|\widetilde{Y}] = M_{\Lambda\cap\Omega}\left(\frac{(1+\alpha^2)f_{\theta^*}(\widetilde{Y})-\widetilde{Y}}{\alpha^2}\right)+M_{(\Lambda\cap\Omega)^c}f_{\theta^*}(\widetilde{Y}). \tag{29}$$

Roughly speaking, Robust SSDU can be thought of as a generalization of Noisier2Full to sub-sampled training data. Specifically, Robust SSDU is mathematically equivalent to Noisier2Full when $\Omega = \{1, 2, \ldots, q\}$ and there is a change in notation of $\Lambda \to \Omega$. More broadly, Robust SSDU can be interpreted as the simultaneous application of additive and multiplicative Noisier2Noise [25,26].

3.3. Loss Weighting of Noisier2Full and Robust SSDU

For Noisier2Full and Robust SSDU, the task during training and inference is not identical; during training, the network maps from $\widetilde{Y}$ to $Y_0 + N$ or $M_\Omega(Y_0 + N)$, while upon inference, it maps from $\widetilde{Y}$ to Y_0 via the α-based correction term. Taking a similar approach to [34,35], this section describes how this can be compensated for by modifying the loss function in such a way that its gradient equals the gradient of the target loss in a conditional expectation.

Claim 2. *Consider the random variables $Y = M_\Omega(Y_0 + N)$ and $\widetilde{Y} = Y + M_\Omega\widetilde{N}$, where N and $\widetilde{N}$ are zero-mean Gaussians distributed with variances of σ_n^2 and $\alpha^2\sigma_n^2$, respectively. We define*

$$\hat{Y}_{Nr2F} = M_\Omega\left(\frac{(1+\alpha^2)f_\theta(\widetilde{Y})-\widetilde{Y}}{\alpha^2}\right)+M_{\Omega^c}f_\theta(\widetilde{Y}) \tag{30}$$

where f_θ is an arbitrary function. Then,

$$\nabla_\theta\mathbb{E}\left[\|\hat{Y}_{Nr2F}-Y_0\|_2^2|\widetilde{Y}\right] = \nabla_\theta\mathbb{E}\left[\left\|W_\Omega(f_\theta(\widetilde{Y})-Y_0-N)\right\|_2^2|\widetilde{Y}\right]. \tag{31}$$

where

$$W_\Omega = \frac{1+\alpha^2}{\alpha^2}M_\Omega + M_{\Omega^c}. \tag{32}$$

Proof. See Appendix B. □

We therefore suggest replacing the Noisier2Full loss stated in Equation (20) with the right-hand side of Equation (31), which increases the weight of the indices in Ω. Intuitively, it uses the ratio of noise removed during training, which has the variance $\mathrm{Var}(\widetilde{N}) = \alpha^2\sigma_n^2$, and the noise removed upon inference, which has the variance $\mathrm{Var}(N+\widetilde{N}) = (1+\alpha^2)\sigma_n^2$, to compensate for the difference between the task during training and inference. The following result concerns the analogous expression for Robust SSDU.

Claim 3. *Consider the random variables $Y = M_\Omega(Y_0 + N)$ and $\widetilde{Y} = M_{\Lambda\cap\Omega}(Y + \widetilde{N})$, where N and $\widetilde{N}$ are zero-mean Gaussians distributed with variances of σ_n^2 and $\alpha^2\sigma_n^2$, respectively. We define*

$$\hat{Y}_{RSSDU} = M_{\Lambda\cap\Omega}\left(\frac{(1+\alpha^2)f_\theta(\widetilde{Y})-\widetilde{Y}}{\alpha^2}\right)+M_{(\Lambda\cap\Omega)^c}f_{\theta^*}(\widetilde{Y}) \tag{33}$$

where f_θ is an arbitrary function. Then,

$$\mathbb{E}\left[\|\hat{Y}_{RSSDU}-Y_0\|_2^2|\widetilde{Y}\right] = \nabla_\theta\mathbb{E}\left[\left\|W_{\Omega,\Lambda}M_\Omega(f_\theta(\widetilde{Y})-Y)\right\|_2^2|\widetilde{Y}\right] \tag{34}$$

where

$$W_{\Omega,\Lambda} = \frac{1+\alpha^2}{\alpha^2} M_{\Lambda\cap\Omega} + \mathcal{P}^{\frac{1}{2}} M_{\Omega\setminus\Lambda} \tag{35}$$

and $\mathcal{P} = \mathbb{E}[M_{\Omega\setminus\Lambda}]^{-1}\mathbb{E}[M_{(\Lambda\cap\Omega)^c}]$.

Proof. See Appendix C. □

The $M_{\Lambda\cap\Omega}$ coefficient has a similar role to the M_Ω coefficient in Equation (31). The $M_{\Omega\setminus\Lambda}$ coefficient compensates for the variable density of Ω and Λ and was first proposed in [25], where it was shown to improve the reconstruction quality and robustness of the distribution of Λ for standard SSDU without denoising (where [25] uses $(\mathbb{1} - K)^{-1}$, this paper uses the more compact $\mathcal{P}$).

The weightings can be thought of as entry-wise modifications of the learning rate [25]. Neither weighting matrices change θ^*, so the proofs of Noisier2Full and Robust SSDU from Sections 3.1 and 3.2 hold. Rather, the role of the weights is to improve the finite-sample case in practice, where θ^* is estimated with $\hat{\theta}$; see Section 5 for an empirical evaluation. Throughout the remainder of this paper, "Noisier2Full" and "Robust SSDU" refer to the versions with the loss weightings proposed in this section and versions without such weightings are explicitly referred to as "Unweighted Noisier2Full" and "Unweighted Robust SSDU".

4. Materials and Methods

4.1. Description of Data

We primarily used the multi-coil brain data from the publicly available fastMRI dataset [36] (available from https://fastmri.med.nyu.edu, accessed on 1 October 2021). We only used data that had 16 coils so that the training, validation, and test sets contained 2004, 320, and 224 slices, respectively. The slices were normalized so that the cropped RSS estimate had a maximum of 1. Here, the cropped RSS was defined as $Z((\sum_c^{N_c} |F^H y_c|^2)^{\frac{1}{2}})$, where the subscript c refers to all entries on the cth coil, F^H is the conjugate transpose of the discrete Fourier transform, N_c is the number of coils, and Z is an operator that crops to a central 320×320 region. RSS images were used for normalization and visualization only; otherwise, the raw, complex, multi-coil, k-space data were used. We retrospectively sub-sampled column-wise with the central 10 lines fully sampled and and the others randomly drawn with polynomial density, with the probability density scaled to achieve a desired acceleration factor, $R_\Omega = q/\sum_j p_j$. For $R_\Omega = 4$ and $\sigma_n = 0.04$, we also trained the methods on 2D Bernoulli sampling, where the sampling was random and independent, and also with polynomial variable density. For each case, the distribution of M_Λ was the same type as the first [25]. The data were treated as noise-free, and we generated white, complex Gaussian measurement noise with the standard deviation σ_n to simulate noisy conditions.

We also tested the methods' performance on the 0.3T dataset M4Raw [37]. For this dataset, which prospectively had a low SNR, no further noise was added. Rather, the noise covariance matrix was estimated using the fully sampled image via a 30×30 square of background from each corner and the data were whitened by left-multiplying with the inverse covariance matrix so that all data had a noise standard deviation of 1. The same column-wise sub-sampling was used as described above for the fastMRI data. Although it was more realistic for the simulated noise setting of fastMRI, for M4Raw we had no "ground truth", so it was only possible to evaluate the methods' performance qualitatively.

An implementation of our method in PyTorch is available on GitHub (https://github.com/charlesmillard/robust_ssdu, accessed on 7 December 2023).

4.2. Comment on Proposed Methods in Practice

The theoretical guarantees for Noisier2Full and Robust SSDU use the further noisy, possibly further sub-sampled $\widetilde{y}_s$ as the input to the network upon inference. In practice, as suggested in the original Noisier2Noise [26] and SSDU [13] papers, we used y_s as the input to the network upon inference, so that the estimate

$$\hat{y}_s = M_{\Omega_s}\left(\frac{(1+\alpha^2)f_{\hat{\theta}}(y_s) - y_s}{\alpha^2}\right) + M_{\Omega_s^c} f_{\hat{\theta}}(y_s) \tag{36}$$

was used in place of Equations (24) and (29). Although this deviates from strict theory, and is not guaranteed to be correct in a conditional expectation, we found that it achieves better reconstruction performance in practice; see [25,26] for a detailed empirical evaluation. All subsequent results for the proposed methods use this estimate upon inference.

4.3. Comparative Training Methods

The training methods evaluated in this paper are summarized in Table 1.

For the noise-free, fully sampled training data, fully supervised training could be employed, where the loss was computed between the output of the network $f_\theta(y_t)$ and the noise-free, fully sampled target $y_{0,t}$; see Table 1. Although it was possible in principle to have higher-SNR data during training than upon inference by acquiring multiple averages [37], such datasets would require an extended acquisition time and are rare in practice. Nonetheless, training a network on this type of data via simulation is instructive as a best-case target. This method is referred to as the "fully supervised benchmark" throughout this paper.

For the noisy, fully sampled training data, we employed three training methods: Unweighted Noisier2Full, Noisier2Full and the standard approach Supervised w/o denoising, as described in Section 3. We did not compare them to Noise2Inverse [38] as it was designed for learned, denoising but fixed reconstruction operators.

For the more challenging scenario where noisy, sub-sampled training data were available, we compared Robust SSDU to the original version of SSDU, which reconstructs sub-sampled data but does not denoise. We refer to this as "Standard SSDU". To our knowledge, the only existing training method that explicitly aims to simultaneously remove noise and reconstruct incoherently sampled data in a fully self-supervised manner is Noise2Recon-SS [27], which, like Robust SSDU, includes adding further noise to the sub-sampled data. However, Noise2Recon-SS has a number of key differences to the method proposed in this paper. With an ℓ_2 k-space loss, training with Noise2Recon-SS consists of minimizing

$$\hat{\theta} = \arg\min_{\theta} \sum_{t\in\mathcal{T}} \|M_{\Omega_t\setminus\Lambda_t}(f_\theta(M_{\Lambda_t}y_t) - y_t)\|_2^2 + \lambda\|f_\theta(y_t + M_{\Omega_t}\widetilde{n}_t) - f_\theta(M_{\Lambda_t}y_t)\|_2^2, \tag{37}$$

where λ is a hand-selected weighting. We used $\lambda = 1$ throughout, which we found performed reasonably well across the range of sampling patterns, noise levels, and datasets explored. We note that it may be possible in principle to improve Noise2Recon-SS's performance by tuning λ to specific datasets, reconstruction patterns, and noise levels. However, since no ground truth is available, such tuning cannot be performed quantitatively, so fixing λ for all experiments is a reasonable reflection of the method's performance in practice. The ℓ_2 loss in k-space was used so that it could be fairly compared to the other methods in this paper, but we note that [27] used image-domain losses. The first term was based on SSDU, and the second ensured that $f_\theta(y_t + M_{\Omega_t}\widetilde{n}_t)$ and $f_\theta(M_{\Lambda_t}y_t)$ yielded similar outputs so that the method was in a sense robust to $\widetilde{n}_t$. Upon inference, Noise2Recon-SS uses $\hat{y}_s = f_{\hat{\theta}}(y_s)$; there is no correction term. We emphasize that, unlike the proposed Robust SSDU, there is no theoretical evidence that Noise2Recon-SS recovers a clean image as expected.

In [20], an untrained denoising algorithm was appended to a reconstruction network. To test this, we denoised the RSS output of Supervised w/o denoising and Standard SSDU with the popular BM3D algorithm [39], which is designed for Gaussian noise. Although the measurement noise was Gaussian, the reconstruction error of the RSS image was not

Gaussian in general [40]. Therefore, unlike the proposed methods, BM3D did not accurately model the noise characteristics [41]. Nonetheless, we found that these methods performed reasonably well in practice.

Table 1. The training methods evaluated in this paper, where $y_t = M_{\Omega_t}(y_{0,t} + n_t)$ and the asterisk denotes the proposed methods. Here, and throughout this paper, the subscripts t and s index the training and test sets, respectively. The function BM3D$(\cdot)$ is defined here to include an RSS transform so that the denoiser acts on the RSS image. The double lines are used to separate types of data available for training. The unweighted variants of Noisier2Full and Robust SSDU, which are not stated here for brevity, are equivalent to the weighted versions with $W_{\Omega_t} = \mathbb{1}$ and $W_{\Omega_t,\Lambda_t} = \mathbb{1}$.

Name	Training Data	Loss	Estimate upon Inference
Fully supervised benchmark	$y_{0,t}$	$\sum_{t\in\mathcal{T}} \lVert f_\theta(y_t) - y_{0,t}\rVert_2^2$	$f_\theta(y_s)$
Supervised w/o denoising	$y_{0,t} + n_t$	$\sum_{t\in\mathcal{T}} \lVert f_\theta(y_t) - (y_{0,t} + n_t)\rVert_2^2$	$f_{\hat{\theta}}(y_s)$
Supervised with BM3D denoising	$y_{0,t} + n_t$	$\sum_{t\in\mathcal{T}} \lVert f_\theta(y_t) - (y_{0,t} + n_t)\rVert_2^2$	BM3D$(f_{\hat{\theta}}(y_s))$
Noisier2Full *	$y_{0,t} + n_t$	$\sum_{t\in\mathcal{T}} \lVert W_{\Omega_t}(f_\theta(y_t + M_{\Omega_t}\widetilde{n}_t) - (y_{0,t} + n_t))\rVert_2^2$	$M_{\Omega_s}\left(\frac{(1+\alpha^2)f_{\hat{\theta}}(y_s)-y_s}{\alpha^2}\right) + M_{\Omega_s^c} f_{\hat{\theta}}(y_s)$
Standard SSDU	y_t	$\sum_{t\in\mathcal{T}} \lVert M_{\Omega_t\setminus\Lambda_t}(f_\theta(M_{\Lambda_t}y_t) - y_t)\rVert_2^2$	$f_{\hat{\theta}}(y_s)$
SSDU with BM3D	y_t	$\sum_{t\in\mathcal{T}} \lVert M_{\Omega_t\setminus\Lambda_t}(f_\theta(M_{\Lambda_t}y_t) - y_t)\rVert_2^2$	BM3D$(f_{\hat{\theta}}(y_s))$
Noise2Recon-SS	y_t	$\sum_{t\in\mathcal{T}} \lVert M_{\Omega_t\setminus\Lambda_t}(f_\theta(M_{\Lambda_t}y_t) - y_t)\rVert_2^2 + \lambda \lVert f_\theta(y_t + M_{\Omega_t}\widetilde{n}_t) - f_\theta(M_{\Lambda_t}y_t)\rVert_2^2$	$f_{\hat{\theta}}(y_s)$
Robust SSDU *	y_t	$\sum_{t\in\mathcal{T}} \lVert W_{\Omega_t,\Lambda_t} M_{\Omega_t}(f_\theta(M_{\Lambda_t\cap\Omega_t}(y_t + \widetilde{n}_t)) - y_t)\rVert_2^2$	$M_{\Omega_s}\left(\frac{(1+\alpha^2)f_{\hat{\theta}}(y_s)-y_s}{\alpha^2}\right) + M_{\Omega_s^c} f_{\hat{\theta}}(y_s)$

4.4. Network Architecture

For all methods considered in this paper, the function f_θ is defined to be k-space to k-space, but it is otherwise agnostic to the network architecture. Architectures can include inverse Fourier transforms, so convolutional layers may be applied in the image domain. We emphasize that the experiments in this paper are designed to compare the performance of the training *method*, not to provide a comprehensive evaluation of possible architectures, which is a somewhat orthogonal goal.

We employed a network architecture based on the Variational Network (VarNet) [7,42], which is available as part of the fastMRI package [36]. VarNet consists of a coil sensitivity map estimation module followed by a series of "cascades". The k-space estimate at the kth cascade takes the form

$$\hat{y}_{k+1} = \hat{y}_k - \eta_k M_{in}(\hat{y}_k - y_{in}) + G_{\theta_k}(\hat{y}_k) \tag{38}$$

where y_{in} and M_{in} are the input k-space and sampling mask, respectively, and the t or s index is dropped for legibility. We use the generic subscript in here because the input is not the same for every method; for instance, the fully supervised training and Noisier2Full have $M_{in} = M_{\Omega_t}$ and $M_{in} = M_{\Lambda_t\cap\Omega_t}$, respectively. Here, η_k is a trainable parameter and $G_{\theta_k}(\hat{y}_k)$ is a neural network with cascade-dependent parameters, θ_k, referred to as a "refinement module", which was an image-domain U-net [43] with real weights in [7,42].

VarNet was originally constructed for reconstruction only, without explicit denoising. For joint reconstruction and denoising, we propose partitioning $G_{\theta_k}(\hat{y}_k)$ into two functions:

$$G_{\theta_k}(\hat{y}_k) = M_{in}G_{\theta_k^D}(\hat{y}_k) + (\mathbb{1} - M_{in})G_{\theta_k^R}(\hat{y}_k). \tag{39}$$

This refinement module is illustrated in Figure 2. We refer to the architecture with the proposed refinement module as "Denoising VarNet" throughout this paper. We used a U-net [43] for both $G_{\theta_k^D}(\hat{y}_k)$ and $G_{\theta_k^R}(\hat{y}_k)$, although we note that, in general, these functions need not be the same. We used 5 cascades, giving a network with 2.5×10^7 parameters.

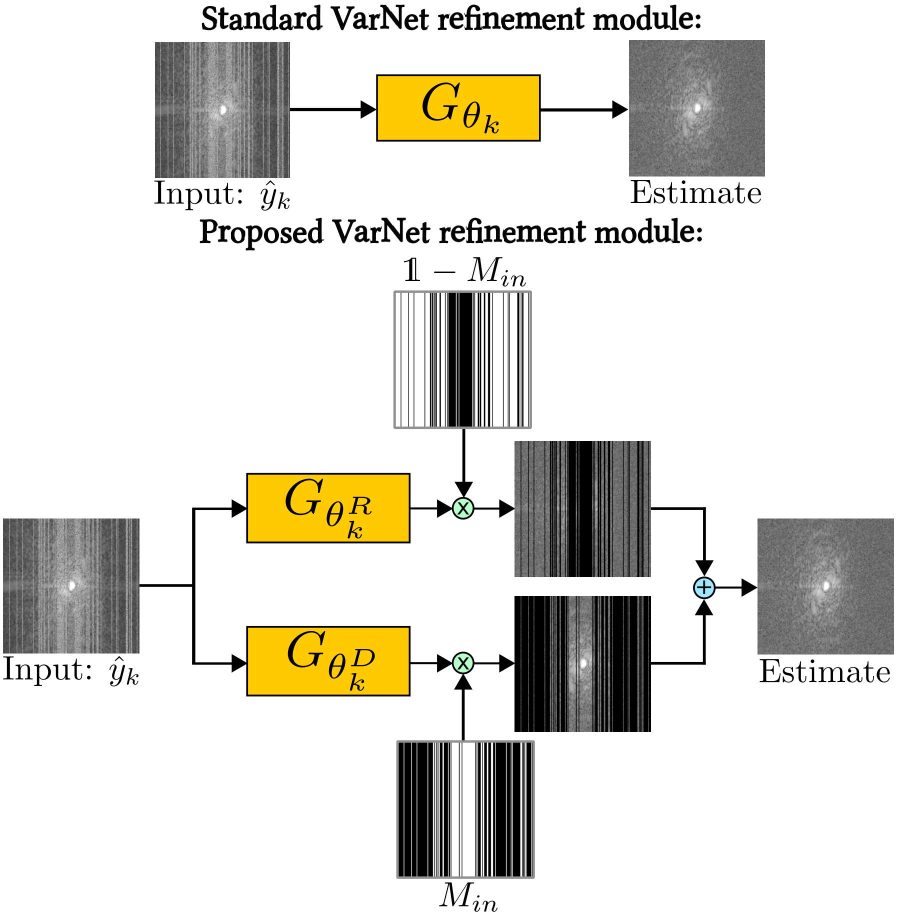

Figure 2. The refinement module for the proposed architecture Denoising VarNet, which trains two networks in parallel, removing noise and aliasing separately.

4.5. Training Details

We used the Adam optimizer [44] and trained for 100 epochs with a learning rate of 10^{-3}. The Ω_t and n_t were fixed but the Λ_t and $\widetilde{n}_t$ were re-generated once per epoch [45], which we found considerably reduced susceptibility to overfitting. As in [25], we used the same distribution of Λ_t as Ω_t but with parameters selected to give a sub-sampling factor of $R_\Lambda = q / \sum_j \widetilde{p}_j = 2$ unless otherwise stated. The choice of α is discussed in Section 5.2. Unless otherwise stated, the training methods were evaluated on data generated with $\sigma_n \in \{0.02, 0.04, 0.06, 0.08\}$ and $R_\Omega \in \{4, 8\}$. We note that the noise's standard deviation, not the SNR, was fixed and that for each training method, σ_n and R_Ω, we trained a separate network from scratch.

4.6. Performance Metrics

Since each of the methods were trained using a squared error loss in k-space, we primarily focused on the k-space normalized mean squared error (NMSE) over the test set, defined as $\frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \|\hat{y}_s - y_{0,s}\|_2^2 / \|y_{0,s}\|_2^2$ where $\hat{y}_s$ is an estimate of k-space. Since the score was in k-space, it was not possible to compute the NMSE of methods that employed BM3D, which acted on the magnitude image so did not retain the complex phase. The peak signal-to-noise ratio (PSNR) was also computed but was found to display very similar trends to the NMSE so is not shown for brevity.

We also computed the mean structural similarity (SSIM) [46] on the RSS images. We emphasize that the networks were *not* trained to minimize the SSIM directly, so such scores are somewhat incidental to the primary NMSE results and not necessarily fundamental to the method.

5. Results

5.1. Evaluation of Denoising VarNet

To evaluate the performance of the proposed Denoising VarNet architecture, we trained the best-case baseline for Standard VarNet with ten cascades and that for Denoising VarNet with five cascades so that they had roughly the same number of parameters. Figure 3 shows that Denoising VarNet outperformed Standard VarNet on the test set for all considered R_Ω and σ_n, especially for more challenging acceleration factors and noise levels.

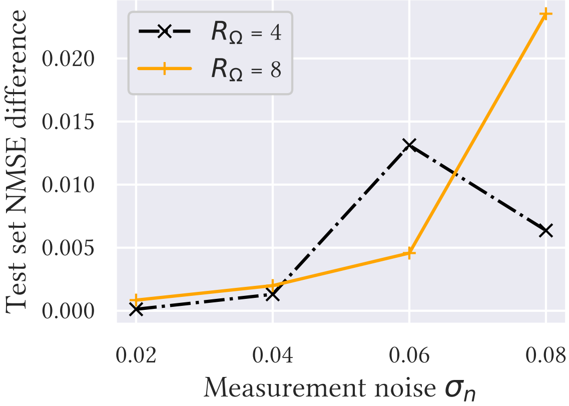

Figure 3. The difference between the test set loss of Standard VarNet and the proposed Denoising VarNet for the benchmark training method. All differences are positive, showing that Denoising VarNet outperformed Standard VarNet, especially for a large σ_n.

5.2. Robustness to α

To evaluate the robustness to α, we trained Noisier2Full, Robust SSDU, and their weighted variants for $\alpha \in \{0.05, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75\}$. We focused solely on the case where $R_\Omega = 8$ and $\sigma_n = 0.06$. The performance with the test set is shown in Figure 4, which shows that the weighted versions were considerably more robust. The weighted and unweighted minima were at $\alpha = 1$ and 1.25 for Noisier2Full and $\alpha = 0.75$ and 0.5 for Robust SSDU, respectively. We employed these values of α for all experiments in Sections 5.3 and 5.4; we assumed that the tuned α at $R_\Omega = 8$ and $\sigma_n = 0.06$ was a reasonable approximation of the optimum for every evaluated R_Ω and σ_n.

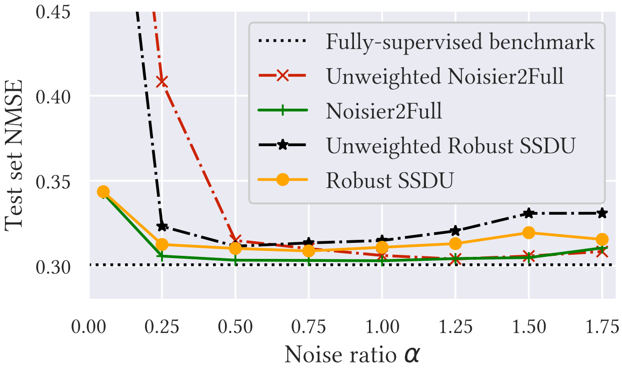

Figure 4. The robustness to α of Noisier2Full, Robust SSDU, and their weighted versions at $R_\Omega = 8$ and $\sigma_n = 0.06$. The performance of the fully supervised benchmark, which did not depend on α, is also shown. The weighted versions were substantially more robust, especially for small α: at $\alpha = 0.05$, the values of the Unweighted Noisier2Full and Robust SSDU, which are excluded from the visualization, were 0.70 and 0.62, respectively.

5.3. Task A: Fully Sampled, Noisy Training Data

Rows 3–5 of Table 2 show how the test set's NMSE of networks trained on fully sampled, noisy data compares to the fully supervised benchmark. Supervised w/o denoising's performance significantly degraded as σ_n increased; for $R_\Omega = 8$ and $\sigma_n = 0.08$, Supervised w/o denoising's test set loss was approximately double that of the fully supervised benchmark. In contrast, Noisier2Full consistently performed similarly to the benchmark; its NMSE was within 0.008 for all σ_n and R_Ω. The performance of Unweighted Noisier2Full was slightly poorer than the weighted version, especially for high noise levels for the more challenging acceleration factor of $R_\Omega = 8$. Two reconstruction examples are shown in Figure 5. Here, and throughout this paper, the example reconstructions show the image-domain RSS cropped to a central 320 × 320 region. The k-space's NMSE and SSIM are also shown. Appendix D shows the mean SSIM for the test set for all methods.

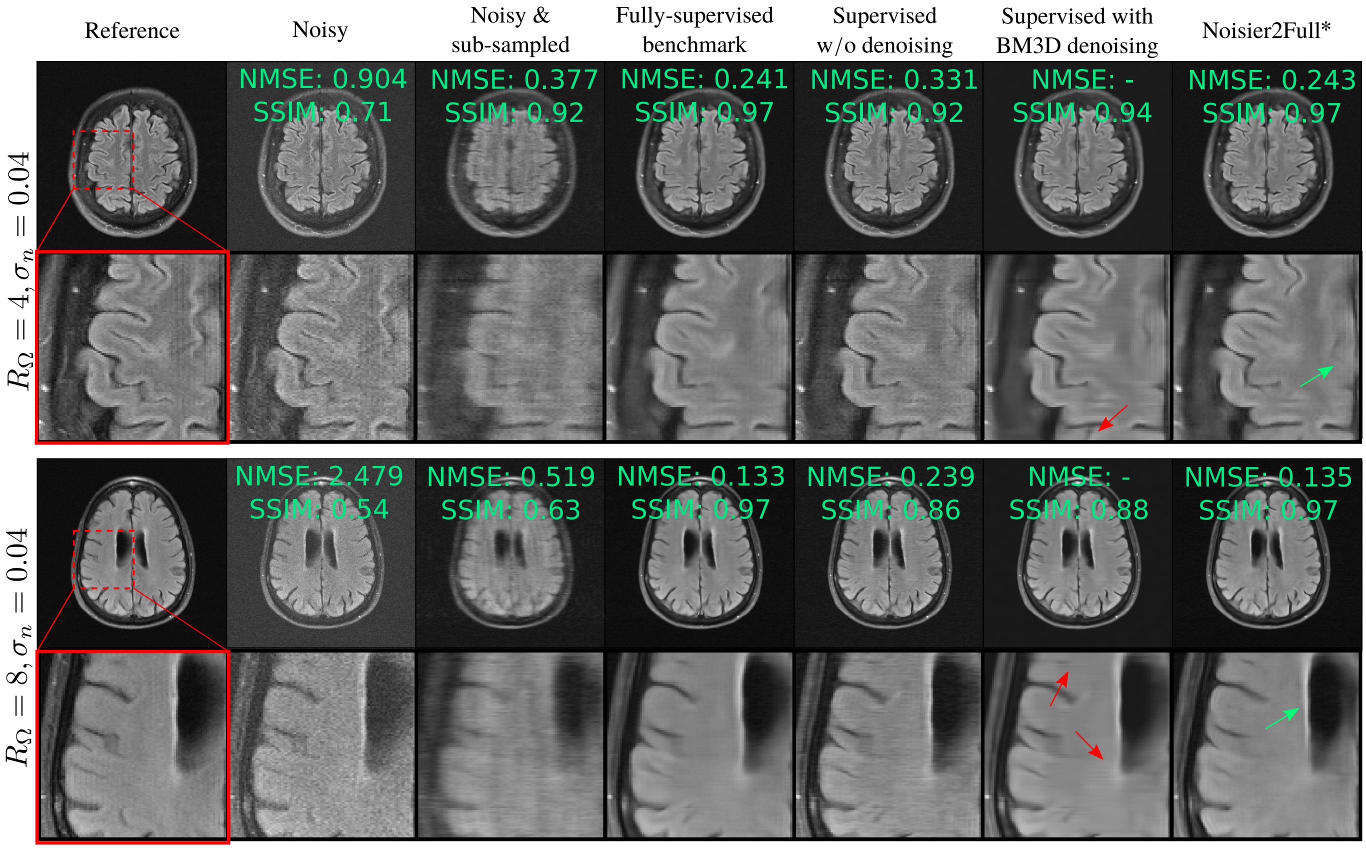

Figure 5. Reconstructions when fully sampled, noisy data are available for training. "Noisy" and "Noisy and sub-sampled" refer to the RSS reconstruction of $y_{0,s} + n_s$ and $M_{\Omega_s}(y_{0,s} + n_s)$, respectively. While there is clear noise in Supervised w/o denoising's reconstruction, the proposed method, which is indicated with an asterisk, performs very similarly to the fully supervised benchmark. The red arrows show artifacts for Supervised with BM3D, and the green arrows show the improved recovery and contrast of fine features for Noisier2Full.

Table 2. The methods' test set's NMSE with the fastMRI multi-coil brain dataset with standard errors. The double lines separate the type of training data available and bold font is used to denote the best performance within each category. Methods that used BM3D could not be included because the NMSE was computed in k-space and BM3D acted on the magnitude image, so the complex phase was not retained. Table A1 shows a similar table for the SSIM. Asterisks denote proposed methods.

	Acceleration Factor $R_\Omega = 4$				**Acceleration Factor $R_\Omega = 8$**			
	$\sigma_n = \mathbf{0.02}$	$\sigma_n = \mathbf{0.04}$	$\sigma_n = \mathbf{0.06}$	$\sigma_n = \mathbf{0.08}$	$\sigma_n = \mathbf{0.02}$	$\sigma_n = \mathbf{0.04}$	$\sigma_n = \mathbf{0.06}$	$\sigma_n = \mathbf{0.08}$
Noisy and sub-sampled	0.210 ± 0.01	0.434 ± 0.01	0.809 ± 0.02	1.333 ± 0.02	0.207 ± 0.02	0.337 ± 0.02	0.554 ± 0.02	0.857 ± 0.02
Fully supervised benchmark	**0.167 ± 0.02**	**0.313 ± 0.02**	**0.537 ± 0.02**	**0.850 ± 0.02**	**0.160 ± 0.02**	**0.217 ± 0.02**	**0.301 ± 0.02**	**0.414 ± 0.02**
Supervised w/o denoising	0.187 ± 0.02	0.412 ± 0.02	0.788 ± 0.02	1.314 ± 0.02	0.178 ± 0.02	0.310 ± 0.02	0.527 ± 0.02	0.833 ± 0.02
Unweighted Noisier2Full *	0.170 ± 0.02	0.319 ± 0.02	0.548 ± 0.02	0.870 ± 0.02	0.164 ± 0.02	0.223 ± 0.02	0.315 ± 0.02	0.441 ± 0.02
Noisier2Full *	**0.169 ± 0.02**	**0.312 ± 0.02**	**0.538 ± 0.02**	**0.853 ± 0.02**	**0.162 ± 0.02**	**0.220 ± 0.02**	**0.305 ± 0.02**	**0.422 ± 0.02**
Standard SSDU	0.188 ± 0.01	0.413 ± 0.01	0.787 ± 0.01	1.310 ± 0.01	0.180 ± 0.01	0.312 ± 0.01	0.531 ± 0.01	0.838 ± 0.01
Noise2Recon-SS	0.180 ± 0.02	0.377 ± 0.02	0.623 ± 0.02	0.975 ± 0.02	0.173 ± 0.02	0.260 ± 0.02	0.452 ± 0.02	0.691 ± 0.02
Unweighted Robust SSDU *	0.170 ± 0.02	**0.314 ± 0.02**	0.548 ± 0.02	0.863 ± 0.02	**0.162 ± 0.02**	**0.222 ± 0.02**	**0.309 ± 0.02**	0.424 ± 0.02
Robust SSDU *	**0.169 ± 0.02**	0.315 ± 0.02	**0.543 ± 0.02**	**0.862 ± 0.02**	**0.162 ± 0.02**	0.224 ± 0.02	**0.309 ± 0.02**	**0.423 ± 0.02**

5.4. Task B: Sub-Sampled, Noisy Training Data

Rows 6–9 of Table 2 show the test set's loss for the methods designed for sub-sampled, noisy training data. Robust SSDU performed within 0.012 of the fully supervised benchmark, despite only having access to noisy, sub-sampled training data. Noise2Recon-SS performed well in some cases, particularly at $R_\Omega = 4$, but was consistently outperformed by both variants of Robust SSDU. To determine whether the observed NMSE improvements in Robust SSDU were statistically significant, a one-sided Wilcoxon signed-rank test was performed with a p-value of 0.01. It was found that the NMSE differences in both versions of Robust SSDU compared to Standard SSDU and Noise2Recon-SS were indeed statistically significant for both acceleration factors and all noise levels. Differences between Unweighted Robust SSDU and Robust SSDU were not significant except for the case where $R_\Omega = 4$ and $\sigma_n = 0.06$.

Figure 6 shows example reconstructions, qualitatively demonstrating similar performance to the fully supervised benchmark. Figure 7 compares Standard SSDU and Robust SSDU using clinical expert bounding boxes from fastMRI+ [47], which shows that the proposed method had substantially enhanced pathology visualization. For the 2D Bernoulli sampling, we found that a lower R_Λ and α achieved better performance in practice; we used $R_\Lambda = 1.5$ and $\alpha = 0.5$. Figure 8 compares Standard SSDU and Robust SSDU for the 2D Bernoulli sampled data at $R_\Omega = 4$ and $\sigma_n = 0.04$, showing that the denoising effect was not specific to column-wise sampling.

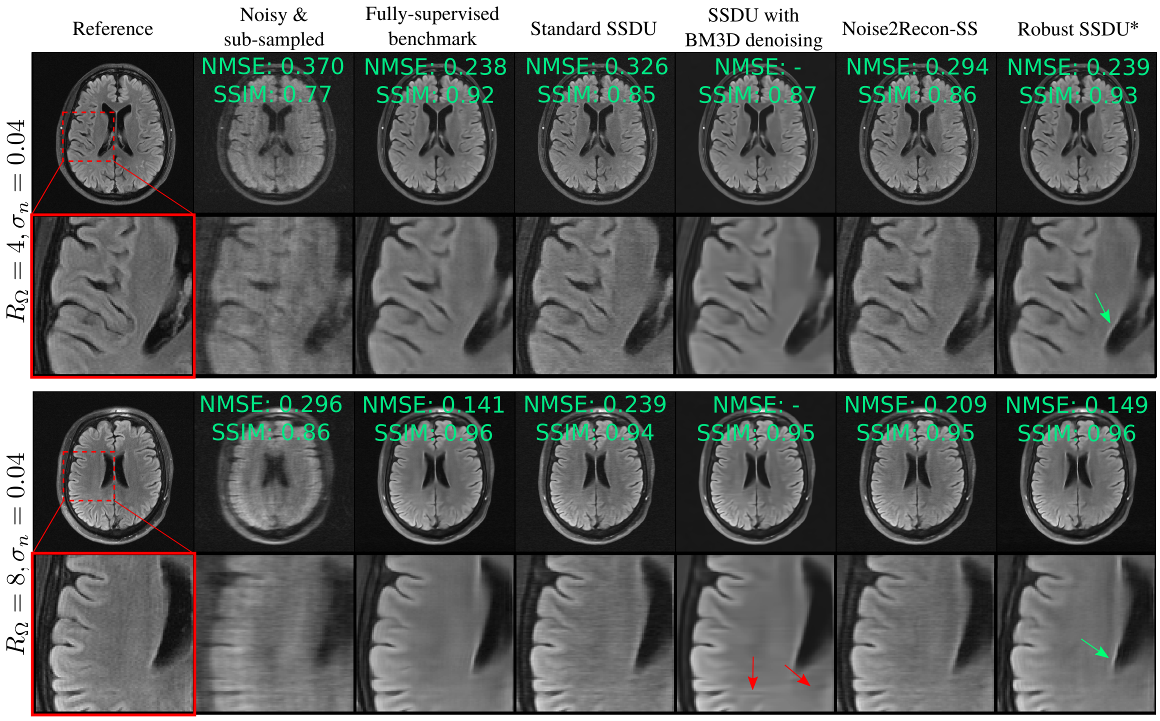

Figure 6. Example reconstructions for networks trained on noisy, sub-sampled data. The proposed method, Robust SSDU, highlighted with an asterisk, performed very similarly to the fully supervised benchmark, even at $R_\Omega = 8$. Red arrows highlight hallucinated features in the SSDU with BM3D image, whereas green arrows highlight good recovery of edge features in the Robust SSDU reconstructions.

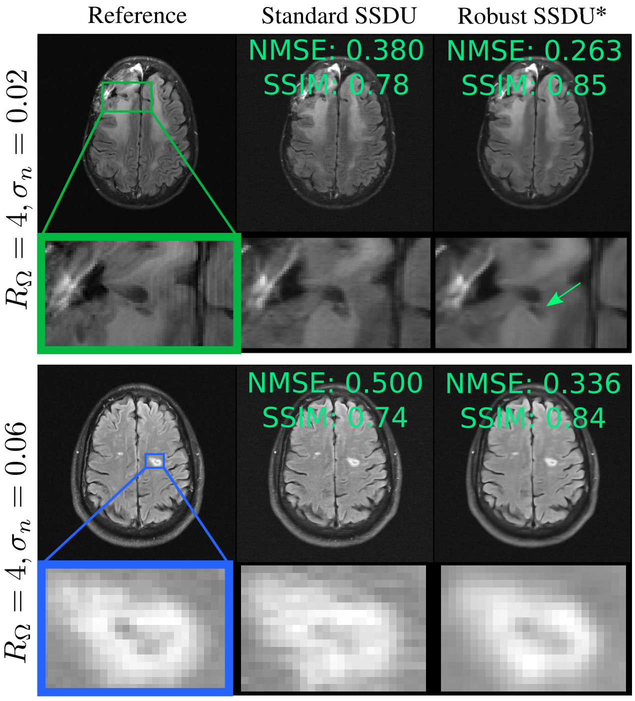

Figure 7. Clinical regions of interest annotated via fastMRI+ [47]. The top image shows a resection cavity and the bottom shows a lacunar infarct. The proposed method, Robust SSDU, highlighted with an asterisk, has improved sharpness compared to Standard SSDU, which has reconstruction errors arising from measurement noise. The arrow highlights improved recovery of infarct geometry.

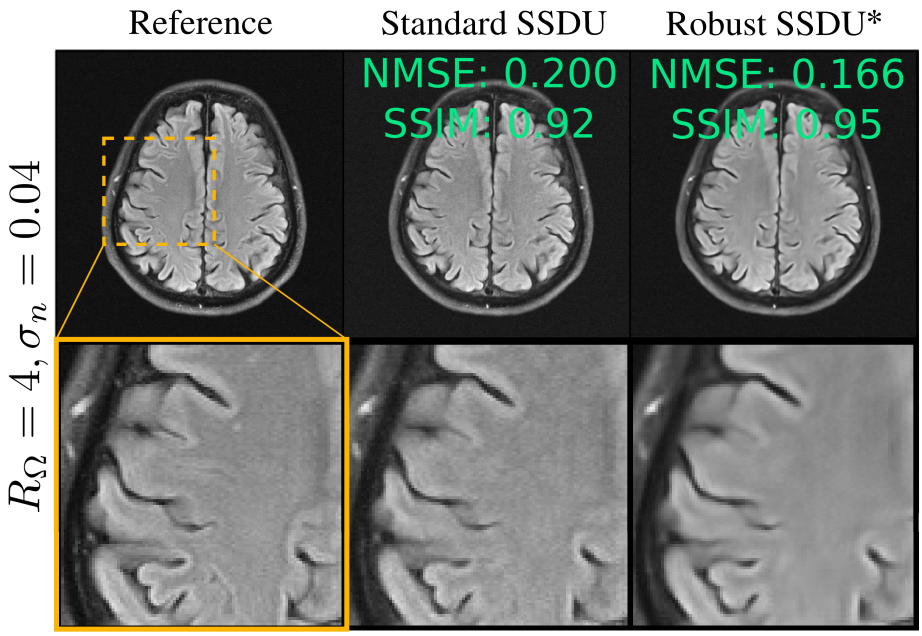

Figure 8. Example reconstruction for 2D Bernoulli sampling. For Standard SSDU, the test set's NMSE and SSIM were 0.383 and 0.72, respectively, and for Robust SSDU, highlighted with an asterisk, the test set's NMSE and SSIM were 0.316 and 0.75, respectively.

6. Discussion

Figure 3 shows that the proposed Denoising VarNet consistently outperformed the Standard VarNet architecture. We understand this to be a consequence of the difference between the distributions of errors due to sub-sampling or measurement noise; Standard VarNet removed both contributions to the error in a single U-net per cascade, while Denoising VarNet simplified the task by decomposing the contributions to the error so that each of the two U-nets per cascade were specialized for the two distinct error distributions.

The improvement in robustness for the weighted versions, shown in Figure 4, was especially prominent for a small α. For instance, at $\alpha = 0.05$, the unweighted variant of Noisier2Full was 0.50 from the benchmark, while the weighted variant was only 0.04 away. For a large α, the α-based weighting was closer to 1, so the weighted Noisier2Full tended to the unweighted method and the difference in performance was small. For instance, when $\alpha = 1.75$, the α-based weighting was 1.33, so it had a relatively marginal effect. Although the performances of the methods were reasonably similar for a tuned α, we recommend using the weighted version in practice due to its improved robustness to α. We emphasize that α tuning was only possible here because the noise and sub-sampling were simulated retrospectively; if the data were prospectively noisy and sub-sampled, it would not possible to evaluate the fidelity of the estimate and the ground truth. Robustness to hyperparameters such as α is therefore of great importance for the method's usefulness in practice.

The examples in Figures 5–7 show that proposed methods are qualitatively very similar to the fully supervised benchmark and substantially improve over methods without denoising, whose reconstructions are visibly corrupted with measurement noise. The examples exhibited some loss of detail and blurring at tissue boundaries, especially at $R_\Omega = 8$. However, the extent of detail loss was similar in the benchmark, indicating that the loss of detail was not a limitation of the proposed methods. Rather, the qualitative performance was limited by other factors such as the architecture, dataset, and choice of loss function. This can also be explained in part by noting that the high-frequency regions of k-space, which provide fine details, typically have a smaller signal so are particularly challenging to recover in the presence of significant measurement noise.

Table 2 shows that the NMSE of the noisy, sub-sampled input to the network was *lower* for the higher acceleration factor. This counter-intuitive incidental finding can be understood by noting that the spectral density was typically highly concentrated towards the center, so much of the k-space had a small magnitude. Therefore, even for moderate noise levels, zero may have been closer to the ground truth than the noisy data, so masking out such regions may have improved the NMSE. This was also reflected in the NMSE scores of the reconstructed images. However, we note that this effect was not generally reflected in the qualitative performance of the methods, which we found more frequently exhibited oversmoothing and artifacts for higher acceleration. We believe this to be because the masked data were biased, so it was more difficult to achieve a high-quality qualitative performance in practice.

The pseudo-denoising effect described in Section 3 is visible in Figure 5, showing less noise in Supervised w/o denoising than Noisy. Table 2 shows that Standard SSDU performs very similarly to Supervised w/o denoising quantitatively and exhibits a similar pseudo-denoising effect in Figure 6.

Although Noise2Recon-SS improved over Standard SSDU, there was a substantial difference between its performance and that of the proposed Robust SSDU both qualitatively and quantitatively. In [27], Noise2Recon-SS was not compared to a fully supervised benchmark; it was only shown to have improved performance compared to Standard SSDU, consistent with the results here. The experimental evaluation in [27] focused on robustness to out of distribution (OOD) shifts, where the training and inference measurement noise variances were not necessarily the same. Another difference was that Noise2Recon-SS's simulated noise in [27] had a standard deviation randomly selected from a fixed range, while the experiments here fixed the simulated noise's standard deviation so that it could be properly compared to the proposed methods.

Robust SSDU required only a few additional cheap computational steps compared to standard training: the addition or multiplication of the further noise and sub-sampling mask, respectively, and the α-based correction upon inference. Accordingly, the compute time and memory requirements of the proposed methods were found to be very similar to those of Supervised w/o denoising or Standard SSDU. In contrast, Noise2Recon-SS used both $M_{\Lambda_t} y_t$ and $y_t + M_{\Omega_t} \widetilde{n}_t$ as the network inputs during training so required twice as many forward passes to train the network compared to Robust SSDU. Accordingly, we found that Noise2Recon-SS required approximately twice as much memory as and took around two times longer per epoch than the proposed methods.

In general, Supervised with BM3D and SSDU with BM3D both performed well qualitatively. We also found that in many cases these methods had an mean SSIM that exceeded even the fully supervised benchmark; see Appendix D for a detailed discussion. However, for some images, such as those shown with the red arrows in Figures 5 and 6, these methods generated potentially clinically misleading artifacts. We believe this to be a consequence of the mismatch between its Gaussian noise model and the actual error of the RSS estimate, which could lead to unreliable noise removal, especially at a high R_Ω. We also found that SSDU with BM3D often led to more oversmoothing and less crisp tissue boundaries than Robust SSDU, which is particularly prominent in the M4Raw examples of Figure 9. Another disadvantage was the computational expense of the BM3D algorithm; we found that the reconstruction time of SSDU with BM3D was around 100 times longer per slice than that of Robust SSDU upon inference.

Another existing method designed for noisy, sub-sampled training data is the robust equivariant imaging (REI) method [48,49]. We did not compare REI as it was designed for reconstruction tasks with a fixed sampling pattern: the Ω_t is the same for all t. This sampling set assumption is central to its use of equivariance and contrasts with the methods proposed here, which assumed that the sampling mask was an instance of a random variable that satisfied $p_j > 0$ everywhere. However, REI's suggestion to use Stein's unbiased risk estimate (SURE) [50] to remove measurement noise would be feasible in combination with SSDU and warrants further investigation in future work.

The theoretical work presented in this paper only applies to the case of ℓ_2 minimization, which can lead to blurry reconstructions. However, it has been established that Standard SSDU can be applied with other losses such as an entry-wise mixed ℓ_1-ℓ_2 loss in k-space [13]. We found that Robust SSDU with an ℓ_2 loss with $\Lambda \cap \Omega$ and mixed ℓ_1-ℓ_2 loss with $\Omega \setminus \Lambda$ also performed competitively with a suitable benchmark in practice (results are not shown for brevity). Future work includes establishing whether Robust SSDU can be modified to be applicable to other loss functions, including potential losses in the RSS image.

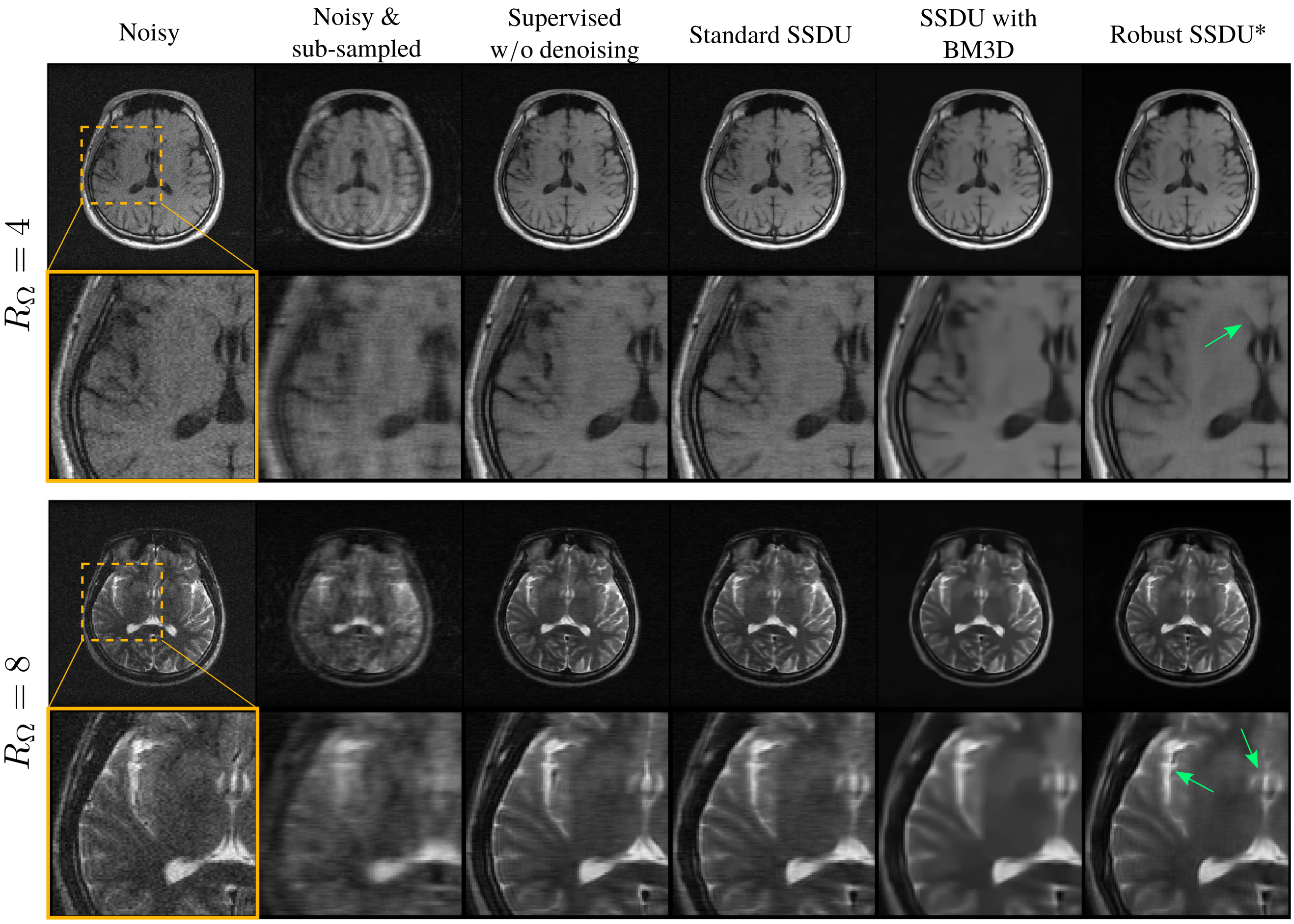

Figure 9. The qualitative performance of the proposed method with the prospectively noisy, low-field dataset M4Raw. While SSDU with BM3D and Robust SSDU (highlighted with an asterisk) both demonstrate a denoising effect, Robust SSDU exhibits improved contrast and visibly sharper boundaries, highlighted by the green arrows.

The methods presented here also assumed that the distribution of M_Ω was fixed; a modification of the method for dealing with a range of sub-sampling patterns and acceleration factors is a potential avenue for future work. It would also be desirable to develop an approach that automatically tunes α and the distribution of M_Λ, whose optimal values are specific to the noise model, M_Ω distribution, and dataset.

The additive Noisier2Noise was designed for Gaussian noise; the α-based correction term applied upon inference is derived on the assumption that the noise is Gaussian [26]. Therefore, the naive application of Robust SSDU would not be expected to perform well for other measurement noise distributions. Future work includes extending the framework to other distributions and sources of error such as other system noise or physiological motion, which has a more complex distribution that may itself be learned [51,52].

Although Denoising VarNet was found to offer improved performance compared to Standard VarNet, the evaluation of possible architectures for simultaneous denoising and reconstruction was not extensive in this paper and warrants future work. For instance, simpler models or the combination of multiple models in parallel [53] may improve computational expense or reconstruction quality in practice. Improvements to the network architecture would be necessary for the more ambitious sub-sampling factors and noise levels investigated in this paper. For instance, at $R_\Omega = 8$ and $\sigma_n = 0.08$, Robust SSDU and the fully supervised benchmark both displayed a significant loss of details at tissue boundaries and were of insufficient quality for clinical applications; see Figure 10.

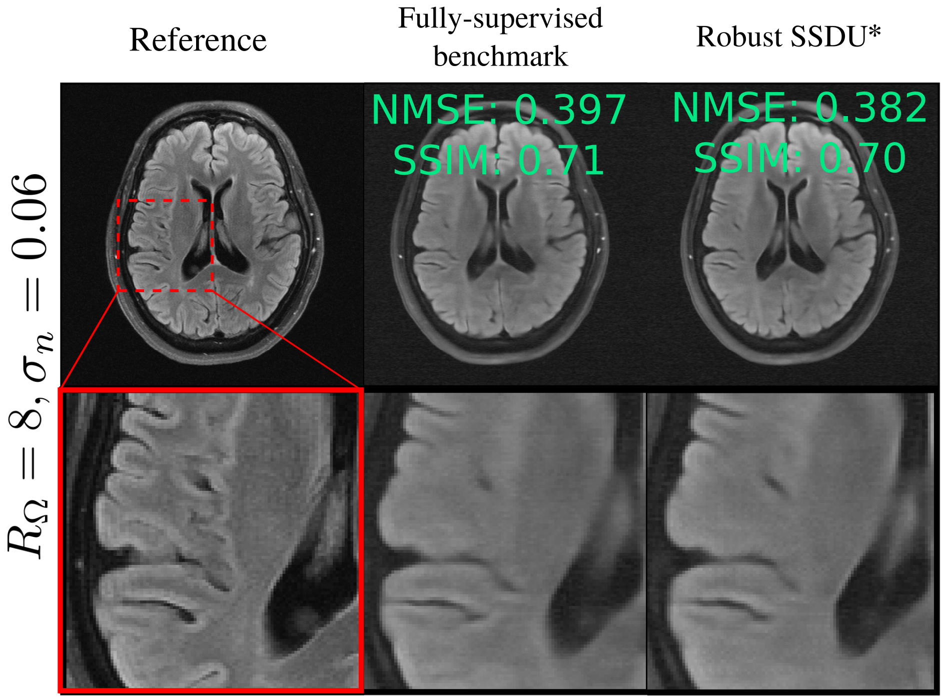

Figure 10. Poor recovery of fine details for ambitious sub-sampling and noise levels at $R_\Omega = 8$ and $\sigma_n = 0.08$ for both the Fully-supervised benchmark, and Robust SSDU (highlighted with an asterisk).

Author Contributions: Conceptualization, C.M. and M.C.; methodology, C.M. and M.C.; software, C.M.; validation, C.M.; formal analysis, C.M.; investigation, C.M.; resources, M.C.; data curation, C.M.; writing—original draft preparation, C.M.; writing—review and editing, M.C.; visualization, C.M.; supervision, M.C.; project administration, M.C.; funding acquisition, M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Engineering and Physical Sciences Research Council, grant EP/T013133/1, by the Royal Academy of Engineering, grant RF201617/16/23, and by the Wellcome Trust, grant 203139/Z/16/Z. The computational aspects of this research were supported by the Wellcome Trust Core Award, grant number 203141/Z/16/Z, and the NIHR Oxford BRC. The views expressed are those of the authors and not necessarily those of the NHS, the NIHR, or the Department of Health. This research was undertaken, in part, thanks to funding from the Canada Research Chairs Program.

Institutional Review Board Statement: Ethical review and approval were waived for this study as confirmed by the license attached with the open access data.

Informed Consent Statement: Patient consent was waived as confirmed by the license attached with the open access data.

Data Availability Statement: This paper uses the public dataset fastMRI, available from https://fastmri.med.nyu.edu, accessed on 1 October 2021.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Proof of SSDU Variant on M_Ω

This appendix proves Claim 1 using a similar approach to Appendix B of [25]. Minimization according to Equation (27) yields a network that satisfies

$$\mathbb{E}[M_\Omega(f_{\theta^*}(\widetilde{Y}) - Y)|\widetilde{Y}] = 0 \tag{A1}$$

We split the conditional expectation into two cases: $\widetilde{Y}_j \neq 0$ and $\widetilde{Y}_j = 0$. Throughout this paper, m_j and $\widetilde{m}_j$ refer to the jth diagonal of M_Ω and M_Λ, respectively.

Case 1 ($\mathbb{E}[m_j(f_{\theta^*}(\widetilde{Y})_j - Y_j)|\widetilde{Y}_j \neq 0]$): When $\widetilde{Y}_j \neq 0$, the measurement model implies that $m_j = 1$ and $Y_j = Y_{0,j} + N_j$. Therefore,

$$\mathbb{E}[m_j(f_{\theta^*}(\widetilde{Y})_j - Y_j)|\widetilde{Y}_j \neq 0] = \mathbb{E}[f_{\theta^*}(\widetilde{Y})_j - Y_{0,j} - N_j|\widetilde{Y}_j \neq 0] \tag{A2}$$

Case 2 ($\mathbb{E}[m_j(f_{\theta^*}(\widetilde{Y})_j - Y_j)|\widetilde{Y}_j = 0]$): We can use the result derived from Equation (27) in (29) in [25], with $Y_{0,j}$ replaced by $Y_{0,j} + N_j$:

$$\mathbb{E}[m_j(f_{\theta^*}(\widetilde{Y})_j - Y_j)|\widetilde{Y}_j = 0] = \mathbb{E}[f_{\theta^*}(\widetilde{Y})_j - Y_{0,j} - N_j|\widetilde{Y}_j = 0] \cdot (1 - k_j) \tag{A3}$$

where

$$k_j = \mathbb{P}[Y_j = 0|\widetilde{Y}_j = 0] = \frac{1 - p_j}{1 - \widetilde{p}_j p_j}. \tag{A4}$$

Combining Cases 1 and 2: Consider the candidate

$$\mathbb{E}[m_j(f_{\theta^*}(\widetilde{Y})_j - Y_j)|\widetilde{Y}_j] = \{1 - k_j(1 - \widetilde{m}_j m_j)\}\mathbb{E}[f_{\theta^*}(\widetilde{Y})_j - Y_{0,j} - N_j|\widetilde{Y}_j].$$

To verify that this expression is correct, we can check that it is consistent with Cases 1 and 2. For Case 1: if $\widetilde{Y}_j \neq 0$, then $\widetilde{m}_j m_j = 1$ and the term in curly brackets is 1, so Equation (A5) is consistent with Equation (A2). For Case 2: if $\widetilde{Y}_j = 0$, then $\widetilde{m}_j m_j = 0$ and the term in curly brackets is $1 - k_j$, so Equation (A5) is consistent with Equation (A3), as required. By Equation (A1),

$$\{1 - k_j(1 - \widetilde{m}_j m_j)\}\mathbb{E}[f_{\theta^*}(\widetilde{Y})_j - Y_{0,j} - N_j|\widetilde{Y}_j] = 0$$

The term in the curly brackets is non-zero for all j if $1 - k_j$ is non-zero for $j \notin \Omega \cap \Lambda$, which is true when Equations (9) and (10) hold, where we note that the special case $\widetilde{p}_j = p_j = 1$ is also allowed since $\widetilde{m}_j m_j = 1$, always. Given this assumption, dividing by the term in the curly brackets gives the following:

$$\mathbb{E}[f_{\theta^*}(\widetilde{Y})_j - Y_{0,j} - N_j|\widetilde{Y}_j] = 0. \tag{A5}$$

Vectorizing gives the required result. □

Appendix B. Proof of Weighted Noisier2Full

To compute the unknown

$$\nabla_\theta \mathbb{E}\left[\|\hat{Y}_{Nr2F} - Y_0\|_2^2|\widetilde{Y}\right]$$

in terms of the known $Y_0 + N$, we compute the contributions to the loss in Ω and Ω^c separately, shown in Lemmas A1 and A2, respectively.

Lemma A1. *Consider the random variables $Y = M_\Omega(Y_0 + N)$ and $\widetilde{Y} = Y + M_\Omega \widetilde{N}$, where N and $\widetilde{N}$ are zero-mean Gaussians distributed with variances of σ_n^2 and $\alpha^2\sigma_n^2$, respectively. For an arbitrary function, f_θ,*

$$\nabla_\theta \mathbb{E}\left[\|M_\Omega(\hat{Y}_{Nr2F} - Y_0)\|_2^2|\widetilde{Y}\right] = \nabla_\theta \mathbb{E}\left[\left\|\frac{1+\alpha^2}{\alpha^2} M_\Omega(f_\theta(\widetilde{Y}) - Y)\right\|_2^2|\widetilde{Y}\right]. \tag{A6}$$

Proof. Using $M_\Omega \widetilde{Y} = M_\Omega(Y + \widetilde{N})$ and $M_\Omega Y_0 = M_\Omega(Y - N)$, the left-hand side of Equation (A6) is

$$\begin{aligned}
&\nabla_\theta \mathbb{E}\left[\left\| M_\Omega\left(\frac{(1+\alpha^2)f_\theta(\widetilde{Y}) - \widetilde{Y}}{\alpha^2} - Y_0\right)\right\|_2^2 |\widetilde{Y}\right] \\
&= \nabla_\theta \mathbb{E}\left[\left\| M_\Omega\left(\frac{(1+\alpha^2)f_\theta(\widetilde{Y}) - Y - \widetilde{N}}{\alpha^2} - Y + N\right)\right\|_2^2 |\widetilde{Y}\right] \\
&= \nabla_\theta \mathbb{E}\left[\left\| M_\Omega\left(\frac{1+\alpha^2}{\alpha^2}(f_\theta(\widetilde{Y}) - Y) + N - \frac{\widetilde{N}}{\alpha^2}\right)\right\|_2^2 |\widetilde{Y}\right] \\
&= \nabla_\theta \mathbb{E}\left[\left\| \frac{1+\alpha^2}{\alpha^2} M_\Omega(f_\theta(\widetilde{Y}) - Y)\right\|_2^2 |\widetilde{Y}\right] + \frac{1+\alpha^2}{\alpha^2}\nabla_\theta \mathbb{E}\left[2 f_\theta(\widetilde{Y})^H M_\Omega\left(N - \frac{\widetilde{N}}{\alpha^2}\right)|\widetilde{Y}\right] \quad \text{(A7)}
\end{aligned}$$

where all the terms in the expansion of the ℓ_2 norm in the last step that are not dependent on θ are zeroed by ∇_θ. Now, we show that the second term on the right-hand side of Equation (A7) is zero. Lemma 3.1 from [26] shows that

$$\mathbb{E}[M_\Omega \widetilde{N}|\widetilde{Y}] = \alpha^2 \mathbb{E}[M_\Omega N|\widetilde{Y}], \quad \text{(A8)}$$

where M_Ω is included as the result only applies to sampled terms. We note that the right-hand side of Equation (A8) scales according to the *variance* of the noise rather than the perhaps more intuitive standard deviation. Following [26], Equation (A8) can be proven by computing the probability $\mathbb{P}[N_j = n|\widetilde{Y}_j, j \in \Omega]$:

$$\begin{aligned}
\mathbb{P}[N_j = n|\widetilde{Y}_j, j \in \Omega] &= \mathbb{P}[N_j = n]\mathbb{P}[\widetilde{N}_j = \widetilde{Y}_j - Y_{0,j} - n_j] \\
&\propto \exp\left(-\frac{n^2}{2\sigma^2}\right)\exp\left(-\frac{(\widetilde{Y}_j - Y_{0,j} - n)^2}{2\alpha^2\sigma^2}\right).
\end{aligned}$$

After some algebraic manipulation not shown here for brevity, this distribution can be shown to have the mean $(\widetilde{Y}_j - Y_{0,j})/(1+\alpha^2)$. A similar computation for $\widetilde{N}_j$ yields a mean of $\alpha^2(\widetilde{Y}_j - Y_{0,j})/(1+\alpha^2)$, giving the jth entry of the relationship stated in Equation (A8), conditional on $j \in \Omega$. Since for the alternative $j \notin \Omega$ both sides are trivially zero, Equation (A8) is correct for all indices.

Applying Equation (A8) to the right-hand side of Equation (A7) gives

$$\mathbb{E}\left[f_\theta(\widetilde{Y})^H M_\Omega\left(N - \frac{\widetilde{N}}{\alpha^2}\right)|\widetilde{Y}\right] = f_\theta(\widetilde{Y})^H \mathbb{E}\left[M_\Omega\left(N - \frac{\widetilde{N}}{\alpha^2}\right)|\widetilde{Y}\right] = 0 \quad \text{(A9)}$$

where the conditional dependence on $\widetilde{Y}$ allows the removal of $f_\theta(\widetilde{Y})$ from the expectation. Therefore, the right-hand side of Equation (A7) equals the right-hand side of Equation (A6), as required. □

Lemma A2. *Consider the random variables Y and $\widetilde{Y}$ as defined in Lemma A1. For an arbitrary function f_θ,*

$$\nabla_\theta \mathbb{E}\left[\left\| M_{\Omega^c}(\hat{Y}_{Nr2F} - Y_0)|\widetilde{Y}\right\|_2^2\right] = \nabla_\theta \mathbb{E}\left[\left\| M_{\Omega^c}(f_\theta(\widetilde{Y}) - Y_0 - N)\right\|_2^2 |\widetilde{Y}\right]. \quad \text{(A10)}$$

Proof. Using $M_{\Omega^c}M_\Omega = 0$ and $M_{\Omega^c}M_{\Omega^c} = M_{\Omega^c}$ and the definition of $\hat{Y}_{Nr2F}$ in Equation (30), we have $M_{\Omega^c}\hat{Y}_{Nr2F} = M_{\Omega^c}f_\theta(\widetilde{Y})$. Therefore, the left-hand side of Equation (A10) is

$$\begin{aligned}&\nabla_\theta\mathbb{E}\left[\left\|M_{\Omega^c}(f_\theta(\widetilde{Y}) - Y_0)\right\|_2^2|\widetilde{Y}\right]\\&= \nabla_\theta\mathbb{E}\left[\left\|M_{\Omega^c}(f_\theta(\widetilde{Y}) - Y_0 - N + N)\right\|_2^2|\widetilde{Y}\right]\\&= \nabla_\theta\mathbb{E}\left[\left\|M_{\Omega^c}(f_\theta(\widetilde{Y}) - Y_0 - N)\right\|_2^2 + 2f_\theta(\widetilde{Y})^H M_{\Omega^c}N|\widetilde{Y}\right]\end{aligned} \tag{A11}$$

where, again, all the terms not dependent on θ are zeroed by ∇_θ. The second term is

$$\mathbb{E}\left[f_\theta(\widetilde{Y})^H M_{\Omega^c}N|\widetilde{Y}\right] = f_\theta(\widetilde{Y})^H\mathbb{E}\left[M_{\Omega^c}N|\widetilde{Y}\right] = 0 \tag{A12}$$

where, as in Equation (16), we use the independence of N from $\widetilde{Y}$ when $j \notin \Omega$. Therefore, Equation (A11) equals the right-hand side of Equation (A10), as required. □

To find the ℓ_2 error of $\hat{Y}_{Nr2F}$, we use $M_\Omega + M_{\Omega^c} = \mathbb{1}$ and sum the results from Lemmas A1 and A2:

$$\begin{aligned}\nabla_\theta\mathbb{E}\left[\|\hat{Y}_{Nr2F} - Y_0\|_2^2|\widetilde{Y}\right] &= \nabla_\theta\mathbb{E}\left[\|(M_\Omega + M_{\Omega^c})(\hat{Y}_{Nr2F} - Y_0)\|_2^2|\widetilde{Y}\right]\\&= \nabla_\theta\mathbb{E}\left[\left\|\left(\frac{1+\alpha^2}{\alpha^2}M_\Omega + M_{\Omega^c}\right)(f_\theta(\widetilde{Y}) - Y_0 - N)\right\|_2^2|\widetilde{Y}\right]\end{aligned}$$

as required.

Appendix C. Proof of Robust SSDU Weighting

Analogous to Appendix B, to compute the unknown

$$\nabla_\theta\mathbb{E}\left[\|\hat{Y}_{RSSDU} - Y_0\|_2^2|\widetilde{Y}\right]$$

in terms of the known sub-sampled, noisy Y, we compute the contributions to the loss from $\Lambda \cap \Omega$ and $(\Lambda \cap \Omega)^c$ separately. For the contribution from $\Lambda \cap \Omega$, an identical approach to the proof in Lemma A1 can be used with Ω replaced by $\Lambda \cap \Omega$ so that

$$\nabla_\theta\mathbb{E}\left[\|M_{\Lambda\cap\Omega}(\hat{Y}_{RSSDU} - Y_0)\|_2^2|\widetilde{Y}\right] = \nabla_\theta\mathbb{E}\left[\left\|\frac{1+\alpha^2}{\alpha^2}M_{\Lambda\cap\Omega}(f_\theta(\widetilde{Y}) - Y)\right\|_2^2|\widetilde{Y}\right]. \tag{A13}$$

The following lemma shows how the remaining loss, which is computed on $\Omega \setminus \Lambda$, can be used to estimate the target ground truth loss, which is over $(\Lambda \cap \Omega)^c$.

Lemma A3. *Consider the random variables $Y = M_\Omega(Y_0 + N)$ and $\widetilde{Y} = M_{\Lambda\cap\Omega}(Y + \widetilde{N})$, where N and $\widetilde{N}$ are zero-mean Gaussians distributed with variances of σ_n^2 and $\alpha^2\sigma_n^2$, respectively. For an arbitrary function f_θ,*

$$\nabla_\theta\mathbb{E}\left[\left\|M_{(\Lambda\cap\Omega)^c}(\hat{Y}_{RSSDU} - Y_0)\right\|_2^2|\widetilde{Y}\right] = \nabla_\theta\mathbb{E}\left[\left\|\mathcal{P}^{1/2}M_{\Omega\setminus\Lambda}(f_\theta(\widetilde{Y}) - Y)\right\|_2^2|\widetilde{Y}\right], \tag{A14}$$

where $\mathcal{P}$ is defined in Equation (3).

Proof. Since $M_{(\Lambda\cap\Omega)^c}\hat{Y}_{RSSDU} = M_{(\Lambda\cap\Omega)^c}f_\theta(\widetilde{Y})$, the left-hand side of Equation (A14) is

$$\nabla_\theta\mathbb{E}\left[\left\|M_{(\Lambda\cap\Omega)^c}(f_\theta(\widetilde{Y}) - Y_0)\right\|_2^2|\widetilde{Y}\right] = \nabla_\theta\mathbb{E}\left[\left\|M_{(\Lambda\cap\Omega)^c}(f_\theta(\widetilde{Y}) - Y_0 - N)\right\|_2^2|\widetilde{Y}\right], \tag{A15}$$

where Lemma A2 with Ω^c replaced by $(\Lambda \cap \Omega)^c$ is used. Using $|\cdot|^2$ to denote the entry-wise magnitude squared, we can write

$$\nabla_\theta \mathbb{E}\left[\left\|M_{(\Lambda\cap\Omega)^c}(f_\theta(\widetilde{Y}) - Y_0 - N)\right\|_2^2 |\widetilde{Y}\right] = \nabla_\theta \mathbb{E}\left[1_q^T M_{(\Lambda\cap\Omega)^c} |f_\theta(\widetilde{Y}) - Y_0 - N|^2 |\widetilde{Y}\right], \tag{A16}$$

where 1_q is a q-dimensional vector of ones. Equation (32) from [25] shows that the conditional expectation of $f_\theta(\widetilde{Y}) - Y$ on $M_{(\Lambda\cap\Omega)^c}$ and $M_{\Omega\setminus\Lambda}$ is related by a factor, $\mathcal{P}$. By repeating that derivation with all instances of $f_\theta(\widetilde{Y}) - Y$ trivially replaced with $|f_\theta(\widetilde{Y}) - Y_0 - N|^2$, a similar relationship can be derived for the latter, yielding the same $\mathcal{P}$ factor; see [25]. In brief, if the jth entry $\widetilde{Y}_j$ is not zero, then the jth diagonal of $M_{\Omega\setminus\Lambda}$ is zero, so

$$\mathbb{E}\left[|M_{\Omega\setminus\Lambda}(f_\theta(\widetilde{Y}) - Y_0 - N)|_j^2 |\widetilde{Y}_j \neq 0\right] = 0. \tag{A17}$$

When the jth entry of $\widetilde{Y}_j$ *is* zero,

$$\mathbb{E}\left[|M_{\Omega\setminus\Lambda}(f_\theta(\widetilde{Y}) - Y_0 - N)|_j^2 |\widetilde{Y}_j = 0\right] = \mathbb{E}\left[\mathcal{P}_{jj}^{-1} |f_\theta(\widetilde{Y}) - Y|_j^2 |\widetilde{Y}_j = 0\right]. \tag{A18}$$

See (31) of [25] for a detailed derivation. By combining both cases from Equations (A17) and (A18), we obtain

$$\mathbb{E}\left[|M_{\Omega\setminus\Lambda}(f_\theta(\widetilde{Y}) - Y_0 - N)|_j^2 |\widetilde{Y}_j\right] = \mathbb{E}\left[\mathcal{P}_{jj}^{-1} |M_{(\Lambda\cap\Omega)^c}(f_\theta(\widetilde{Y}) - Y_0 - N)|_j^2 |\widetilde{Y}_j\right]. \tag{A19}$$

By applying this result to Equation (A16) by multiplying with 1_q^T and bringing the masks outside the entry-wise magnitude, we obtain

$$\begin{aligned}\nabla_\theta \mathbb{E}\left[1_q^T M_{(\Lambda\cap\Omega)^c} |f_\theta(\widetilde{Y}) - Y_0 - N|^2 |\widetilde{Y}\right] &= \nabla_\theta \mathbb{E}\left[1_q^T \mathcal{P} M_{\Omega\setminus\Lambda} |f_\theta(\widetilde{Y}) - Y_0 - N|^2 |\widetilde{Y}\right] \\ &= \nabla_\theta \mathbb{E}\left[\left\|\mathcal{P}^{1/2} M_{\Omega\setminus\Lambda}(f_\theta(\widetilde{Y}) - Y)\right\|_2^2 |\widetilde{Y}\right],\end{aligned}$$

as required. □

To find the ℓ_2 error of $\hat{Y}_{RSSDU}$, we use $M_{\Lambda\cap\Omega} + M_{(\Lambda\cap\Omega)^c} = \mathbb{1}$ and the sum Equations (A13) and (A14):

$$\begin{aligned}\nabla_\theta \mathbb{E}\left[\|\hat{Y}_{RSSDU} - Y_0\|_2^2 |\widetilde{Y}\right] &= \nabla_\theta \mathbb{E}\left[\left\|(M_{\Lambda\cap\Omega} + M_{(\Lambda\cap\Omega)^c})(\hat{Y}_{Nr2F} - Y_0)\right\|_2^2 |\widetilde{Y}\right] \\ &= \nabla_\theta \mathbb{E}\left[\left\|\left(\frac{1+\alpha^2}{\alpha^2} M_{\Lambda\cap\Omega} + \mathcal{P}^{1/2} M_{\Omega\setminus\Lambda}\right)(f_\theta(\widetilde{Y}) - Y)\right\|_2^2 |\widetilde{Y}\right]\end{aligned}$$

as required.

Appendix D. Table of SSIM on Test Set

The mean SSIM of the magnitude images are shown in Table A1. The SSIM of the proposed methods is comparable to the fully supervised benchmark. However, in many cases, the methods that used BM3D outperformed even the fully supervised benchmark, implying that BM3D achieved a better SSIM than the machine learning-based approach to denoising used in this paper. We emphasize that the entirely data-driven approaches were not trained to minimize for the SSIM, and the SSIM would be expected to substantially improve if it was included in the loss function [36].

The methods that used BM3D had a considerably higher standard error, which indicates a substantially higher variation in the quality of the output. We believe that this was a consequence of the mismatch between BM3D's Gaussian noise model and the actual error of the RSS estimate, which led to a higher risk of oversmoothing and artifacts such as those shown in Figures 5 and 6.

Table A1. The methods' test sets' SSIM in the magnitude images with standard errors. The double lines separate the type of training data available and bold font is used to denote the best performance within each category. Asterisks denote proposed methods.

	Acceleration Factor $R_\Omega = 4$				**Acceleration Factor $R_\Omega = 8$**			
	$\sigma_n = 0.02$	$\sigma_n = 0.04$	$\sigma_n = 0.06$	$\sigma_n = 0.08$	$\sigma_n = 0.02$	$\sigma_n = 0.04$	$\sigma_n = 0.06$	$\sigma_n = 0.08$
Noisy and sub-sampled	0.76 ± 0.006	0.64 ± 0.006	0.50 ± 0.006	0.40 ± 0.005	0.72 ± 0.008	0.67 ± 0.007	0.57 ± 0.006	0.48 ± 0.006
Fully supervised benchmark	**0.83 ± 0.007**	**0.75 ± 0.006**	**0.63 ± 0.006**	**0.52 ± 0.005**	**0.75 ± 0.008**	**0.77 ± 0.007**	**0.73 ± 0.007**	**0.67 ± 0.006**
Supervised w/o denoising	0.83 ± 0.006	0.70 ± 0.006	0.55 ± 0.005	0.43 ± 0.005	0.80 ± 0.008	0.74 ± 0.006	0.63 ± 0.006	0.52 ± 0.005
Supervised with BM3D	**0.86 ± 0.025**	**0.75 ± 0.042**	**0.64 ± 0.043**	**0.55 ± 0.041**	**0.85 ± 0.014**	**0.78 ± 0.033**	0.69 ± 0.039	0.61 ± 0.039
Unweighted Noisier2Full *	0.83 ± 0.007	**0.75 ± 0.006**	0.63 ± 0.006	0.52 ± 0.005	0.76 ± 0.008	0.77 ± 0.007	**0.73 ± 0.007**	**0.66 ± 0.006**
Noisier2Full *	0.82 ± 0.007	0.74 ± 0.006	0.62 ± 0.006	0.50 ± 0.005	0.74 ± 0.008	0.76 ± 0.007	0.72 ± 0.007	0.65 ± 0.006
Standard SSDU	0.83 ± 0.004	0.69 ± 0.004	0.55 ± 0.003	0.43 ± 0.003	0.79 ± 0.005	0.74 ± 0.004	0.63 ± 0.004	0.52 ± 0.003
SSDU with BM3D	**0.86 ± 0.025**	**0.75 ± 0.042**	**0.64 ± 0.043**	**0.56 ± 0.041**	**0.84 ± 0.014**	**0.78 ± 0.033**	0.69 ± 0.039	0.61 ± 0.039
Noise2Recon-SS	0.83 ± 0.006	0.71 ± 0.006	0.56 ± 0.005	0.47 ± 0.005	0.79 ± 0.008	0.73 ± 0.006	0.66 ± 0.006	0.56 ± 0.005
Unweighted Robust SSDU *	0.83 ± 0.007	**0.75 ± 0.006**	0.62 ± 0.006	0.51 ± 0.005	0.75 ± 0.008	0.77 ± 0.007	**0.72 ± 0.006**	**0.65 ± 0.006**
Robust SSDU *	0.83 ± 0.007	0.74 ± 0.006	0.62 ± 0.006	0.50 ± 0.005	0.75 ± 0.008	0.76 ± 0.007	**0.72 ± 0.007**	**0.65 ± 0.006**

References

1. Bustin, A.; Fuin, N.; Botnar, R.M.; Prieto, C. From compressed-sensing to artificial intelligence-based cardiac MRI reconstruction. *Front. Cardiovasc. Med.* **2020**, *7*, 17. [CrossRef] [PubMed]
2. Pruessmann, K.P.; Weiger, M.; Scheidegger, M.B.; Boesiger, P. SENSE: Sensitivity encoding for fast MRI. *Magn. Reson. Med.* **1999**, *42*, 952–962. [CrossRef]
3. Lustig, M.; Donoho, D.; Pauly, J.M. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magn. Reson. Med.* **2007**, *58*, 1182–1195. [CrossRef] [PubMed]
4. Ye, J.C. Compressed sensing MRI: A review from signal processing perspective. *BMC Biomed. Eng.* **2019**, *1*, 8. [CrossRef]
5. Wang, S.; Su, Z.; Ying, L.; Peng, X.; Zhu, S.; Liang, F.; Feng, D.; Liang, D. Accelerating magnetic resonance imaging via deep learning. In Proceedings of the 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI), Prague, Czech Republic, 13–16 April 2016; pp. 514–517.
6. Kwon, K.; Kim, D.; Park, H. A parallel MR imaging method using multilayer perceptron. *Med. Phys.* **2017**, *44*, 6209–6224. [CrossRef]
7. Hammernik, K.; Klatzer, T.; Kobler, E.; Recht, M.P.; Sodickson, D.K.; Pock, T.; Knoll, F. Learning a variational network for reconstruction of accelerated MRI data. *Magn. Reson. Med.* **2018**, *79*, 3055–3071. [CrossRef] [PubMed]
8. Uecker, M.; Zhang, S.; Voit, D.; Karaus, A.; Merboldt, K.D.; Frahm, J. Real-time MRI at a resolution of 20 ms. *NMR Biomed.* **2010**, *23*, 986–994. [CrossRef]
9. Haji-Valizadeh, H.; Rahsepar, A.A.; Collins, J.D.; Bassett, E.; Isakova, T.; Block, T.; Adluru, G.; DiBella, E.V.; Lee, D.C.; Carr, J.C.; et al. Validation of highly accelerated real-time cardiac cine MRI with radial k-space sampling and compressed sensing in patients at 1.5 T and 3T. *Magn. Reson. Med.* **2018**, *79*, 2745–2751. [CrossRef] [PubMed]
10. Lim, Y.; Zhu, Y.; Lingala, S.G.; Byrd, D.; Narayanan, S.; Nayak, K.S. 3D dynamic MRI of the vocal tract during natural speech. *Magn. Reson. Med.* **2019**, *81*, 1511–1520. [CrossRef]

11. Tamir, J.I.; Stella, X.Y.; Lustig, M. Unsupervised deep basis pursuit: Learning reconstruction without ground-truth data. In Proceedings of the 27th Annual ISMRM Annual Meeting, Montréal, QC, Canada, 11–16 May 2019.
12. Huang, P.; Zhang, C.; Li, H.; Gaire, S.K.; Liu, R.; Zhang, X.; Li, X.; Ying, L. Deep MRI reconstruction without ground truth for training. In Proceedings of the 27th Annual ISMRM Annual Meeting, Montréal, QC, Canada, 11–16 May 2019.
13. Yaman, B.; Hosseini, S.A.H.; Moeller, S.; Ellermann, J.; Uğurbil, K.; Akçakaya, M. Self-supervised learning of physics-guided reconstruction neural networks without fully sampled reference data. *Magn. Reson. Med.* **2020**, *84*, 3172–3191. [CrossRef] [PubMed]
14. Aggarwal, H.K.; Pramanik, A.; Jacob, M. ENSURE: Ensemble Stein's unbiased risk estimator for unsupervised learning. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 1160–1164.
15. Acar, M.; Çukur, T.; Öksüz, İ. Self-supervised Dynamic MRI Reconstruction. In *Machine Learning for Medical Image Reconstruction*; Haq, N., Johnson, P., Maier, A., Würfl, T., Yoo, J., Eds.; Springer International Publishing: Cham, The Switzerland, 2021; pp. 35–44.
16. Yaman, B.; Shenoy, C.; Deng, Z.; Moeller, S.; El-Rewaidy, H.; Nezafat, R.; Akçakaya, M. Self-Supervised Physics-Guided Deep Learning Reconstruction for High-Resolution 3D LGE CMR. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 100–104. [CrossRef]
17. Demirel, O.B.; Yaman, B.; Dowdle, L.; Moeller, S.; Vizioli, L.; Yacoub, E.; Strupp, J.; Olman, C.A.; Uğurbil, K.; Akçakaya, M. Improved Simultaneous Multi-Slice Functional MRI Using Self-supervised Deep Learning. In Proceedings of the 2021 55th Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 31 October–3 November 2021; pp. 890–894. [CrossRef]
18. Zhou, B.; Dey, N.; Schlemper, J.; Salehi, S.S.M.; Liu, C.; Duncan, J.S.; Sofka, M. DSFormer: A dual-domain self-supervised transformer for accelerated multi-contrast MRI reconstruction. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–7 January 2023; pp. 4966–4975.
19. Obungoloch, J.; Harper, J.R.; Consevage, S.; Savukov, I.M.; Neuberger, T.; Tadigadapa, S.; Schiff, S.J. Design of a sustainable prepolarizing magnetic resonance imaging system for infant hydrocephalus. *Magn. Reson. Mater. Physics, Biol. Med.* **2018**, *31*, 665–676. [CrossRef] [PubMed]
20. Koonjoo, N.; Zhu, B.; Bagnall, G.C.; Bhutto, D.; Rosen, M. Boosting the signal-to-noise of low-field MRI with deep learning image reconstruction. *Sci. Rep.* **2021**, *11*, 8248. [CrossRef]
21. Schlemper, J.; Salehi, S.S.M.; Lazarus, C.; Dyvorne, H.; O'Halloran, R.; de Zwart, N.; Sacolick, L.; Stein, J.M.; Rueckert, D.; Sofka, M.; et al. Deep learning MRI reconstruction in application to point-of-care MRI. *Proc. Intl. Soc. Mag. Reson. Med.* **2020**, *28*, 991.
22. Xie, Y.; Wang, Z.; Ji, S. Noise2same: Optimizing a self-supervised bound for image denoising. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 20320–20330.
23. Lehtinen, J.; Munkberg, J.; Hasselgren, J.; Laine, S.; Karras, T.; Aittala, M.; Aila, T. Noise2Noise: Learning image restoration without clean data. *arXiv* **2018**, arXiv:1803.04189.
24. Batson, J.; Royer, L. Noise2self: Blind denoising by self-supervision. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 524–533.
25. Millard, C.; Chiew, M. A Theoretical Framework for Self-Supervised MR Image Reconstruction Using Sub-Sampling via Variable Density Noisier2Noise. *IEEE Trans. Comput. Imaging* **2023**, *9*, 707–720. [CrossRef]
26. Moran, N.; Schmidt, D.; Zhong, Y.; Coady, P. Noisier2Noise: Learning to denoise from unpaired noisy data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12064–12072.
27. Desai, A.D.; Ozturkler, B.M.; Sandino, C.M.; Boutin, R.; Willis, M.; Vasanawala, S.; Hargreaves, B.A.; Ré, C.; Pauly, J.M.; Chaudhari, A.S. Noise2Recon: Enabling SNR-robust MRI reconstruction with semi-supervised and self-supervised learning. *Magn. Reson. Med.* **2023**, *90*, 2052–2070. [CrossRef]
28. Berk, R.A. *Statistical Learning from a Regression Perspective*; Springer: Berlin/Heidelberg, Germany, 2008; Volume 14.
29. Tian, C.; Fei, L.; Zheng, W.; Xu, Y.; Zuo, W.; Lin, C.W. Deep learning on image denoising: An overview. *Neural Netw.* **2020**, *131*, 251–275. [CrossRef]
30. Krull, A.; Buchholz, T.O.; Jug, F. Noise2void-learning denoising from single noisy images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2129–2137.
31. Huang, T.; Li, S.; Jia, X.; Lu, H.; Liu, J. Neighbor2neighbor: Self-supervised denoising from single noisy images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14781–14790.
32. Hansen, M.S.; Kellman, P. Image reconstruction: An overview for clinicians. *J. Magn. Reson. Imaging* **2015**, *41*, 573–585. [CrossRef]
33. Zeng, G.; Guo, Y.; Zhan, J.; Wang, Z.; Lai, Z.; Du, X.; Qu, X.; Guo, D. A review on deep learning MRI reconstruction without fully sampled k-space. *BMC Med. Imaging* **2021**, *21*, 195. [CrossRef]
34. Wang, F.; Qi, H.; De Goyeneche, A.; Heckel, R.; Lustig, M.; Shimron, E. K-band: Self-supervised MRI Reconstruction via Stochastic Gradient Descent over K-space Subsets. *arXiv* **2023**, arXiv:2308.02958.
35. Wiedemann, S.; Heckel, R. A Deep Learning Method for Simultaneous Denoising and Missing Wedge Reconstruction in Cryogenic Electron Tomography. *arXiv* **2023**, arXiv:2311.05539. [CrossRef]
36. Zbontar, J.; Knoll, F.; Sriram, A.; Murrell, T.; Huang, Z.; Muckley, M.J.; Defazio, A.; Stern, R.; Johnson, P.; Bruno, M.; et al. fastMRI: An open dataset and benchmarks for accelerated MRI. *arXiv* **2018**, arXiv:1811.08839.

37. Lyu, M.; Mei, L.; Huang, S.; Liu, S.; Li, Y.; Yang, K.; Liu, Y.; Dong, Y.; Dong, L.; Wu, E.X. M4Raw: A multi-contrast, multi-repetition, multi-channel MRI k-space dataset for low-field MRI research. *Sci. Data* **2023**, *10*, 264. [CrossRef] [PubMed]
38. Hendriksen, A.A.; Pelt, D.M.; Batenburg, K.J. Noise2Inverse: Self-supervised deep convolutional denoising for tomography. *IEEE Trans. Comput. Imaging* **2020**, *6*, 1320–1335. [CrossRef]
39. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc.* **2007**, *16*, 2080–2095. [CrossRef]
40. Millard, C.; Hess, A.T.; Mailhe, B.; Tanner, J. Approximate Message Passing with a Colored Aliasing Model for Variable Density Fourier Sampled Images. *IEEE Open J. Signal Process.* **2020**, *1*, 146–158. [CrossRef]
41. Virtue, P.; Lustig, M. The Empirical Effect of Gaussian Noise in Undersampled MRI Reconstruction. *Tomography* **2017**, *3*, 211–221. [CrossRef] [PubMed]
42. Sriram, A.; Zbontar, J.; Murrell, T.; Defazio, A.; Zitnick, C.L.; Yakubova, N.; Knoll, F.; Johnson, P. End-to-end variational networks for accelerated MRI reconstruction. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Lima, Peru, 4–8 October 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 64–73.
43. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
44. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
45. Yaman, B.; Hosseini, S.A.H.; Moeller, S.; Ellermann, J.; Uğurbil, K.; Akçakaya, M. Ground-Truth Free Multi-Mask Self-Supervised Physics-Guided Deep Learning in Highly Accelerated MRI. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 1850–1854. [CrossRef]
46. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
47. Zhao, R.; Yaman, B.; Zhang, Y.; Stewart, R.; Dixon, A.; Knoll, F.; Huang, Z.; Lui, Y.W.; Hansen, M.S.; Lungren, M.P. fastMRI+, Clinical pathology annotations for knee and brain fully sampled magnetic resonance imaging data. *Sci. Data* **2022**, *9*, 152. [CrossRef]
48. Chen, D.; Tachella, J.; Davies, M.E. Robust equivariant imaging: A fully unsupervised framework for learning to image from noisy and partial measurements. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5647–5656.
49. Chen, D.; Tachella, J.; Davies, M.E. Equivariant imaging: Learning beyond the range space. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 4379–4388.
50. Stein, C.M. Estimation of the Mean of a Multivariate Normal Distribution. *Ann. Stat.* **1981**, *9*, 1135–1151. [CrossRef]
51. Kumar, G.P.; Vijay Arputharaj, J.; Kumar, P.R.; Kumar, D.V.; Satyanarayana, B.V.V.; Budumuru, P.R. A Comprehensive Review on Image Restoration Methods due to Salt and Pepper Noise. In Proceedings of the 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 11–13 December 2023; pp. 562–567. [CrossRef]
52. Spieker, V.; Eichhorn, H.; Hammernik, K.; Rueckert, D.; Preibisch, C.; Karampinos, D.C.; Schnabel, J.A. Deep learning for retrospective motion correction in MRI: A comprehensive review. *IEEE Trans. Med. Imaging* **2023** , *43*, 846–859. [CrossRef]
53. Hu, C.; Li, C.; Wang, H.; Liu, Q.; Zheng, H.; Wang, S. Self-supervised learning for mri reconstruction with a parallel network training framework. In Proceedings of the Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, 27 September–1 October 2021; Proceedings, Part VI 24; Springer: Berlin/Heidelberg, Germany, 2021; pp. 382–391.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.