Appendix to "Sampling possible reconstructions of undersampled acquisitions in MR imaging with a deep learned prior"

Kerem C. Tezcan[†], Neerav Karani[†], Christian F. Baumgartner[†,‡], Ender Konukoglu[†]

[†] Computer Vision Lab, ETH Zürich
[‡] Machine Learning in Medical Image Analysis Group, University of Tübingen

A Derivation of the closed form of $p(y|z)$

As mentioned in the main text the form that is easiest to interpret is given by the marginalization, however this integral difficult to evaluate directly. Instead, we do this by using conjugacy relations for Normal distributions. We begin by writing

$$p(y|x,z)p(x|z) = p(y|z)p(x|y,z). \tag{1}$$

Since $p(y|x,z)$ and $p(x|z)$ are Normal distributions, due to the conjugacy, the posterior $p(x|y,z)$ is also a Normal distribution given as $N(\mu_{post}, \Sigma_{post})$. Then

$$p(y|z) = \frac{p(y|x,z)p(x|z)}{N(\mu_{post}, \Sigma_{post})}, \quad \text{or} \quad p(y|z)N(\mu_{post}, \Sigma_{post}) = p(y|x,z)p(x|z). \tag{2}$$

Hence the posterior $p(y|z)$ acts as a normalizer to the product distribution to yield a Gaussian. We derive $p(y|z)$ using this relation in Eqn. 2. In the following we also use the conditional independence $p(y|x,z) = p(y|x)$ meaning that when the image is given, this posterior distribution in the k-space is determined without the need for the latent variable. For the derivation we use this strategy: i) we first write the product of the two distributions $p(y|x)p(x|z)$, ii) then recognize the mean and covariance of the Normal posterior distribution $N(\mu_{post}, \Sigma_{post})$ in this, iii) and separate a Gaussian with these parameters from the whole expression. What is left gives us the target distribution.

The product can be written as

$$p(y|x)p(x|z) = \tag{3}$$

$$\det(2\pi\Sigma_{ns})^{-1/2} \det(2\pi\Sigma_x)^{-1/2} \exp\left\{-\frac{1}{2}\left[(y-Ex)^H \Sigma_{ns}^{-1}(y-Ex)\right]\right\} \tag{4}$$

$$\cdot \exp\left\{-\frac{1}{2}\left[(x-\mu_x)^H \Sigma_x^{-1}(x-\mu_x)\right]\right\} \tag{5}$$

$$= \det(2\pi\Sigma_{ns})^{-1/2} \det(2\pi\Sigma_x)^{-1/2} \exp\Bigg\{ -\frac{1}{2} x^H \underbrace{(\Sigma_x^{-1} + E^H \Sigma_{ns}^{-1} E)}_{\Sigma_{post}^{-1}} x \tag{6}$$

$$+ Re\{x^H \underbrace{(E^H \Sigma_{ns}^{-1} y + \Sigma_x^{-1}\mu_x)}_{\Sigma_{post}^{-1}\mu_{post}}\} - \frac{1}{2} y^H \Sigma_{ns}^{-1} y - \frac{1}{2}\mu_x^H \Sigma_x^{-1}\mu_x \Bigg\}, \tag{7}$$

where we have recognized the parameters of the posterior. With these we have enough information to complete the posterior Gaussian. We can replace the terms with posterior parameters and add

the missing term $\pm\frac{1}{2}\mu_{post}^H\Sigma_{post}^{-1}\mu_{post}$ to complete the quadratic form as well as the normalizing determinant $\det(2\pi\Sigma_{post})^{\pm 1/2}$, which yields

$$= \det(2\pi\Sigma_{ns})^{-1/2}\det(2\pi\Sigma_x)^{-1/2}\det(2\pi\Sigma_{post})^{+1/2}\det(2\pi\Sigma_{post})^{-1/2} \tag{8}$$

$$\cdot \exp\Bigg\{ \underbrace{-\frac{1}{2}x^H\Sigma_{post}^{-1}x + Re\{x^H\Sigma_{post}^{-1}\mu_{post}\} - \frac{1}{2}\mu_{post}^H\Sigma_{post}^{-1}\mu_{post}}_{-\frac{1}{2}(x-\mu_{post})^H\Sigma_{post}^{-1}(x-\mu_{post})} \tag{9}$$

$$+ \frac{1}{2}\mu_{post}^H\Sigma_{post}^{-1}\mu_{post} - \frac{1}{2}y^H\Sigma_{ns}^{-1}y - \frac{1}{2}\mu_x^H\Sigma_x^{-1}\mu_x\Bigg\}. \tag{10}$$

We can combine the quadratic term in the exponent with the determinant term and obtain the complete posterior Gaussian. In this case the expression becomes

$$p(y|x)p(x|z) = N(\mu_{post},\Sigma_{post})\det(2\pi\Sigma_{ns})^{-1/2}\det(2\pi\Sigma_x)^{-1/2}\det(2\pi\Sigma_{post})^{+1/2} \tag{11}$$

$$\cdot \exp\Bigg\{ + \frac{1}{2}\mu_{post}^H\Sigma_{post}^{-1}\mu_{post} - \frac{1}{2}y^H\Sigma_{ns}^{-1}y - \frac{1}{2}\mu_x^H\Sigma_x^{-1}\mu_x\Bigg\}. \tag{12}$$

Remembering Eqn. 2, we obtain

$$p(y|z) = \frac{\det(2\pi\Sigma_{post})^{+1/2}}{\det(2\pi\Sigma_{ns})^{1/2}\det(2\pi\Sigma_x)^{1/2}}\cdot\exp\Bigg\{ -\frac{1}{2}y^H\Sigma_{ns}^{-1}y + \frac{1}{2}\mu_{post}^H\Sigma_{post}^{-1}\mu_{post} - \frac{1}{2}\mu_x^H\Sigma_x^{-1}\mu_x\Bigg\}. \tag{13}$$

Now taking the logarithm and leaving out the terms that are independent of z we can arrive at the expression we use as

$$\log p(y|z) = +\frac{1}{2}\mu_{post}^H\Sigma_{post}^{-1}\mu_{post} - \frac{1}{2}\mu_x^H\Sigma_x^{-1}\mu_x + C, \tag{14}$$

where C denotes some constant with z. Notice that we could leave out the determinant term in the nominator due to our model choice of constant Σ_x.

Now we need the closed form expression for the first term in the above equation. Also we need to arrive at this using the terms we have access to from the above equations 6 and 7, namely $\Sigma_{post}^{-1}\mu_{post}$ and Σ_{post}^{-1}. First we write $\mu_{post} = (\Sigma_{post}^{-1})^{-1}\Sigma_{post}^{-1}\mu_{post}$ and rewrite the target term as $\mu_{post}^H\Sigma_{post}^{-1}\mu_{post} = (\Sigma_{post}^{-1}\mu_{post})^H\mu_{post}$. Combining the expressions and isoliting the terms constant with z as C then yields

$$\mu_{post}^H\Sigma_{post}^{-1}\mu_{post} = \mu_x^H\Sigma_x^{-1}(\Sigma_x^{-1} + E^H\Sigma_{ns}^{-1}E)^{-1}\Sigma_x^{-1}\mu_x \tag{15}$$

$$+ 2\mathrm{Re}\left\{y^H\Sigma_{ns}^{-1}E(\Sigma_x^{-1} + E^H\Sigma_{ns}^{-1}E)^{-1}\Sigma_x^{-1}\mu_x\right\} + C \tag{16}$$

Applying the Woodburry identity on the term $(\Sigma_x^{-1} + E^H\Sigma_{ns}^{-1}E)$ followed by some algebraic manipulations reveals that this is equivalent to the expression given in [1].

B Derivation of the closed form solution of $p(x|z,y)$ in k-space

Here we derive and present the mean and covariance parameters of the posterior distribution $p(x|y,z)$. First we write

$$p(x|y,z) = \frac{p(x|z)p(y|x,z)}{p(y|z)} = \frac{p(x|z)p(y|x)}{p(y|z)}, \tag{17}$$

using the model assumption that given the image, k-space is independent of the latent variable, i.e. $p(y|x,z) = p(y|x)$. We then write the two distributions on the nominator: i) $p(x|z) = N(x;\mu_x,\Sigma_x)$

and ii) $p(y|x) = N(y; Ex, \Sigma_{ns})$. Since both of these are Normal, the posterior is also Normal due to conjugacy. However, instead of working in the image space, we prefer to derive the solution in the k-space due to reasons which will be evident later. To this end we write our variable of interest as

$$k = FSB\varphi Px, \tag{18}$$

with individual terms explained as in the next section. But in essence k is the encoded k-space version of the image variable without the undersampling operator. Notice that as there is no undersampling, we can always recover the image from k as

$$x = P^H \varphi^H B^H S^H F^H k, \tag{19}$$

assuming the coil maps are normalized, i.e. $S^H S = I$. Furthermore as the encoding operation is linear the resulting variable k is also Normal distributed. First we write $p(k|z)$ as

$$p(k|z) = N(k; \mu_k, \Sigma_k) = N(k; FSB\varphi P\mu_x, \left[FSB\varphi P\Sigma_x^{-1} P^H \varphi^H B^H S^H F^H\right]^{-1}). \tag{20}$$

Similarly the data likelihood term becomes

$$p(y|k) = N(y; Uk, \Sigma_{ns}), \tag{21}$$

i.e. y is the noisy observation of the undersampled version of the k-space variable k. We now write the product again in terms of k as

$$p(k|z, y) = N(k; \mu_{k|z,y}, \Sigma_{k|z,y}) = p(k|z)p(y|k). \tag{22}$$

We can then write the product of these two Normal distributions, complete the square in the exponent and arrive at the resulting mean and covariance matrix as

$$\mu_{k|z,y} = \left[FSB\varphi P\Sigma_x^{-1} P^H \varphi^H B^H S^H F^H + U^H \Sigma_{ns} U\right]^{-1} \left[FSB\varphi P\Sigma_x^{-1} P^H \varphi^H B^H B\varphi P\mu_x + U^H \Sigma_{ns}^{-1} y\right] \tag{23}$$

and

$$\Sigma_{k|z,y} = \left[FSB\varphi P\Sigma_x^{-1} P^H \varphi^H B^H S^H F^H + U^H \Sigma_{ns} U\right]. \tag{24}$$

As also stated in the main text, we implement sampling from this distribution as taking the mean for each latent z^t sample, i.e. $k^t = \mu_{k|z^t,y}$. However, the matrix inversion in Eq. 23 is not analytically solvable as the involved matrices are too big, hence we use conjugate gradients to solve the matrix inversion to obtain the solution $\mu_{k|z^t,y}$ for a given z^t. To obtain the image x, we then simply take the inverse operations on k as given in Eq. 19 as

$$x^t = P^H \varphi^H B^H S^H F^H k^t. \tag{25}$$

To make the expression easier to read, we define the fully sampled encoding operation $E_F \triangleq FSB\varphi P$ by only removing the undersampling operation from the usual encoding operation, i.e. $E = UE_F$. Then we can rewrite Eq. 25 compactly as

$$x^t = E_F^H \left[E_F \Sigma_x^{-1} E_F^H + U^H \Sigma_{ns} U\right]^{-1} \left[E_F \Sigma_x^{-1} E_F^H \mu_x + U^H \Sigma_{ns} y\right] \tag{26}$$

C Description of the vanilla variational autoencoder (VAE) model

Here we describe the variational autoencoder model [2, 3] for completeness.

The VAE is essentially an unsupervised learning based density estimation method. It learns a function called the evidence lower bound (ELBO) that is a lower bound to the target probability density. Here we describe the vanilla VAE and refer the reader to the next section for the description of the 2D latent space architecture.

The basic equation of VAE can be derived by writing

$$\log p(x) = \log \frac{p(x,z)}{p(z|x)} \tag{27}$$

for images $x \in \mathbb{R}^W$ and latent vectors $z \in \mathbb{R}^V$ (generally $V \leq W$) with a simple prior $p(z)$. We can then introduce an auxiliary distribution and rewrite as

$$\log p(x) = \log \frac{p(x,z)}{p(z|x)} \frac{q(z|x)}{q(z|x)} = \log \frac{p(x,z)}{q(z|x)} \frac{q(z|x)}{p(z|x)}. \tag{28}$$

Then taking an expectation of both sides with $q(z|x)$ yields

$$\log p(x) = \mathbb{E}_{q(z|x)}[\log p(x|z)] - KL[q(z|x)||p(z)] + KL[q(z|x)||p(z|x)], \tag{29}$$

where KL denotes the Kullback-Leibler divergence. The VAE is trained to maximize the first two terms, called the evidence lower bound (ELBO) with $ELBO(x) = \mathbb{E}_{q(z|x)}[\log p(x|z)] - KL[q(z|x)||p(z)]$, which minimizes the rightmost KL term. After the training, the rightmost KL term becomes small and the ELBO approximates the true distribution, i.e. $\log p(x) \approx ELBO(x)$.

The realization of VAE is done as follows: first an image x is passed through a neural network mapping called the encoder with parameters θ^{enc} that predicts the distribution $q_{\theta^{enc}}(z|x)$ for the latent variable z. This $q(z|x)$ distribution is parameterized using a Normal distribution, i.e. $q(z|x) = N(\mu_{lat}, I \cdot \sigma_{lat})$, where I is the identity matrix, hence the encoder function outputs the two variables $\mu_{lat} \in \mathbb{R}^V$ $\sigma_{lat} \in \mathbb{R}^V_+$. The decoder mapping is again a neural network with parameters θ^{dec}, which outputs the distribution $p_{\theta^{dec}}(x|z) = N(\mu_{out}, I \cdot \sigma_{out})$. Then the VAE takes samples $z^l \sim q(z|x) = N(\mu_{lat}, I \cdot \sigma_{lat})$ and decodes these using the decoder mapping as $p(x|z^l)$. Then the VAE is trained using training samples to maximize the ELBO by optimizing for the network weights θ^{enc} and θ^{dec}.

D The 2D latent space VAE architecture

All convolutions are padded and have a kernel size (3, 3) and stride (1, 1) and use a ReLU unless noted otherwise.

The encoder begins with four convolutional layers with 32, 64, 64, 64 output channels, respectively. Then a convolutional layer with kernel size (14, 14), stride (19, 19) and 60 output channels produces the mean of $q(z|x)$ from the fourth layer. Similarly another convolutional layer from the third layer produces the log standard deviation values for $q(z|x)$ with a kernel size of (14, 14), stride (19, 19), without ReLU and 60 output channels. The network is fully convolutional, hence can work with different image sizes. Assuming an input image size of 252x308 for demonstration, the latent space size becomes bx18x22x60, where b is the batch size. We use the usual reparameterization trick to sample z's [2]. At the beginning of the decoder, we apply a scheme of increasing channel dimensions and using these to increase spatial dimensions. We do this in two steps, once for the first image dimension and once again for the second image dimension to obtain a proper reshaping while using the implementation of Tensorflow's reshaping function. First convolutional layer of the decoder does not use ReLU and has 64·19=1216 output channels, resulting in a tensor of size bx18x22x1216. The output of this layer is first transposed to bx18x1216x22 and reshaped to bx252x64x22. This layer then gets transposed to bx252x22x64, then goes through a convolutional layer with again 1216 output channels and without ReLU and becomes bx252x22x1216. This then gets reshaped again to yield a tensor size of bx252x308x64, which is the input image size. This tensor then goes through a ReLU. We then apply 6 convolutional layers with each 60 output channels. Finally another convolutional layer with 1 output channel yields the mean prediction.

E The extended encoding matrix

As mentioned in the main text, we extend the usual encoding operation in the MR acquisition model to consider additional effects of the image acquisition process. Typically, the encoding oper-

ation consists of the coil sensitivities [4], the Fourier transform and the undersampling operation. We include four additional factors in E. The goal of these extensions is to integrate acquisition-specific knowledge to make the image corresponding to the observed k-space data as similar as possible to the VAE's training images.

Let $\tilde{E} = UFS$ denote the usual MR encoding matrix, where $S : \mathbb{C}^N \to \mathbb{C}^{Nc}$ is the sensitivity encoding matrix [4] with c coils, $F : \mathbb{C}^{Nc} \to \mathbb{C}^{Nc}$ is the coil-wise Fourier transform and $U : \mathbb{C}^{Nc} \to \mathbb{C}^{Mc}$ is the undersampling operation. ven as:

$$E = \tilde{E}B\varphi Ps. \tag{30}$$

where P is a padding operator, φ is an operator that incorporates phase information, B models the bias field in the acquisition and s is a scaling factor. We now describe each of them in more detail.

E.0.1 Padding operator, P

The role of P is to minimize any field of view (FOV) differences between the image corresponding to the given k-space data and the space of training images of the VAE. Although our fully convolutional architecture is agnostic to the image size, the empirical prior is estimated for a specific resolution and FOV. Thus, the k-space size can be different due to varying FOV during acquisition. P bridges this gap by padding or cropping the test image to make its size similar to that of the VAE's training images.

E.0.2 Phase matrix, φ

For computational as well as implementation related purposes, we assume that the phase of structural images is highly independent of the magnitude image and smooth. Hence, the same phase image is used for all our posterior samples. This allows us to separate the magnitude and the phase of the image and run the sampling only on the magnitude of the image. However, note that this assumption is not a requirement for the proposed method (as the phase could be sampled as well) but rather a methodological simplification motivated by empirical observations. Following this assumption, we write the phase as a diagonal matrix φ acting on the image.

E.0.3 Bias field matrix, B

We use a diagonal matrix, B, to explicitly model the bias field in the acquisition [5]. The MR images unavoidably have a bias field due to several factors [6]. However, it is easy to estimate it from the measured data. As the bias varies between different acquisitions, this is a potential source of discrepancy between the test image and the VAE's training images. In order to minimize such a discrepancy, we train the VAE on bias free images. Thus, the samples obtained from the VAE are also free of bias fields. However, as the measured data y has the bias field in it, we estimate this field and apply it to the sampled images.

E.0.4 Scaling factor, s

Finally, we introduce an intensity scale factor to make the data likelihood invariant to any scaling difference between the samples and the k-space. During the random walk in the latent space, the corresponding images might get scaled at each step, meaning the image may be multiplied globally by a scale factor. If this scale factor moves away from 1, this causes the data likelihood to increase, since the scales of the k-space data and the image samples do not match. However, from the perspective of sample quality, this does not pose a problem as long as the scaling factors are known. The sampled images can be brought to the same scale by multiplying them with the inverse of the scaling factor. Furthermore, allowing the scale factor to be different for each sample, allows more freedom to the random walk in the latent space, as it is less constrained by the increase in data likelihood due to scale changes. Hence, such scale invariance is desirable. To this

end, we introduce a scalar s, that keeps the data likelihood at the lowest, inducing an invariance to scaling. We calculate its value by solving $s^* = \min_s ||Es\mu_x(z^t) - y||_2^2$, where we separate the s term from the extended encoding and use the mean of the decoder as the image. Then, we take the derivative of the expression with respect to s and set it to zero to obtain the minimizing s value, which is given analytically as $s^* = \frac{Re\{\mu_x(z^t)^H E^H y\}}{\mu_x(z^t)^H E^H E \mu_x(z^t)}$. We do this estimation separately for each z^t sample at each step.

The complex conjugate of the extended encoding operation operation is given as $E^H = s^H P^H \varphi^H B^H \tilde{E}^H$, where we implement P^H as cropping if P is a padding operation and vice versa, φ^H is multiplication with the complex conjugate of the phase, $B^H = B$ since the bias field is real and $s^H = s$, again since the scale factor is real.

F Measuring the quality of samples

As we do not have access to the ground truth posterior distribution of images given the k-space data, we resort to using indirect measures and characterize two aspects of the samples. Firstly, the samples have to be in agreement with the measured k-space data. Secondly, the samples have to have a high diversity to the extent allowed by the measured data and the noise in k-space. Notice that there is a trade-off between these two aspects, that is, the measured data constrains the sample diversity and a high sample diversity requires moving away from measured data, increasing the error in k-space.

We use two metrics to quantify the first aspect in Section F.1. First is the error in k-space between the samples and the given data for an image. Secondly, though this also considers the parts of the k-space that are not measured, we look at the RMSE between the samples and the original image. We then introduce a pairwise RMSE metric to quantify the sample diversity and present the results in Section F.5.

F.1 Distribution of voxelwise error in the measured parts of k-space and NMSE, pSNR and RMSE in the image space

Here we show the k-space error histograms in Figure 1 for a test slice at R=5. We calculate these as follows: we take 50 image samples $\{x_s\}_{s=1}^{50}$ from each method and apply the undersampled Fourier transform to transform each of them to k-space and take the measured voxels. Then we calculate the voxelwise difference between these and the measured data for all measured k-space voxels for all the samples together. The histogram then shows the distribution of the error for all these k-space voxels from all 50 samples. As this difference is complex, we show two histograms seperately for the real and imaginary parts and also for the magnitude values. We can also look at the image-wise k-space absolute error as

$$\text{absolute error}_s = \frac{1}{\text{no of meas. voxels}} \sum_{\text{all meas. voxels}} |Ex_{FS} - Ex_s|, \tag{31}$$

for a sample image x_s and the fully sampled image x_{FS}. The $|.|$ denotes the magnitude of the complex error value for a pixel and the average is taken over all measured k-space voxels. When calculated for all 50 samples, the mean (std) values for this slice are given as 0.0309 (0.0003), 0.0373 (0.0012) and 0.0381 (0.00047), for the l-MALA, cWGAN and local sampling methods, respectively.

To show how this generalizes, we do a similar analysis using slices from 9 test subjects. We undersample the slices with different patterns for each subject at R=5. Again, for each test subject we generate 50 samples for the three methods each. We then calculate the absolute errors and report the mean and standard deviation values for these in the main text. We also calculate the root mean squared error (RMSE) between the 50 samples and the fully sampled image. Though achieving a low RMSE is not the main purpose of any of the methods, we present these results as they still provide some insight into the performance of the methods. To calculate the RMSE we

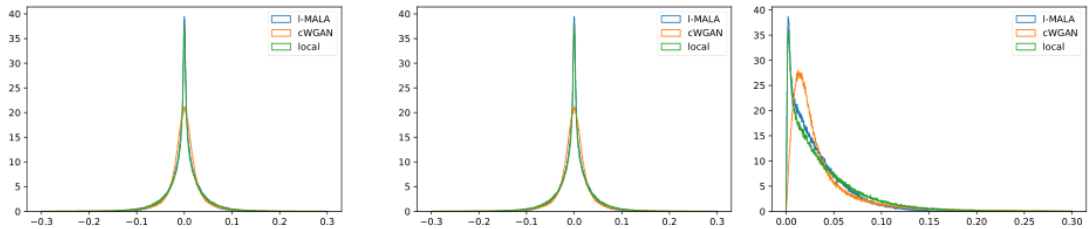


Figure 1: Histograms of the voxelwise error in the measured voxels in the k-space for three different methods for a subject at R=5. As the error is complex, the real and imaginary parts as well as the magnitude of the error values are shown seperately.

use the formula given in [7] and use a mask to disregard the background. For in-house images we also perform a bias field correction for both the sample and the fully sampled image before the calculation. Furthermore, again, though sampling is inherently different than only reconstruction, for comparison purposes we present the absolute error and RMSE values of two reconstruction methods, namely the DDP [7] and the variational network (VarNet) [8]. As the data consistency projection in the DDP method inputs the measured data into the reconstructed k-space, it has a zero absolute error. For both metrics the reconstruction methods yield better performance than sampling methods, which is expected as these are designed to yield the best performance rather than characterize the solution space comprehensively.

We took the definitions of the normalized mean squared error (NMSE) and peak signal-to-noise ratio (pSNR) from the fastMRI repository [9].

F.2 Implementation of the variational network (Varnet) and cWGAN

For the variational network reconstruction [8] we used the implementation given in `https://github.com/visva89/VarNetRecon`. We used a batch size of 2, 48 filters with kernel size 11 at each layer, 10 unfolding layers, filter response as 3.5, 31 knots and cubic interpolation for modeling the activation functions, the L_2 loss at the output and otherwise the default parameters. During training we generated undersampled/fully sampled image pairs from the same training set as for the VAE with different patterns at each iteration and fed these into the network. Furthermore the network required an image size of powers of two, for which we padded the images to a size 256x320. We trained for 200000 iterations for R=3,4,5 and 114000 iterations for R=2 with a learning rate of 0.001. At test time we padded the test images (originally 252x308) as well as their undersampling patterns and after reconstruction cropped back to the original size for comparing the performance with different methods.

We trained the cWGAN method [10] as given in the code shared by the authors. We modified it minimally to work with MR images and trained for 150000/600000/800000/800000 iterations for R=2,3,4,5, respectively, with the decay ratio for the noisy linear cosine decay as 2000000 but otherwise with the default settings in the code provided by the authors and the augmentation used for the VAE.

subject	RMSE (%)				
	l-MALA	cWGAN	Local	VarNet	DDP
#1	8.5 (8.3, 0.11)	11.3 (10.7, 0.35)	11.7 (11.4, 0.13)	8.7	8.2
#2	9.8 (9.7, 0.06)	13.4 (12.8, 0.40)	12.9 (12.6, 0.11)	9.7	9.2
#3	10.9 (10.9, 0.01)	15.6 (14.3, 0.76)	14.5 (14.1, 0.15)	10.7	9.7
#4	7.8 (7.8, 0.03)	11.0 (10.4, 0.29)	10.8 (10.6, 0.10)	8.2	7.4
#5	7.0 (6.9, 0.03)	11.3 (10.6, 0.46)	10.4 (10.1, 0.12)	8.4	6.5
#6	6.8 (6.8, 0.04)	9.9 (9.2, 0.35)	10.2 (10.0, 0.14)	6.4	5.8
#7	8.0 (8.0, 0.02)	12.5 (11.3, 0.55)	11.0 (10.7, 0.12)	8.3	7.7
#8	9.1 (9.0, 0.07)	12.8 (12.0, 0.41)	12.1 (11.9, 0.10)	9.8	9.0
#9	6.7 (6.6, 0.02)	10.2 (9.5, 0.42)	9.7 (9.5, 0.11)	7.3	6.1
mean (std)	8.30 (1.35)	12.0 (1.75)	11.47 (1.42)	8.61 (1.24)	7.74 (1.32)

Table 1: RMSE values in percentage for 9 HCP test subjects at R=5. Values shown in format: mean (min, std) for the sampling methods and the single value for the VarNet and DDP reconstruction methods. The last line shows the mean (std) of the 9 subjects.

F.3 Samples and segmentations at different undersampling ratios and k-space noise levels

Here we present two figures demonstrating how the changing undersampling ratio and k-space noise levels change the samples. Similarly we present two figures which show how the segmentations change under the same conditions.

In Fig. 2, we show how the statistics from the samples change with changing undersampling ratios. Firstly, we show histograms from three pixels indicated on the FS image for R=2, 3, 4 and 5, from which one can observe that the pixel histograms become wider with increasing R, indicating higher uncertainty. This increase is also reflected in the std maps, which show an increase in std values for increasing R. This result shows that the proposed model is able to capture increasing ambiguity due to higher undersampling ratio.

Next we present results in Fig. 3 to show the methods sensitivity to the noise in the k-space. The quality of the MAP image degrades due to the high noise. This is reflected less in the mean maps, however the standard deviation values increase. This is how the model should behave since the added noise increases the values in Σ_{ns}, which then allows samples to move farther away from the measured data and show higher diversity. This is also reflected in the histograms of three pixel's intensities, which are indicated in the top std map, as the distributions become wider with increasing noise.

In Figures 4 and 5 we show the segmentation results for the same settings. In the final row of each figure, we show the pixels where the binarized standard deviation map, i.e. 1 if there is variation in that pixel among samples, 0 if there is no variation in that pixel among samples. We do this for visualisation purposes and as the segmentation maps are binary, their standard deviation values are not informative in any case. These also confirm the observations from Figures 3 and 2, that the samples behave as expected from the theory.

F.4 An in-house measured image at different undersampling ratios

Here we present an image at different undersampling ratios in Figure 6.

F.5 Comparison of sample diversity using pairwise RMSE at different noise rates and undersampling ratios

In this section we introduce the pairwise RMSE metric, which we use to measure the sample diversity at different noise levels and undersampling ratios for different methods [11] (We use RMSE instead of the structural similarity index measure as in the reference as the first reflects

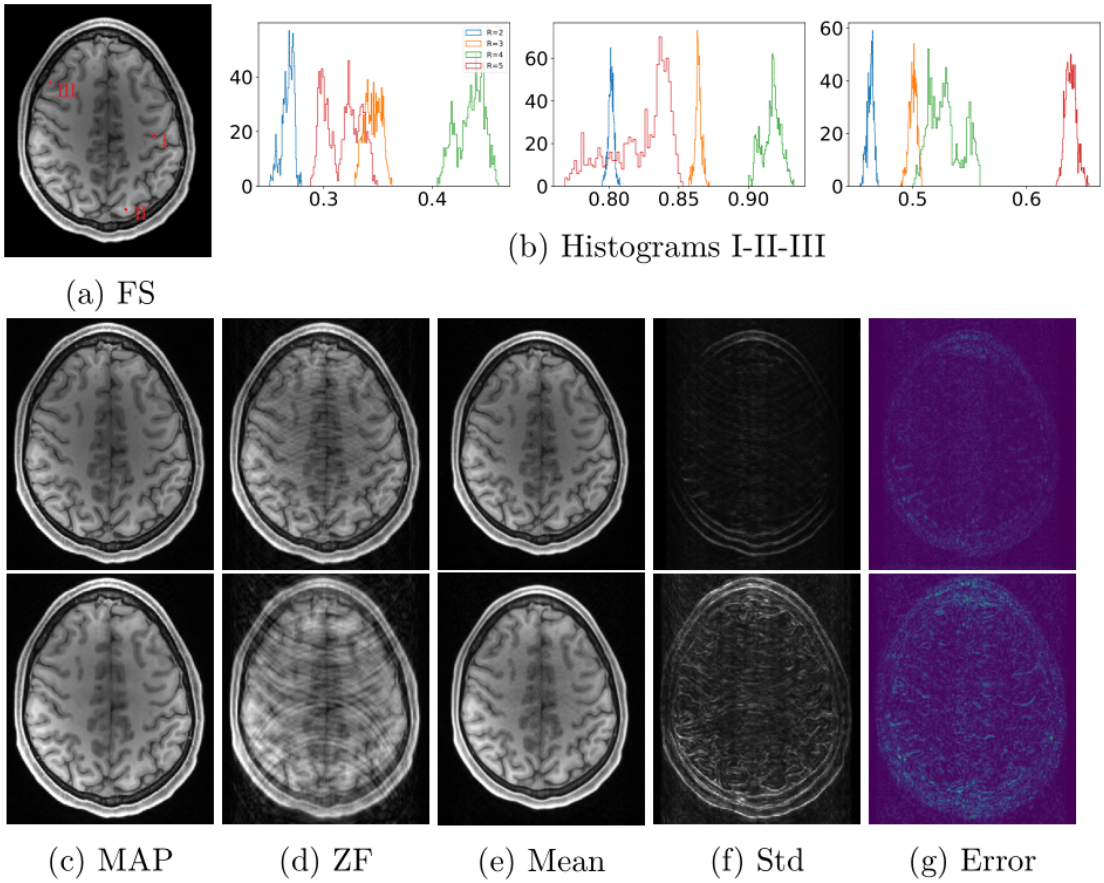


Figure 2: Results for changing undersampling ratios. First row shows the fully sampled image (FS) and histograms of pixels values in all samples for the pixels indicated on the FS image as I, II and III, respectively. Note the different bin positions for the histograms. Rows two and three show results for R=2 and R=4, respectively. Each row shows the MAP estimation, the zero filled image (ZF), the pixelwise mean and standard deviation maps and the absolute error map between the mean and the FS image (clipped to (0,0.3)).

structural changes better). For this metric we take 1000 pairs of random samples from a method at a noise level or undersampling ratio for a subject and calculate the RMSE between these pairs. This yields 1000 RMSE values, of which we then take the mean to obtain the pairwise RMSE value for this subject and for the method at this noise level or undersampling ratio.

The aim here, again, to verify the behavior of samples with changing setting, i.e. the "sanity check" experiment. The main idea is that if the k-space noise level is higher, this means that images that are possible solutions to the inverse problem can be farther away from the measured k-space data, which allows these images to be more different than each other, i.e. more diverse. Similarly, if the undersampling ratio is higher, there is less measured data that determine the solutions to the inverse problem, which again allows solution images to be more different from each other leading to higher diversity. As this is a basic relationship between the measurement setting and diversity of solutions, any sampling algorithm should also adhere to this relationship and hence we use this as a "sanity check".

To this end we added noise on the k-space of an HCP image at R=5 similar to the experiment in the main text. We also experimented with varying the undersampling ratio similar the experiment in the main text at the base noise level. We show these in the main text. One can see that for

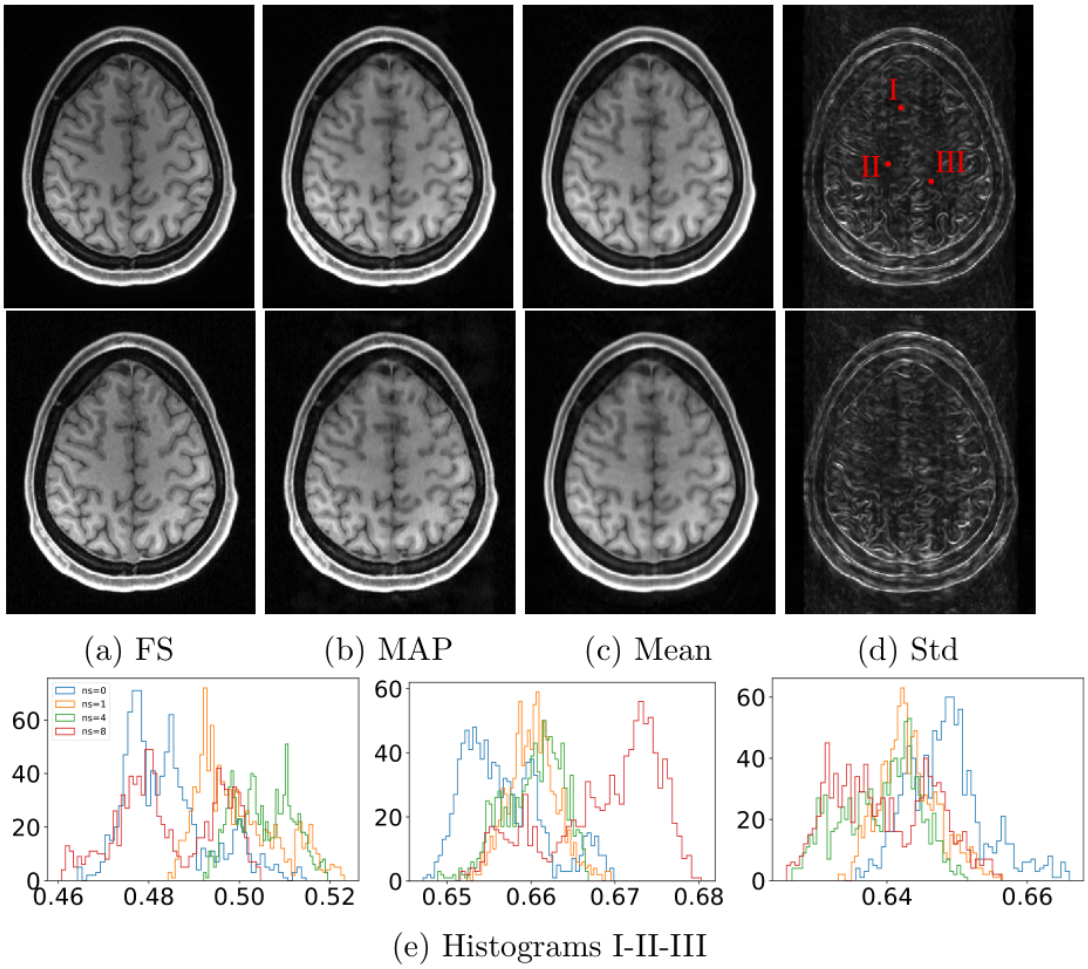


Figure 3: Results for changing the noise in k-space at R=5. First row shows the results with the basis HCP k-space noise. Second row shows the results with noise added on the k-space with 8 times the original noise standard deviation. Third row shows histograms of values of the pixels indicated on the std map (with added noise 1, 4 and 8 times of the basis noise). Note that the fully sampled (FS) image also changes due to the added noise.

both l-MALA and cWGAN the pairwise RMSE, i.e. sample diversity is increasing with increasing noise levels in the k-space, as expected. The same trend is not observed for the local sampling method, meaning that the method does not fulfil the expectation. A similar conclusion can be made for the results of changing the undersampling ratio as seen in the main text.

G Error metrics for the in-house measured images

In Tables 2, 3, 4 and 5 we present the used metrics for 6 subjects from the in-house measured dataset at R=2 to 5, respectively. The absolute error is shown as the average of all coils. The DDP absolute error is not zero as the final inverse-forward encoding operations change the k-space in case of multiple coils. Multiple factors contribute to higher error values compared to the HCP images, such as the domain shift between the HCP training images and in-house test images, errors in coil sensitivity estimations from ESPIRiT, or higher errors in the used phase from the MAP estimate. Another observation is that the RMSE values for the l-MALA samples are higher than

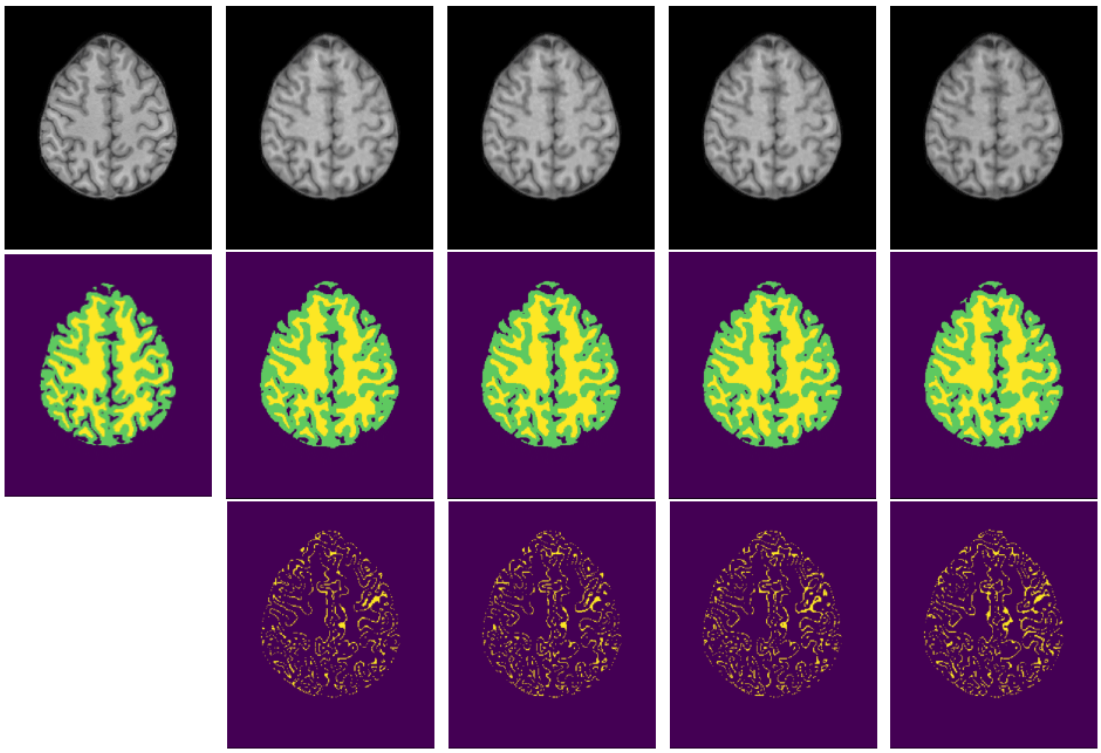
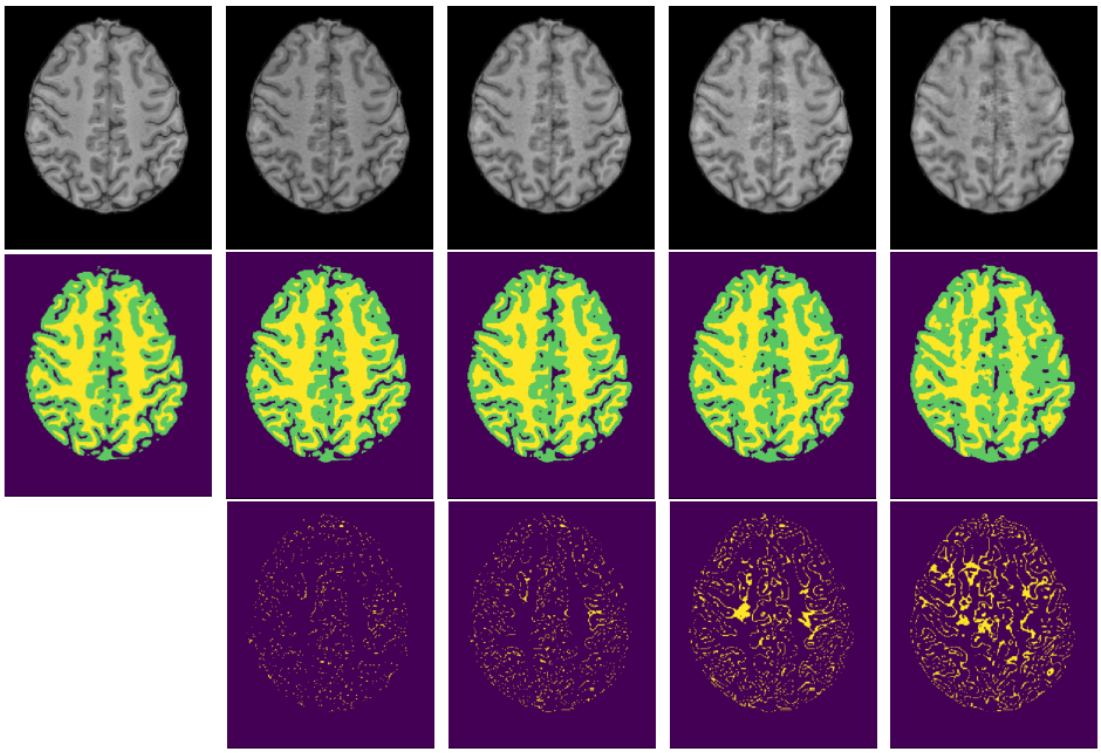


Figure 4: Segmentation results for changing the noise in k-space at R=5. Leftmost column show the fully-sampled image and its segmentation. First row shows a random sample at increasing added k-space noise levels. Second row shows the segmentation of the corresponding sample in the above row. Last row shows the pixels where there is variation in the segmentations for all samples.

for the DDP reconstructions. This is mostly because the DDP has a data projection inputting the measured data to the reconstructed k-space, whereas sampling allows for some distance to the measured data to account for the noise.

H Convergence of the chain

Here we take a random HCP image and show the mean intensity in the brain in this image (i.e. after masking) throughout the MCMC iterations. The plot in Figure 7 shows that the chain converges to a range of values close to its initialization. A long burn-in period seems to be avoided by initializing with the MAP estimate. Theoretically, with infinitely many steps the MCMC chain has to discover all solutions. It is, however, possible that for finite chains the initialization introduces some bias. Though it is difficult to show here that this is not the case, the intensities seem to vary throughout the chain, providing empirical evidence that the chain can explore freely.



(a) R=2 (b) R=3 (c) R=4 (d) R=5

Figure 5: Segmentation results for changing the undersampling ratios. Leftmost column show the fully-sampled image and its segmentation. First row shows a random sample at varying undersampling ratios. Second row shows the segmentation of the corresponding sample in the above row. Last row shows the pixels where there is variation in the segmentations for all samples.

I Comparing the samples $x^t \sim p(x|z^t)$ vs $x^t \sim p(x|y, z^t)$

Here we compare the sample quality for the proposed l-MALA with the samples as direct outputs of the decoder. As seen in Figure 8, the proposed sampling approach improves the sample quality drastically.

J More comparisons on the HCP data

Here we provide more figures for comparing with the alternative methods for different images and undersampling ratios in Figures 9-15.

References

[1] K. Tóthová, S. Parisot, M. C. H. Lee, E. Puyol-Antón, L. M. Koch, A. P. King, E. Konukoglu, and M. Pollefeys, "Uncertainty quantification in cnn-based surface prediction using shape priors," in *Shape in Medical Imaging*. Springer International Publishing, 2018, pp. 300–310.

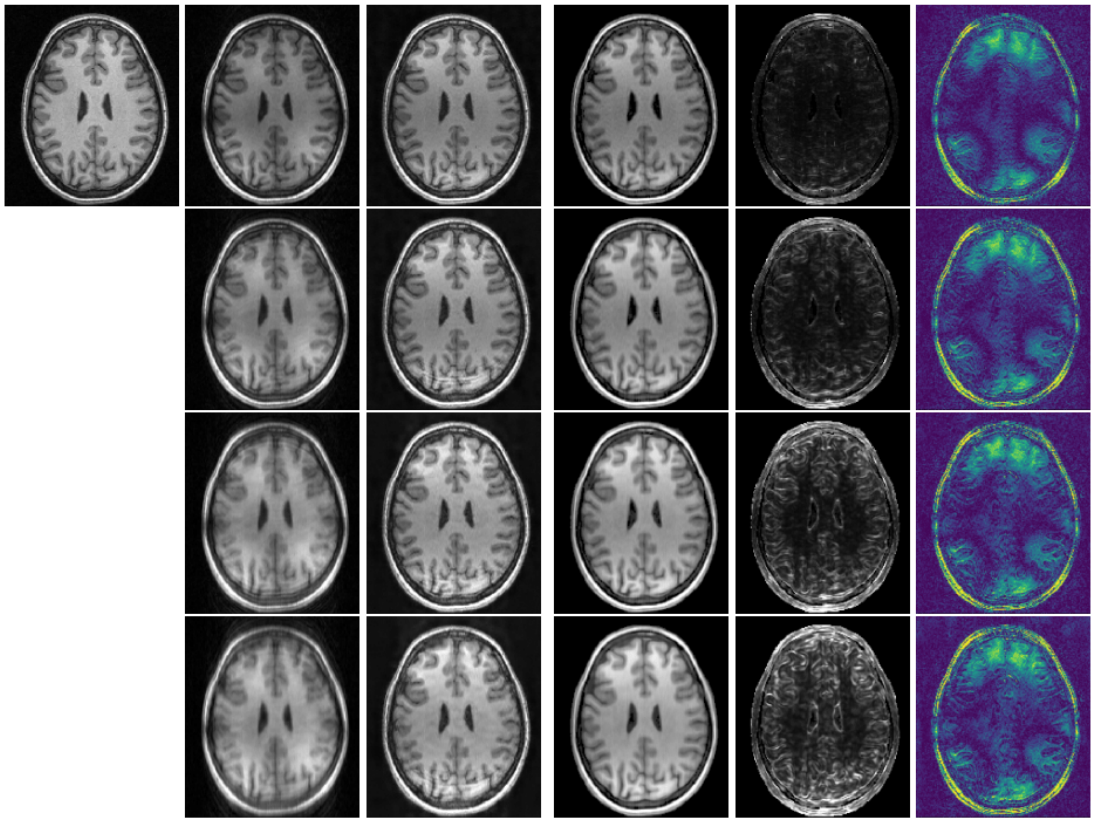


Figure 6: Sampling results for the in-house measured images for changing undersampling ratios at R=2,3,4 and 5 from top down, respectively.

[2] D. P. Kingma and M. Welling, "Auto-encoding variational bayes." *CoRR*, vol. abs/1312.6114, 2013.

[3] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 32, no. 2. Bejing, China: PMLR, 22–24 Jun 2014, pp. 1278–1286.

[4] K. P. Pruessmann, M. Weiger, P. Börnert, and P. Boesiger, "Advances in sensitivity encoding with arbitrary k-space trajectories," *Magnetic Resonance in Medicine*, vol. 46, no. 4, pp. 638–651, 2001.

[5] M. Gaillochet, K. Tezcan, and E. Konukoglu, "Joint reconstruction and bias field correction for undersampled mr imaging," *MICCAI*, 2020.

subject	absolute k-space error (x10^3)			RMSE (%)			NMSE (x10^3)			pSNR		
	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet
#1	61.16 (61.09, 0.05)	29.24	55.34	9.47 (9.40, 0.04)	6.20	6.01	8.98 (8.84, 0.08)	3.85	3.61	34.81 (34.87, 0.04)	38.48	38.76
#2	60.14 (60.06, 0.06)	28.21	52.60	9.83 (9.63, 0.13)	6.78	7.16	9.66 (9.28, 0.25)	4.60	5.12	35.48 (35.65, 0.11)	38.70	38.24
#3	68.13 (68.07, 0.03)	33.85	56.45	9.06 (8.99, 0.03)	6.84	6.10	8.21 (8.08, 0.05)	4.67	3.71	35.00 (35.07, 0.03)	37.45	38.44
#4	67.46 (67.40, 0.03)	31.08	54.79	11.49 (11.44, 0.03)	8.21	7.61	13.19 (13.08, 0.07)	6.66	5.78	35.97 (36.01, 0.02)	38.94	39.56
#5	72.14 (72.09, 0.03)	30.37	56.70	10.96 (10.90, 0.06)	7.29	7.07	12.01 (11.88, 0.14)	5.30	4.98	34.89 (34.94, 0.05)	38.45	38.71
#6	56.06 (56.01, 0.02)	29.20	51.98	9.76 (9.70, 0.03)	6.89	6.25	9.52 (9.41, 0.06)	4.75	3.90	35.33 (35.38, 0.03)	38.35	39.21
mean (std)	64.18 (5.49)	30.32 (1.82)	54.64 (1.79)	10.09 (0.85)	7.04 (0.61)	6.70 (0.61)	10.26 (1.76)	4.97 (0.87)	4.52 (0.82)	35.25 (0.41)	38.39 (0.46)	38.82 (0.45)

Table 2: Different metrics for the 6 in-house measured subjects at R=2.

subject	absolute k-space error (x10^3)			RMSE (%)			NMSE (x10^3)			pSNR		
	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet
#1	72.66 (72.52, 0.14)	30.78	65.95	11.04 (10.82, 0.12)	8.35	8.14	12.20 (11.72, 0.26)	6.97	6.61	33.48 (33.65, 0.09)	35.91	36.13
#2	72.20 (72.09, 0.09)	30.58	62.63	13.50 (13.40, 0.15)	10.25	11.28	18.23 (17.95, 0.41)	10.53	12.72	32.72 (32.79, 0.09)	35.10	34.28
#3	80.91 (80.72, 0.12)	35.46	65.59	10.92 (10.69, 0.15)	8.88	8.55	11.92 (11.42, 0.32)	7.88	7.30	33.38 (33.57, 0.12)	35.18	35.51
#4	84.91 (84.76, 0.07)	35.76	68.25	14.42 (14.14, 0.14)	11.54	11.73	20.81 (20.01, 0.41)	13.29	13.77	33.99 (34.16, 0.09)	35.94	35.78
#5	84.96 (84.74, 0.12)	32.15	66.58	13.47 (13.38, 0.07)	9.73	10.26	18.15 (17.90, 0.20)	9.45	10.52	33.09 (33.15, 0.05)	35.93	35.46
#6	64.34 (64.26, 0.05)	30.52	60.06	11.49 (11.37, 0.06)	8.87	8.53	13.21 (12.93, 0.13)	7.85	7.27	33.91 (34.00, 0.04)	36.17	36.50
mean (std)	76.66 (7.56)	32.54 (2.24)	64.84 (2.71)	12.48 (1.38)	9.60 (1.07)	9.75 (1.42)	15.75 (3.46)	9.33 (2.12)	9.70 (2.82)	33.43 (0.45)	35.70 (0.41)	35.61 (0.69)

Table 3: Different metrics for the 6 in-house measured subjects at R=3.

subject	absolute k-space error (x10^3)			RMSE (%)			NMSE (x10^3)			pSNR		
	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet
#1	80.38 (80.21, 0.13)	31.06	72.78	11.72 (11.54, 0.13)	9.02	11.38	13.73 (13.31, 0.30)	8.13	12.95	32.96 (33.10, 0.09)	35.24	33.22
#2	82.53 (82.43, 0.07)	33.03	71.21	14.02 (13.79, 0.18)	11.35	14.80	19.66 (19.02, 0.50)	12.92	21.91	32.39 (32.53, 0.11)	34.22	31.93
#3	87.19 (87.06, 0.13)	36.20	69.90	12.52 (12.36, 0.08)	10.92	13.42	15.68 (15.28, 0.20)	11.93	18.02	32.19 (32.30, 0.06)	33.37	31.58
#4	98.13 (97.94, 0.10)	37.39	77.40	15.26 (15.04, 0.11)	13.02	14.54	23.28 (22.63, 0.33)	16.96	21.13	33.50 (33.62, 0.06)	34.88	33.92
#5	94.31 (94.13, 0.16)	32.86	73.28	16.58 (16.40, 0.12)	14.53	17.00	27.50 (26.91, 0.41)	21.21	28.91	31.29 (31.38, 0.06)	32.42	31.07
#6	69.98 (69.87, 0.04)	30.94	65.90	12.72 (12.58, 0.08)	10.42	12.19	16.18 (15.81, 0.19)	10.87	14.86	33.03 (33.13, 0.05)	34.76	33.40
mean (std)	85.42 (9.27)	33.58 (2.43)	71.74 (3.50)	13.80 (1.69)	11.54 (1.79)	13.89 (1.84)	19.34 (4.79)	13.67 (4.28)	19.63 (5.22)	32.56 (0.72)	34.15 (0.97)	32.52 (1.04)

Table 4: Different metrics for the 6 in-house measured subjects at R=4.

subject	absolute k-space error (x10^3)			RMSE (%)			NMSE (x10^3)			pSNR		
	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet	l-MALA	DDP	VarNet
#1	85.07 (84.97, 0.07)	31.60	78.71	14.91 (14.72, 0.18)	14.33	16.46	22.24 (21.67, 0.54)	20.63	27.15	30.87 (30.98, 0.10)	31.19	30.00
#2	89.32 (89.20, 0.10)	34.01	77.06	16.42 (16.16, 0.15)	13.99	16.05	26.97 (26.11, 0.49)	19.67	25.79	31.02 (31.16, 0.08)	32.39	31.22
#3	97.16 (97.05, 0.05)	36.41	76.11	13.59 (13.41, 0.08)	12.11	13.60	18.47 (17.99, 0.22)	14.69	18.50	31.48 (31.60, 0.05)	32.48	31.47
#4	109.11 (109.04, 0.03)	40.62	86.56	19.08 (18.85, 0.16)	17.36	18.60	36.40 (35.52, 0.60)	30.32	34.70	31.56 (31.66, 0.07)	32.35	31.76
#5	105.48 (105.18, 0.22)	33.63	80.25	16.38 (16.03, 0.25)	13.19	15.26	26.84 (25.69, 0.83)	17.47	23.31	31.40 (31.59, 0.13)	33.26	32.01
#6	76.34 (76.18, 0.09)	30.92	71.17	13.36 (13.28, 0.06)	11.67	12.33	17.86 (17.63, 0.17)	13.65	15.21	32.60 (32.66, 0.04)	33.77	33.30
mean (std)	93.75 (11.43)	34.53 (3.25)	78.31 (4.64)	15.62 (1.96)	13.78 (1.86)	15.38 (2.02)	24.80 (6.32)	19.40 (5.47)	24.11 (6.26)	31.49 (0.56)	32.57 (0.81)	31.63 (0.98)

Table 5: Different metrics for the 6 in-house measured subjects at R=5.

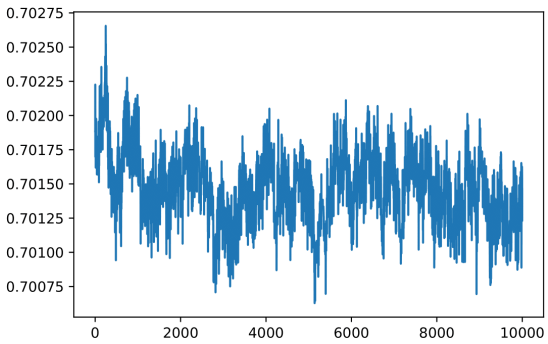


Figure 7: Change of mean signal intensity in the brain throughout MCMC iterations for a random HCP image showing convergence of the chain.

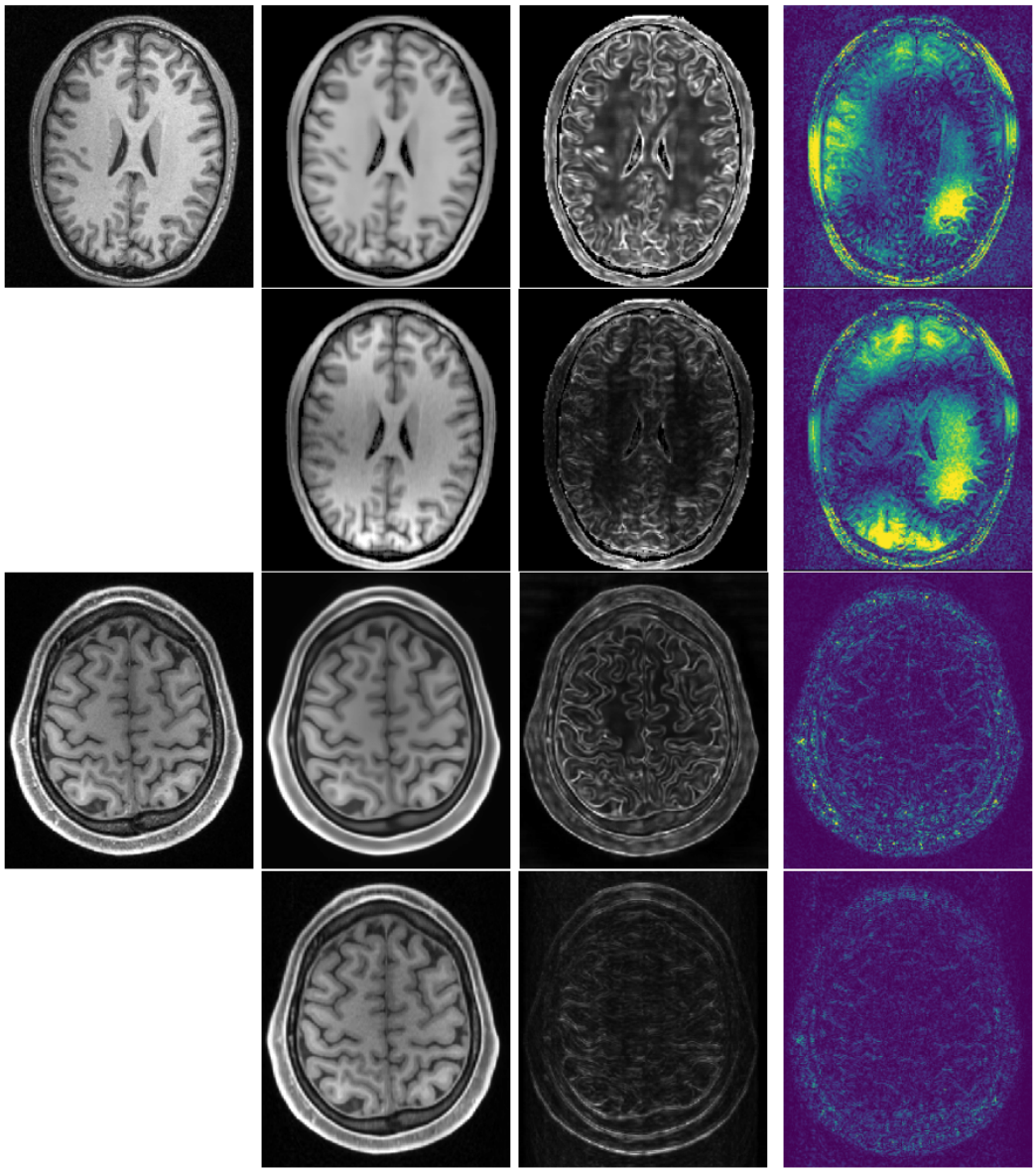


Figure 8: Comparison between the proposed sampling versus the simpler alternative of taking the decoder output at R=3. The upper block shows an in-house measured image, the lower block shows an HCP image. The leftmost column presents the full-sampled images. In each block the upper row represents the decoder output ($x \sim p(x|z^t)$) and the lower row represents the sample from $x \sim p(x|y, z^t)$.

[6] P. G. Sled J.G., "Understanding intensity non-uniformity in mri." *MICCAI*, 1998.

[7] K. C. Tezcan, C. F. Baumgartner, R. Luechinger, K. P. Pruessmann, and E. Konukoglu, "Mr image reconstruction using deep density priors," *IEEE Transactions on Medical Imaging*, vol. 38, no. 7, 2019.

[8] K. Hammernik, T. Klatzer, E. Kobler, M. P. Recht, D. K. Sodickson, T. Pock, and F. Knoll, "Learning a variational network for reconstruction of accelerated mri data," *Magnetic Resonance in Medicine*, pp. n/a–n/a, 2017.

[9] M. J. Muckley, B. Riemenschneider, A. Radmanesh, S. Kim, G. Jeong, J. Ko, Y. Jun, H. Shin, D. Hwang, M. Mostapha, S. Arberet, D. Nickel, Z. Ramzi, P. Ciuciu, J.-L. Starck, J. Teuwen, D. Karkalousos, C. Zhang, A. Sriram, Z. Huang, N. Yakubova, Y. W. Lui, and F. Knoll, "Results of the 2020 fastmri challenge for machine learning mr image reconstruction," *IEEE Transactions on Medical Imaging*, vol. 40, no. 9, pp. 2306–2317, 2021.

[10] J. Adler and O. Öktem, "Deep bayesian inversion," *arXiv:1811.05910*, 2018.

[11] A. Volokitin, E. Erdil, N. Karani, K. C. Tezcan, X. Chen, L. V. Gool, and E. Konukoglu, "Modelling the distribution of 3d brain mri using a 2d slice vae," *MICCAI*, 2020.

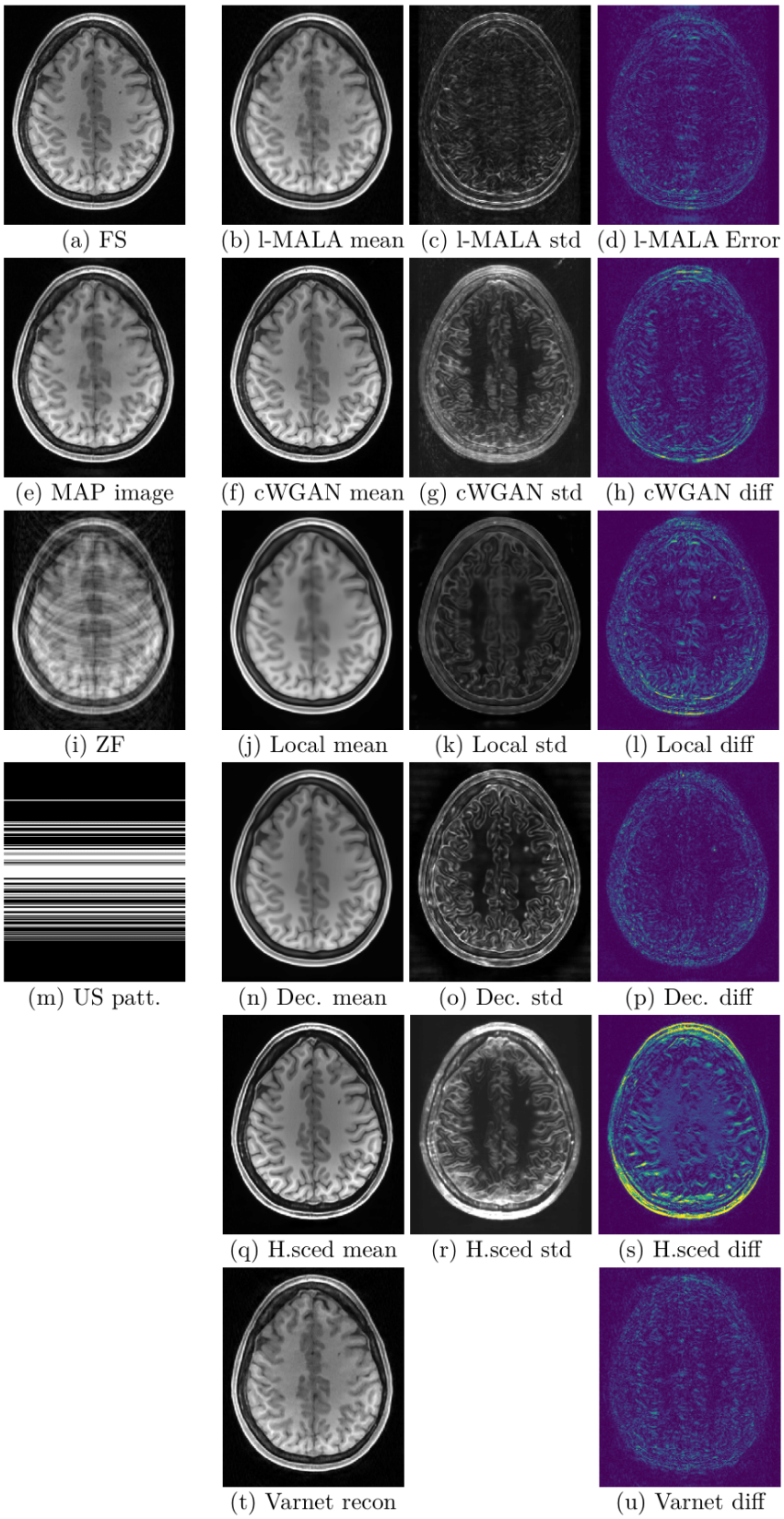


Figure 9: Comparisons at R=4. Figure description same as in main text.

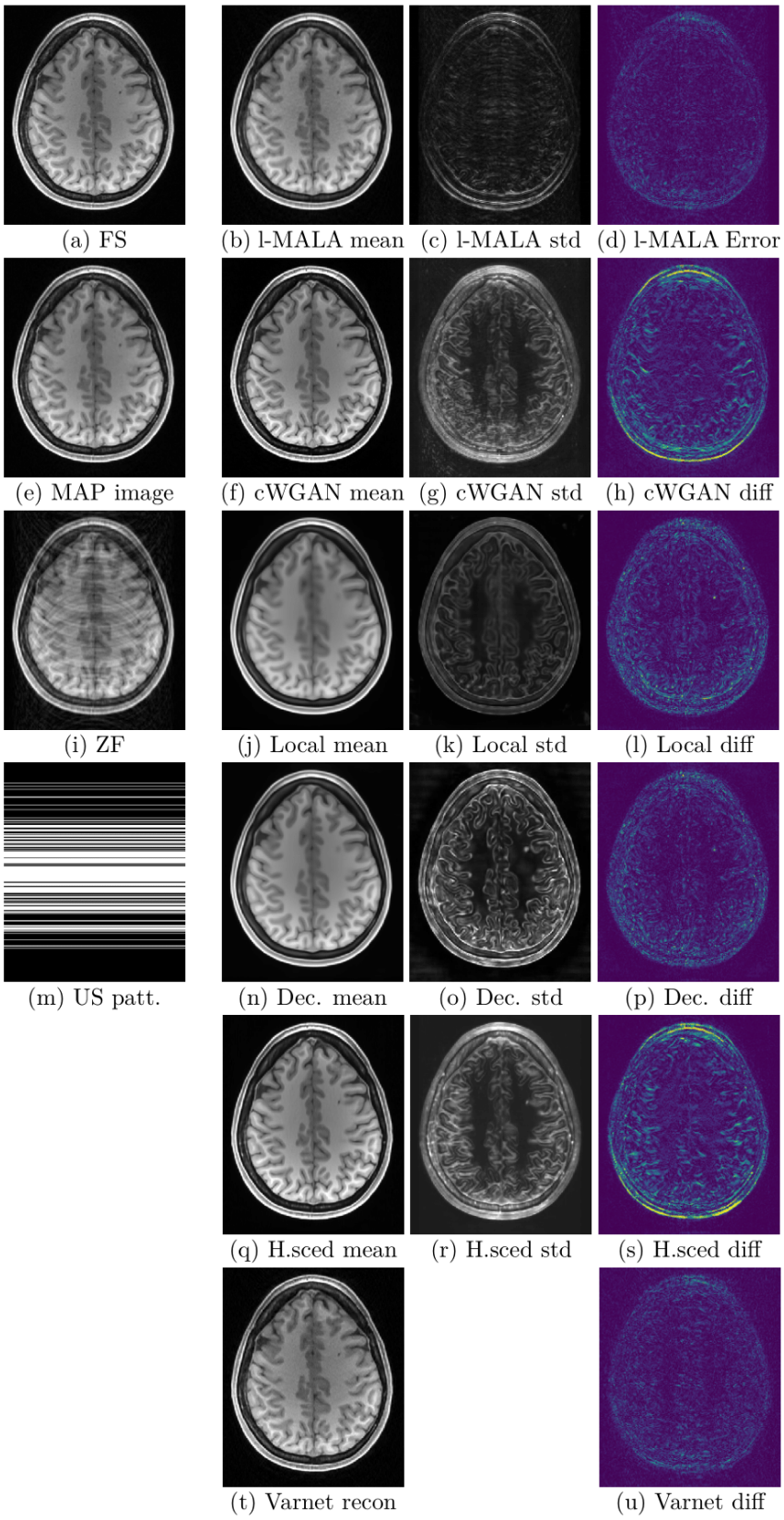


Figure 10: Comparisons at R=3. Figure description same as in main text.

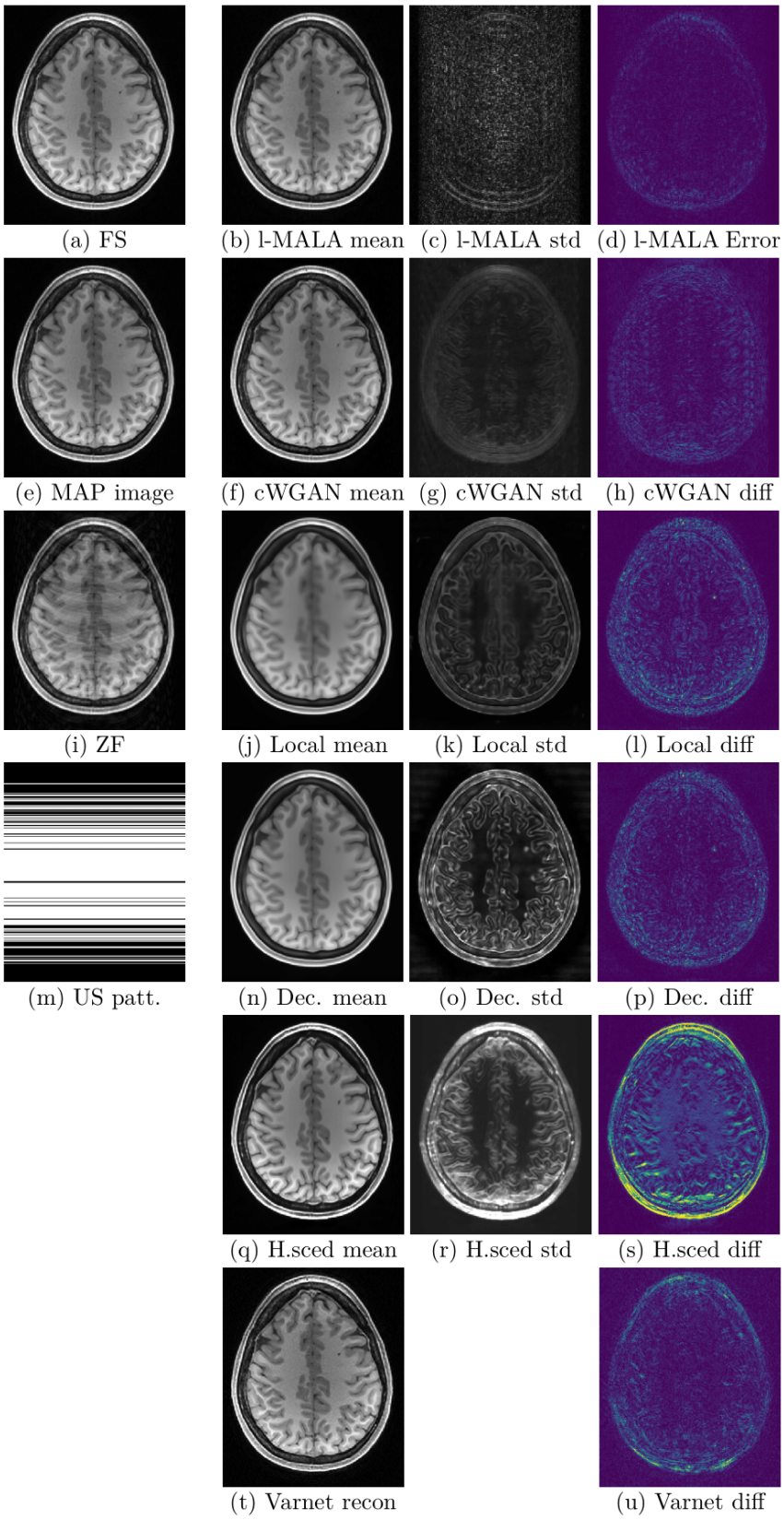


Figure 11: Comparisons at R=2. Figure description same as in main text.

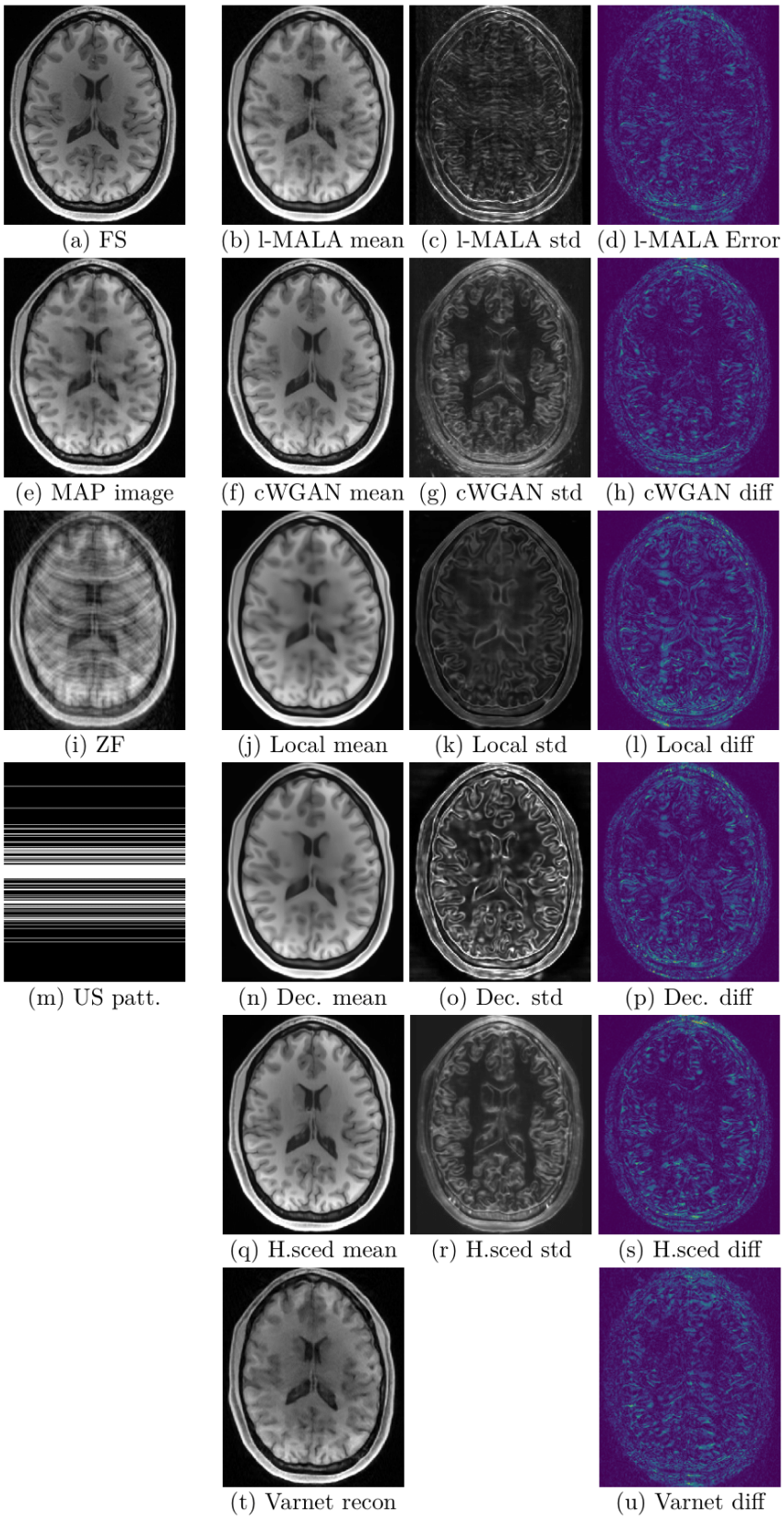


Figure 12: Comparisons at R=5. Figure description same as in main text.

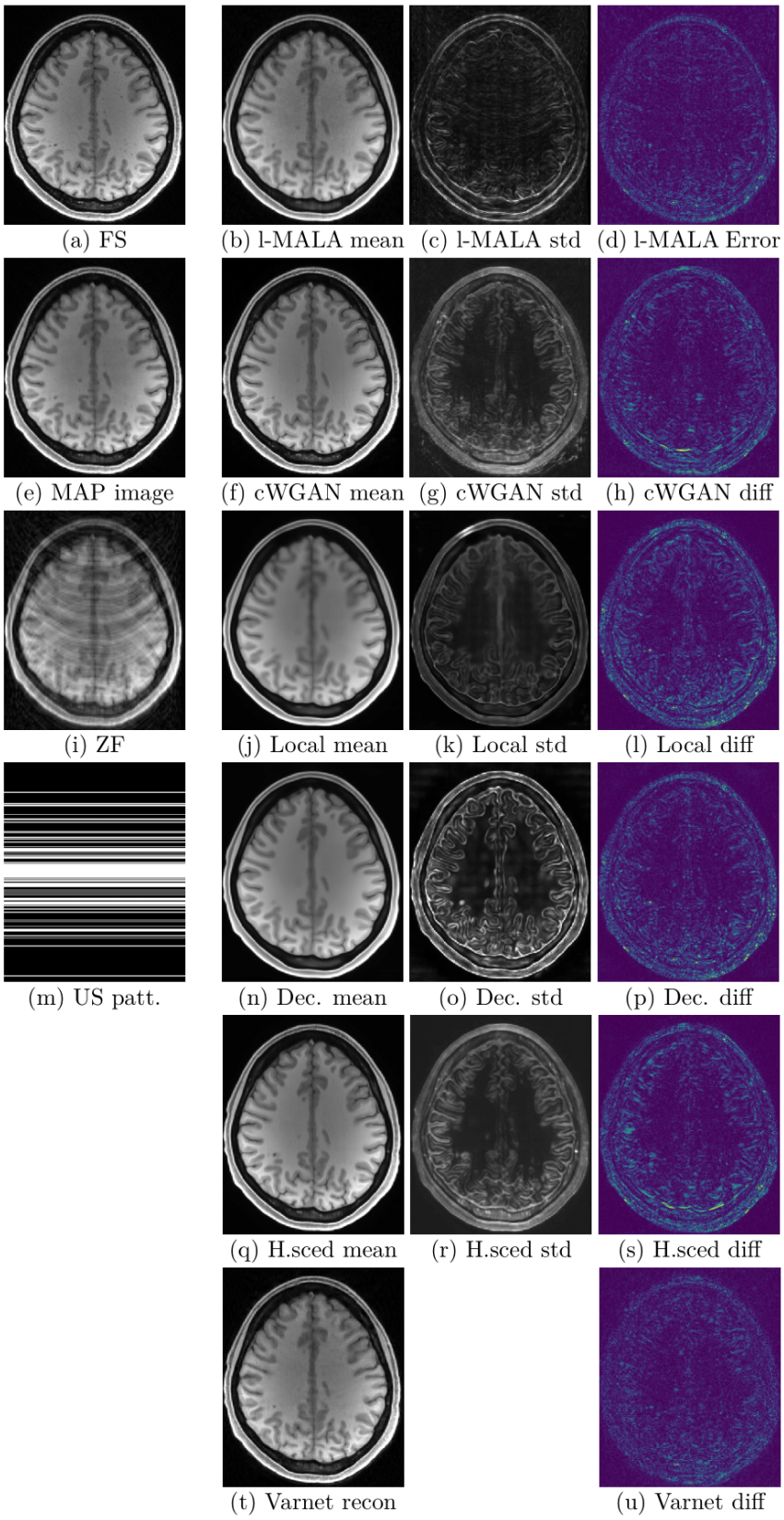


Figure 13: Comparisons at R=4. Figure description same as in main text.

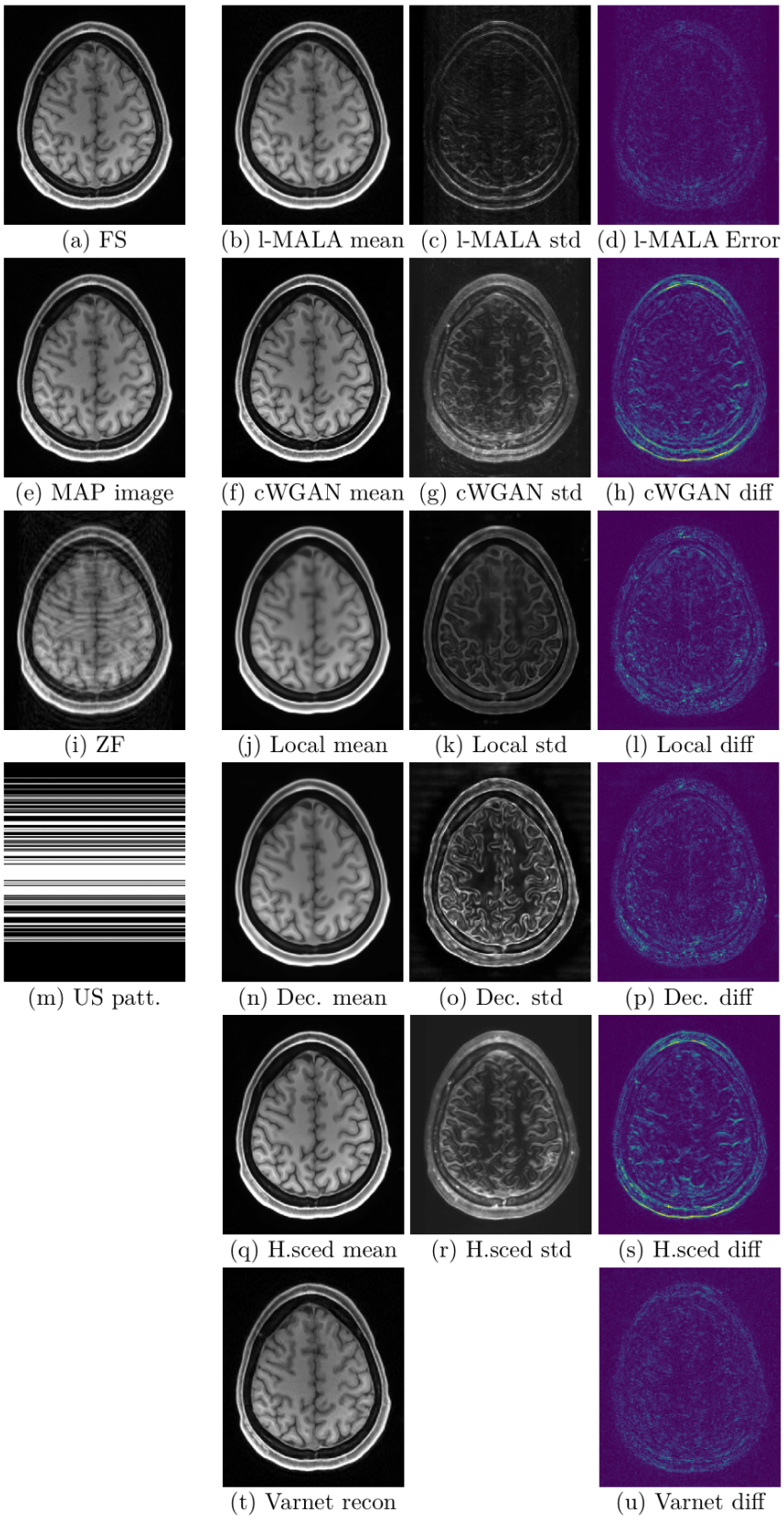


Figure 14: Comparisons at R=3. Figure description same as in main text.

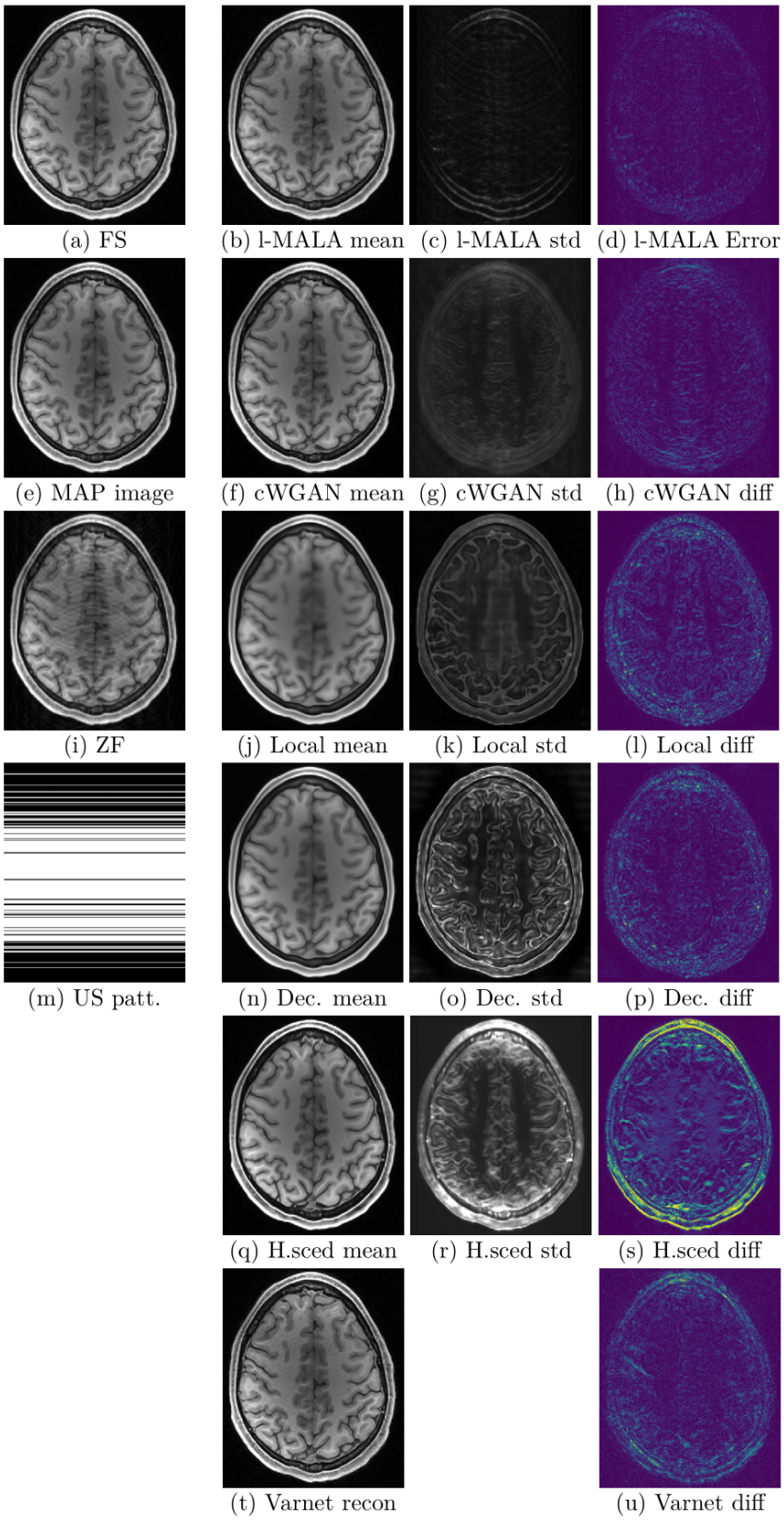


Figure 15: Comparisons at R=2. Figure description same as in main text.