Review Article

Received: July 3, 2020

Revised: September 28, 2020

Accepted: November 10, 2020

Correspondence to:

Qiegen Liu, Ph.D.

Department of Electronic
Information Engineering,
Nanchang University, Nanchang,
Jiangxi Province, 330031, China.

Tel. +86-13732908412

Fax. +86-791-88304443

E-mail: liuqiegen@hotmail.com

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/4.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Copyright © 2020 Korean Society
of Magnetic Resonance in
Medicine (KSMRM)

A Comparative Study of Unsupervised Deep Learning Methods for MRI Reconstruction

Zhuonan He¹, Cong Quan¹, Siyuan Wang¹, Yuanzheng Zhu¹, Minghui Zhang¹, Yanjie Zhu², Qiegen Liu¹

¹Department of Electronic Information Engineering, Nanchang University, Nanchang, China

²Paul C. Lauterbur Research Center for Biomedical Imaging, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

Recently, unsupervised deep learning methods have shown great potential in image processing. Compared with a large-amount demand for paired training data of supervised methods with a specific task, unsupervised methods can learn a universal and explicit prior information on data distribution and integrate it into the reconstruction process. Therefore, it can be used in various image reconstruction environments without showing degraded performance. The importance of unsupervised learning in MRI reconstruction appears to be growing. Nevertheless, the establishment of prior formulation in unsupervised deep learning varies a lot depending on mathematical approximation and network architectures. In this work, we summarized basic concepts of unsupervised deep learning comprehensively and compared performances of several state-of-the-art unsupervised learning methods for MRI reconstruction.

Keywords: MRI reconstruction; Unsupervised deep learning; Prior information; Data distribution

INTRODUCTION

Magnetic resonance imaging (MRI) has been widely used in the field of bio-medicine because of its high resolution, non-invasiveness, bio-safety, and many other advantages. Nevertheless, MRI is associated with a slow acquisition process because data samples are acquired in k-space and the speed is limited by underlying MR physics and physiology. A long sampling acquisition procedure places significant demands on patients, making this tool expensive and less accessible.

To overcome such drawback, compressed sensing MRI (CS-MRI) has been extensively exploited in the past decade (1-10). Nevertheless, there exist many intrinsic limitations in traditional CS-MRI reconstruction methods. In particular, most of them have merely focused on information of interest from MR images while neglecting other potentially useful information. For instance, total variation regularization (8) only focuses on local information of MR images. Patch-based approaches (9, 10) use linear combinations of dictionary learning to reconstruct MR images, leading to insufficient modeling of image features.

To alleviate these deficiencies mentioned above, deep learning-based (DL) models have appeared to leverage CS-MRI (11-21). As shown in Figure 1, DL models can be primarily categorized into two groups: supervised learning (SL) and unsupervised learning (USL) (22). SL is a kind of learning technique that integrates prior knowledge implicitly by using pairs of data samples. Wang et al. (11) have used a convolutional neural network (CNN) to learn the mapping from a zero-filled input image to a fully-sampled image directly. Ronneberger et al. (12) have presented a U-net architecture that can efficiently use feature information in the image domain. This structure includes a contracting path to get context information which can effectively obtain a low-level information of the image. Lee et al. (13) have presented a residual learning method into the original U-net which can reduce potential aliasing artifacts to get more accurate MR images. In a latter study (14, 15), several data-consistency layers have been embedded in a feed-forward CNN to keep the reconstructed image to be consistent with k-space data. Following the well-known “unrolling” of iterative reconstruction approach, a semi-iterative convolutional network has been developed as a model-based deep learning architecture for inverse problems (MoDL) (14). These methods may exhibit superior performance in some predetermined acquisitions. However, they lack flexibility and stability when environments of under-sampling schemes and acceleration factors have some perturbations (23).

In contrast to previously proposed SL approaches that require paired datasets for training, USL learns the image's prior information from unlabeled data for various reconstruction tasks (16-21). In particular, the prior model has an explicit probability distribution function which can be used as a regularization term for iterative image reconstruction after the prior network architecture is

trained. Liu et al. (16) have proposed an effective way to incorporate the denoising autoencoder (DAE) prior to address the highly under-sampling MR image reconstruction problem. It could be applied to reconstruction tasks with different sampling trajectories and acceleration factors once the DAE prior is obtained. Zhang et al. (17) have presented a multi-channel enhanced deep mean-shift prior (MEDMSP) to address the highly under-sampled MRI reconstruction problem. Tezcan et al. (18) have used variational autoencoder (VAE) as an explicit prior term into the MRI reconstruction task. It learns from the probability distribution of fully sampled MR images. Mardani et al. (19, 20) have developed a CS framework by utilizing generative adversarial networks (GAN) to train a distribution of diagnostic-quality MR images from historical patients. Luo et al. (21) have utilized PixelCNN++ as a generative network for exploiting the image prior by maximizing the posterior possibility. In summary, as USL provides an explicit prior term, all above-mentioned approaches can model the MRI reconstruction problem from Bayes's theorem originally.

In this study, we provide an overview of different USL schemes for MRI reconstruction. First, we summarized different prior formulations in USL architectures. Second, their uniform formulation as a regularization term for MRI reconstruction was obtained. Finally, extensive experimental comparisons were performed.

1. An Overview of USL

In this section, some representative USL architectures are briefly reviewed. The utmost characteristic of USL methods (22, 24-27) is that the trained model can be used as a universal priori information for image reconstruction under various imaging conditions. Decoupling of prior and data consistency (DC) terms eliminates possible sensitivities in accuracy to deviations in acquisition specifications. USL methods vary from the perspective of learning data distribution. Overall, there are mainly two USL methods: autoencoder (AE) learning (28-30) (e.g., denoising AE and variational AE, etc.) and deep generative networks (21, 31, 32) (e.g., GAN, PixelCNN, flow-based generative model, etc.).

Given a data sample x , we aim to obtain data distribution $p(x)$. Differences among various USL models are used to optimize the distribution $p(x)$ of training data (33-40):

(1) The relationship between DAE and generative model has not been fully revealed by researchers in the early period. Until Song et al. (41) used denoising score matching

Fig. 1. Different types of existing deep learning-based CS-MRI algorithms.

and Langevin dynamics method to generate images in 2019, DAE was regarded as a generative model. At the same time, Liu et al. (16) and Zhang et al. (17) utilized it as an explicit prior information for MRI reconstruction.

(2) VAE can realize image generation by constraining latent coding vectors to have Gaussian distribution (26-28). Tezcan et al. (18) have used it to estimate the distribution of MR image patch and developed a deep density prior for MRI reconstruction. Essentially, VAE is an approximate generative model as it maximizes the lower bound of the data likelihood function.

(3) An invertible flow-based model expects to represent data with a simple latent variable distribution (42). It can provide efficient inference and synthesis. However, the reversible neural network computing gradient requires a certain amount of memory which limits the depth of the network architecture.

(4) An autoregressive model optimizes the explicit likelihood function through chain rule. It shows an excellent performance on PixelCNN/PixelRNN (43). Unfortunately, since the image is generated by pixels Bayesian estimation, it leads to a high computational cost.

(5) GAN gradually optimizes the generator and discriminator to generate image that has the same distribution as the ground-truth (29-32, 41). Although GAN may avoid some limitations of likelihood-based models, its training can be unstable due to the alternatingly adversarial scheme (44, 45). It also causes problems such as the lack of generative diversity and not encoding the latent space, although its performance is promising. Meanwhile, many current GAN-based approaches still belong to SL as they only utilize a discriminator to leverage the end-to-end learning ability (44, 45).

Some state-of-the-art USL methods are described in Table 1. It can be seen that, except for GAN, other methods including VAE, autoregressive method, and flow-based model are all likelihood-based methods using log-likelihood as the training objective.

2. USL Methods for MRI Reconstruction

2.1 Brief Overviews of MRI Reconstruction

Motivated by the compressed sensing theory, a number of related methods have been proposed to accelerate MRI data acquisition. In particular, CS-MRI can be introduced to help reduce the number of measure data in k-space without requiring additional hardware (46). It not only reduces extra cost and the scanning time, but also

removes potential motion artefacts and relieves a patient's discomfort. However, due to incoherence of under-sampled measure data in k-space, it will produce aliasing artifacts in the image domain. One solution for this issue is to find a method that can effectively utilize the priori information to reconstruct partially damaged MR data (9, 47-52).

Mathematically speaking, the CS-MRI problem can be formulated as follows:

$$y = F_p x + e \quad [1]$$

where $F_p \in \mathbb{C}^{M \times N}$ is a undersampling Fourier encoding matrix with $F_p = MF$ and $M \ll N$. $e \in \mathbb{C}^M$ is the effect on measurements of noise and scanner imprecisions. In order to reconstruct x , we have to exploit the priori information of MR images and construct an optimization problem to obtain an optimal solution:

$$\min_x ||F_p x - y||_2^2 + \lambda R(x) \quad [2]$$

where the term $R(x)$ denotes the prior term of x , $||F_p x - y||_2^2$ is the data fidelity term, and λ is a positive parameter that balances the importance of the two terms mentioned earlier.

In the past decade, the development of MRI reconstruction algorithms has mainly explored novel image priors based on compressed sensing theory. One major trend is to

Table 1. Some USL Algorithms and Their Corresponding Brief Descriptions (Y/N stand for Yes/No)

Methods	Core architecture	Is it based on likelihood?	Unsupervised or not?	Descriptions
EDAEPRec (16) (2020)	DAE	Y	Y	Enhanced DAE prior
MEDMSPRec (17) (2020)	DAE	Y	Y	Multi-noise and multi-channel prior
DDP (18) (2019)	VAE	Y	Y	Based on VAE prior learning
GDP (21) (2020)	PixelCNN	Y	Y	Deep Bayesian estimation
Glow (42) (2018)	Flow-based	Y	Y	Invertible generative network
RefineGAN (20) (2018)	GAN	N	N	GAN with cycle loss
DCGAN (45) (2017)	GAN	N	Y	Deep convolutional GAN

exploit sparse or low-rank prior information with an adaptive strategy (49–53). Ravishankar and Bresler (9) have proposed a dictionary learning framework for adaptively learning the sparse transform and reconstructing the image simultaneously from highly under-sampled k-space data. It is termed DLMRI. Especially, the prior term can be expressed as $R(x) = \sum_{ij} \|P_{ij}x - Da_{ij}\|_2^2$, where P_{ij} represents the operator to extract the image patch x_{ij} as $x_{ij} = P_{ij}x$ and D is the dictionary. The whole term captures the patch-based quality of sparse approximations with respect to the dictionary. After the dictionary to be learned, the reconstruction problem can be solved with fixed dictionary and sparse representation. The corresponding update problem is:

$$\min_x \|F_p x - y\|_2^2 + \lambda \sum_{ij} \|P_{ij}x - Da_{ij}\|_2^2 \quad [3]$$

which is a simple least square problem. An optimal analytical solution can then be obtained.

Recently, deep learning that collects a large size of dataset to train a network to learn the mapping between observed data and ideal reconstruction data purely from the perspective of end-to-end learning has been reported (14, 54–56). Schlemper et al. (14) have innovatively integrated the iterative reconstruction into the deep learning with a CNN named a deep cascade of CNN (DC-CNN). Unlike DLMRI, DC-CNN uses convolutional neural network to learn image representations instead of dictionary learning. The optimization problem becomes:

$$\begin{cases} x'_p = f_{cnn}(x_p | \theta) \\ x_p = \min_x \|F_p x - y\|_2^2 + \lambda \|x - x'_p\|_2^2 \end{cases} \quad [4]$$

Here, f_{cnn} is the forward mapping of the CNN with parameter θ , which takes in the under-sampled reconstruction $x_p = F_p^H y$ as an input and directly outputs

a reconstruction image. Furthermore, data fidelity is incorporated into the network architecture which is called DC in k-space.

After defining the forward and backward of DC layer, it can be trained as a second CNN which learns to reconstruct from the output of the first CNN. The DC-CNN architecture is exhibited in Figure 2. As can be seen, a new CNN is concatenated on the previous output to build an extremely deep network which iterates between intermediate de-aliasing and the DC layer. In fact, it essentially unfolds the optimization process of DLMRI. This category of DL integrates priori information of the network structure of fully-sampled images to reconstruct for missing k-space measurements in an implicit manner and embeds them into trained weights of feed-forward networks.

2.2. USL for MRI Reconstruction

In contrast to an SL scheme that learns the prior information implicitly, USL tackles MRI reconstruction by exploiting image prior explicitly. A flowchart of MRI reconstruction with general USL methods is shown in Figure 3, consisting of an unsupervised prior learning step and an iterative reconstruction step. These methods aim to learn the probability distribution of MR images, particularly the gradient of the prior, by means of network training. After that, priors as explicit constraints are applied to the MRI reconstruction.

At the prior learning stage, the network usually optimizes model parameters by learning the distribution mapping from image-to-image, variable-to-image, or image-to-variable. With the trained prior model at the iterative reconstruction stage, the explicit $prior(x)$ implements the iterative reconstruction algorithm for constrained solution space while enforcing k-space data fidelity.

By introducing the proximal gradient descent, the MRI reconstruction can be expressed as an alternating iterative

Fig. 2. A cascade of CNNs in DC-CNN. The depth of cascade is denoted by nc .

Fig. 3. Flowchart of the present USL schemes (left) and the associated iterative reconstruction procedure (right) for CS-MRI.

process which decouples updated priori information and Tikhonov regularization with DC constraint. In general, the USL model can be written as:

$$\begin{cases} x^{k+1/2} = x^k - \alpha \nabla \text{prior}(x^k) \\ x^{k+1} = \arg \min_x ||F_p x - y||_2^2 + \lambda ||x - x^{k+1/2}||_2^2 \end{cases} \quad [5]$$

where α is the gradient step size and the second minimization formula is a least square problem which can be solved by calculating its gradient:

$$(\gamma F_p^H F_p + 1) x^{k+1} = \gamma F_p^H y + x^{k+1/2} \quad [6]$$

In particular, let $F \in \mathbb{C}^{N \times N}$ denote the full Fourier encoding matrix with $F^H F = I_N$, by transforming Eq. [6] from image space to k-space, it can be rewritten as:

$$(\gamma F F_p^H F_p^H + F F^H) F x^{k+1} = \gamma F F_p^H y + F x^k \quad [7]$$

Assuming that Ω represents the sampled data subset and $S_1 = F F_p^H y$, $S_2 = F x^k$, it yields

$$F x(k_i, k_j) = \begin{cases} S_2(k_i, k_j), & (k_i, k_j) \notin \Omega \\ \frac{S_1(k_i, k_j) + \lambda S_2(k_i, k_j)}{1 + \lambda}, & (k_i, k_j) \in \Omega \end{cases} \quad [8]$$

where $F x(k_i, k_j)$ stands for the updated value at an under-sampled k-space location (k_i, k_j) .

In summary, a two-step alternative manner is needed for MRI reconstruction by tackling the data-fidelity term and the regularization term subsequently. Generally, each USL algorithm learns a different prior. For instance, VAE learns priori information in the image patch rather than the whole image. Therefore, patch extraction and summarization operations are conducted before and after entering the network, respectively. Since the complexity of PixelCNN is very high, it should split the image to multiple batches first and then compose them as the network input. Algorithm 1 provides a summary of the reconstruction procedure.

2.3. Various Formulation of Explicit Prior $p(x)$

Except for GAN, almost all deep generative models learn data distribution by optimizing the principle of logarithmic data density. What they differ is how they represent or approximate the likelihood. Explicit models are those that explicitly evaluate the density p_{model} and maximize the likelihood.

In this section, we will review priori explicit representations of DAE, VAE, PixelCNN model, flow-based model, and GAN in details and analyze main characteristics of each method.

2.3.1. Denoising Autoencoding (DAE)

A DAE (29) is an AE trained to reconstruct data corrupted by Gaussian noise as shown in Figure 4. An interesting property of the DAE is that it can explicitly correspond to a generative model. Its training criterion has limitation for the log-likelihood of a generative model. Another interesting property of DAE is that it is naturally suitable for data with missing values or multi-modal data. This is because it is trained with inputs that have "missing" parts corrupted by noise. Intuitively, a DAE implements two things: 1) it tries to encode the input and manage to undo effects of the corruption process stochastically applied to the input of the AE; and 2) it captures statistical dependencies between inputs to complete the denoising process. Previously, Alain and Bengio (57) and Nguyen et al. (58) have used DAEs to construct generative models. Essentially, denoising in DAE serves as a regularized criterion for learning to capture useful structure from the input data.

Bigdeli and Zwicker (59) have used the magnitude of DAE error as a prior (DAEP) for image reconstruction. DAE A_σ was

trained to minimize the expectation over all input images x and noise μ :

$$L_{DAE}(A) = E_{x, \mu} [||A_\sigma(x+\mu) - x||^2] \quad [9]$$

where the output $A_\sigma(x)$ was trained by adding artificial Gaussian noise μ with standard deviation σ .

According to (59), the network output $A_\sigma(x)$ is related to true data density $p(x)$ as follows:

$$\begin{aligned} A_\sigma(x) &= x - \frac{\int g_\sigma(\mu)p(x-\mu)d\mu}{\int g_\sigma(\mu)p(x-\mu)d\mu} \\ &= x + \nabla \log \int g_\sigma(\mu)p(x+\mu)d\mu \end{aligned} \quad [10]$$

where $g_\sigma(\mu)$ represents a local Gaussian kernel with standard deviation σ . More importantly, the AE error $P_{model}(x) = A_\sigma(x) - x = \nabla \log \int g_\sigma(\mu)p(x+\mu)d\mu$ is proportional to the gradient of the log likelihood of the smoothed density. Due to this observation, Liu et al. (16) have proposed a simple and effective way to incorporate the DAE prior into a highly under-sampling MRI reconstruction, leading to a promising performance.

As seen in Eq. [10], DAE learns a mean-shift vector field from a given set of data samples. It is proportional to the gradient of the logarithm of the prior. Hence, a new prior called deep mean-shift prior (DMSP) has been proposed by Bigdeli et al. (60). They utilized it in a gradient descent approach to perform Bayes risk minimization. DMSP is formulated as follows:

$$\begin{aligned} P_{model}(x) &= \nabla prior(x) = \nabla \log \int g_\sigma(\mu)p(x+\mu)d\mu \\ &= [(A_\sigma(x) - x)/\sigma^2] \end{aligned} \quad [11]$$

By extending the naive DMSP with integration of multi-model aggregation and multi-channel network learning, Zhang et al. (17) have proposed a high-dimensional embedding network derived prior and applied the learned prior to single-channel MRI reconstruction via a variable augmentation technique.

Algorithm 1: Unsupervised Learning for Reconstruction (USLearn)

Prior learning stage
Input: MR dataset: $x \in \mathbb{C}^{n \times n}$
Output: Trained network model by learning $p(x)$
Iterative reconstruction stage
1: Initialization: $x^0 = F_p^H Y$
2: For $k=0,1,2,\dots,k$ do
3: Pre-process to get the corresponding network input $m^k = pre(x^k)$
4: Get gradient $\nabla_m prior(m)$ at m^k
5: Update $m^{k+1} = m^k + \alpha \nabla_m prior(m^k)$
6: Post-process $x^{k+1} = post(m^{k+1})$ for projection
7: Projection x^{k+1} in Eq. [8]
8: Return x^k

Fig. 4. Visual comparison of DAE architecture.

Fig. 5. The standard VAE model presented as a graphical model.

2.3.2. Variational Autoencoders (VAE)

VAE approximates the distribution of high-dimensional data as a USL method. Figure 5 depicts the graphical model of VAE. Specifically, the CNN Encoder provides mean and covariance variables from the input data x and then forms z by adding Gaussian noise e . The CNN Decoder aims to map the latent vector z to the conditional distribution of the original data. It uses simple ancestor sampling to perform very effective approximate posterior inference, which in turn enables us to effectively learn model parameters without requiring expensive iterative inference schemes for each data point (61, 62). The learned approximate posterior reasoning model can also be used for a series of tasks such as recognition, denoising, and reconstruction.

Unlike many USL methods that handle a complex-valued image directly, VAE focuses on the magnitude of image patches in MRI reconstruction (18). Using VAE, it is possible to approximate the distribution of MR image patches and the possibility of under-sampled missing images. $m^k = pre(x^k)$ is a decompose operator and $x^{k+1} = post(m^{k+1})$ is combine operator. In addition, the approximate likelihood function uses a neural network. Thus, it becomes differentiable. Consequently, VAE can be used as a priori model to approximate the distribution of large image patches for iterative reconstruction. The main goal of the VAE algorithm is to combine the encoder and the decoder to make latent variables approximate data distribution of the original image patch and use variational approximation to optimize parameters of a given data set. The VAE model is given as follows:

$$p(x) = \int_z p(x, z) dz = \int_z p(x|z) p(z) dz \quad [12]$$

where x represents image patch of P pixels of complete MR image. $z \in R^L$ denotes the latent variable. $P(z)$ is expressed as a prior of z and $L \ll P$. A known distribution is assumed for $p(z)$ and a parameterized $p(x|z)$ is optimized to maximize $\log p(x)$ of observed samples. Other probabilistic latent

variable models such as probabilistic principal component analysis (63) also use this modeling strategy. The VAE model parameterizes $p(x|z)$ into a neural network. To optimize $\log p(x)$ for a given data set, the integral over z needs to be evaluated. This is not feasible even for moderate L . This problem can be solved with variational approximation that uses an approximate distribution for the posterior $q(z|x) \approx p(z|x)$. Using $q(z|x)$, $\log p(x)$ can be decomposed into two terms (63):

$$\log p(x) = E_{q(z|x)} \left[\log \frac{p(x, z)}{q(z|x)} \right] + D_{KL}[q(z|x) || p(z|x)] \quad [13]$$

The first term is denoted as the lower bound of evidence (ELBO) and the second term is the Kullback-Leibler divergence (KLD) between the approximate and the true posterior. The second term is intractable because the true posterior $p(z|x)$ is recalcitrant. However, it is always larger than or equal to zero, making ELBO to be a lower bound for $\log p(x)$. The strategy of VAE is to maximize the ELBO as a proxy to $\log p(x)$.

In a previous study (18), the authors used the VAE model to learn the distribution of MR patches from fully-sampled images, avoiding the sensitivity of the model to the sampling pattern. Then they used this learned distribution as a probabilistic prior in a Bayesian reconstruction framework. The VAE yielded promising results for MRI reconstruction.

2.3.3. Pixel CNN

PixelCNN is a USL algorithm that models the discrete probability of raw pixel values and encodes pixel dependencies in the image (43). $m^k = pre(x^k)$ is a split operator and $x^{k+1} = post(m^{k+1})$ is a combined operator. In order to model a conditional distribution of pixels in the image, the model fully factorizes the probability density function on an image x over all its sub-pixels up and to the left (43) as described below:

$$p(x) = \prod_{i=1}^{n^2} p(x_i | x_{1, \dots, x_{i-1}}) \quad [14]$$

Salimans et al. (64) have put forward a number of modifications to PixelCNN called PixelCNN++. It both simplified its structure and improved its performance. They used a discretized logistic mixture likelihood on pixels and modelled a latent color intensity v with a continuous distribution as follows:

$$v \sim \sum_{i=1}^K \pi_i \text{logistic}(\mu_i, s_i) \quad [15]$$

where π_i is the mixture indicator, μ_i and s_i are the mean and scale of logistic distribution, respectively. PixelCNN++ serves as a generative AE that provides a hierarchic representation of the input image. It predicts a mixture image distribution. This method explicitly models the distribution of each pixel in relation to its causal neighbor. It can help better reconstruct low-level details without artifacts.

Using PixelCNN++ (65) as a data-driven MRI prior for Bayesian inference, Luo et al. (21) have proposed a generic and interpretable MRI reconstruction framework. The difference in modeling pixel distribution with PixelCNN++ is that the number of image channels is changed from three (i.e., red, green, and blue channels for color image) to two (i.e., real and imaginary channels for MR image). The conditional distribution of the subsequent pixel is given by

$$p(x_{ij}|C_{ij}) = p(\text{Re}(x_{ij}), \text{Im}(x_{ij})|C_{ij}) = P(\text{Re}(x_{ij})|\mu_{\text{Re}}(C_{ij}), S_{\text{Re}}(C_{ij})) \times P(\text{Im}(x_{ij})|\mu_{\text{Im}}(C_{ij}), \text{Re}(x_{ij}), S_{\text{Im}}(C_{ij})) \quad [16]$$

where $C_{ij} = \{x_{i-1,j}, x_{i-2,j}, \dots, x_{1,1}\}$ denotes the context information. Therefore, the model is expected to predict joint probability distribution of all pixels in the input image as illustrated. The architecture of the PixelCNN++-based model is shown in Figure 6. As depicted, it follows a two-stream convolutional architecture with residual connections. The

network is grouped into sequences of six layers, most of which are separated by downsampling or upsampling. This prior ensures pixel-level consistencies and improves performance in preserving image details and reducing aliasing artifacts in the reconstruction.

2.3.4. Generative Flow (Glow)

By designing a chain of invertible transformations with Jacobian determinants, flow-based generative models can map points from a simple distribution to a complex one. This methodology guarantees that the density of the transformed distribution can be analytically estimated, making it feasible to perform maximum likelihood learning (66-69).

Specifically, the relationship between variable x and latent variable $z \in \mathbb{R}^n$ is a bijective map (68, 69). The complicated probability density x can be expressed as the likelihood of the transformation $z = F_\theta(x)$ evaluated by p_z , i.e.,

$$p_\theta(u) = p_z(z) |az/au| \quad [17]$$

where az/ax stands for the Jacobian F_θ of x and $p_z(z)$ is the prior distribution of z . The bijective mapping of x to z can be expressed as $x \xrightarrow{F_1} h_1 \xrightarrow{F_2} h_2 \xrightarrow{F_3} \dots \xrightarrow{F_L} z$, which is often referred to as a normalizing flow (42, 70). $\{h_i\}_{i=1}^L$ are intermediate representations produced by layers of the neural network, where $z = h_L$ and $h_0 = x$. The log-likelihood of x is equal to the log-likelihood of z and the log Jacobian determinant of each layer:

$$\log p_\theta(x) = \log p_z(z) + \sum_{i=1}^L \log |ah_i/ah_{i-1}| \quad [18]$$

Based on the NICE (67) and RealNVP (68), a generative flow-based model (Glow) (42) introduces a new flow

Fig. 6. Flowchart of PixelCNN++ network (64) as a prior model in MRI reconstruction.

Fig. 7. Schematic flowchart of Glow model. This architecture has a depth of Glow block L and the number of levels K . Detailed architecture of Glow can be referred to (42).

Fig. 8. The standard GAN presented as a generative model.

where each step consists of an Actnorm layer, an invertible convolution layer, and an affine coupling transformation. Concretely, this architecture allows exact latent-variable inference, direct log-likelihood evaluation, and efficient image synthesis. The specific architecture of the network is shown in Figure 7. The model output has a one-dimensional Gaussian distribution. Therefore, it has the potential for significant memory savings in gradient computations.

2.3.5. Generative Adversarial Network (GAN)

The GAN (71) consists of two separate networks: a generator G and a discriminator D . The generator G learns a distribution $p_{model}(x)$ via a mapping $G(\cdot)$ of the latent variable z , where z is a 1D vector of uniformly distributed input.

As can be seen in Figure 8, the generator architecture is equivalent to a decoder by utilizing a convolution network. The discriminator D is composed of a series of standard

CNNs. It maps a scalar value $D(x) \in [0,1]$, where $D(x)$ denotes the probability that the input follows the data distribution p_{data} . The purpose of the discriminator is to distinguish real and fake data. On the other hand, G takes z from a given distribution p_z to create data as real as possible to fool the discriminator. This problem can be formulated as a minimax game:

$$\min_G \max_D L(D, G) = E_{x \sim p_{data}} [\log D(x)] + E_{z \sim p_z(z)} [\log (1 - D(G(z)))] \quad [19]$$

where E denotes expectation and x and z are samples drawn from p_{data} and p_z , respectively.

GAN is an implicit generative model (66) that directly samples from the distribution represented by the model and evaluates a likelihood mapped from the latent space to the image domain (66). To find the optimal z in the reconstruction phase, one starts randomly sampling z_1

from the latent spatial distribution Z and then feeds it into a trained generator to obtain the generated image $G(z_i)$. Based on the generated image $G(z_i)$, we can define a loss function between real and generated distribution. In order to find the most similar image $G(z_n)$, the location of z in the latent space Z is optimized in an iterative process via back-propagation steps.

3. Experimental Results

In this section, four types of competing algorithms (DAE, VAE, PixelCNN, and flow-based model) are provided for MRI reconstruction. Besides, the popular dictionary learning DLMRI and the recent supervised deep learning method DC-CNN are selected. For the convenience of reproducible research, source code for experimental results is available at <https://github.com/yqx7150/USLearn>. It should be emphasized that only some competing algorithms are available for MRI reconstruction. For algorithms that are unavailable, we can modify them to be suitable for the reconstruction task. Particularly, the code of DLMRI is available at <http://www.ifp.illinois.edu/~yoram/DLMRI-Lab/DLMRI.html>. Code of DC-CNN is available at <https://github.com/js3611/Deep-MRI-Reconstruction>. Code of DDP is originally from https://github.com/kctezcan/ddp_recon. These codes of PixelCNN and Glow are originally from <https://github.com/openai/pixel-cnn> and <https://github.com/openai/glow>, respectively.

3.1. Training Data

The performance of these methods is verified under a variety of sampling patterns and acceleration factors (R). The sampling schemes include 2D random sampling, Pseudo radial sampling and 1D Cartesian sampling. Specially, $R = 3.3, 4, 5, 6.7, 10$ is used in the experiments. The raw data were acquired from T2-weighted 3D fast-spin-echo sequence through a 3.0T whole body MR system. The repetition time (TR) is 2000 ms and the echo time (TE) is 13.0 ms. Each pale has 192 slices, and the thickness of each slice is 0.86 mm.

The field-of-view (FOV) and voxel are set to $220 \times 220 \text{ mm}^2$ and $0.9 \times 0.9 \times 0.9 \text{ mm}$, respectively. The complex-valued raw data are obtained by the SENSE-TV reconstruction algorithm (72). In particular, we divided the collected complex-valued 531 MR images with the size of 256×256 into three groups: 480 images were used for network training, 20 images for validation, and 31 images were used as test set to evaluate the performance. Figure 9 displays the fully-sampled data of 31 complex-valued MR images. For fair comparison, all reference MR images are normalized to a maximum magnitude of 1 in this paper.

3.2 Reconstruction Measures and Running Environment

Three quantitative criteria are utilized to evaluate the reconstruction performance. The first measure is peak signal-to-noise ratio (PSNR) that represents error estimated between original image and reconstruction image:

$$PSNR(x, \hat{x}) = 20 \log_{10} \frac{\text{Max}(\hat{x})}{\|x - \hat{x}\|_2} \quad [20]$$

The second quality measure is the structural similarity index (SSIM). SSIM index is bounded. The closer the index is to 1, the better the reconstruction scheme is implied.

$$SSIM(x, \hat{x}) = \frac{(2\mu_x \mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)} \quad [21]$$

where μ_x and σ_x^2 are the average and variances of x . $\sigma_{x\hat{x}}$ is the covariance of x and $\hat{x}$, c_1 and c_2 are used to maintain a stable constant. SSIM gives the measure of the structural similarity between the ground truth and the estimated image.

Finally, HFEN metric is used to quantify the quality of reconstruction of edges and fine features. It employs a rotationally symmetric LoG (Laplacian of Gaussian) filter to capture edges. The filter kernel is of size pixels and has a standard deviation of 1.5 pixels. HFEN is calculated as the ratio between two norms:

Fig. 9. 31 test image examples in SIAT dataset. The images were acquired with a 12-channel head coil by T1-weighted turbo spin-echo (TSE) sequence (TE = 13.0 ms, TR = 2000 ms, FOV = 220 mm^2 , matrix = 256×256 , slice thickness = 0.86 mm).

Table 2. The Running Environment of Different Algorithms

	DLMRI	DC-CNN	EDAEP Rec	MEDMSP Rec	DDP	PixelCNN	Glow
Operating system	Windows	Windows	Windows	Windows	Ubuntu	Ubuntu	Ubuntu
Programming language	MATLAB	Python	MATLAB	MATLAB	Python	Python	Python
Machine learning framework	---	Theano	Caffe	MatConvNet	TensorFlow	TensorFlow	PyTorch

Table 3. Average PSNR, SSIM and HFEN Values Obtained by Various Algorithms at 2D Random and Pseudo Radial Sampling Trajectories with $R = 3.3, 4$, and 5 . The Best Value of Each Row is Highlighted with Bold

(a)	DLMRI	DC-CNN	DAEPRec	DMSPRec	DDP	PixelCNN	Glow	EDAEPRec	MEDMSPRec
$R=3.3, 2D$	33.98		33.75	33.22	36.31	35.55	34.29	36.32	36.24
Random	0.9204	---	0.9035	0.8956	0.9469	0.9353	0.9317	0.9447	0.9446
	0.45		0.70	0.75	0.37	0.60	0.62	0.31	0.32
$R=4, 2D$	33.13		32.81	32.25	35.44	34.33	33.18	35.61	35.26
Random	0.9063	---	0.8876	0.8775	0.9381	0.9206	0.9197	0.9360	0.9348
	0.57		0.83	0.89	0.47	0.78	0.79	0.39	0.36
$R=5, 2D$	32.18		31.26	30.81	33.89	32.96	32.17	34.59	34.35
Random	0.8865	---	0.8615	0.8497	0.9198	0.8985	0.8923	0.9190	0.9018
	0.83		0.98	0.06	0.69	0.94	0.95	0.54	0.46
(b)	DLMRI	---	DAEPRec	DMSPRec	DDP	PixelCNN	Glow	EDAEPRec	MEDMSPRec
$R=3.3,$	34.10		34.11	33.51	36.29	36.46	34.63	36.52	36.55
Pseudo	0.9216	---	0.9085	0.9002	0.9450	0.9485	0.9347	0.9464	0.9473
Radial	0.45		0.67	0.71	0.36	0.46	0.56	0.33	0.33
$R=4, Pseudo$	33.21		33.28	32.62	34.98	35.31	33.48	35.69	35.64
Radial	0.9066	---	0.8946	0.884	0.9320	0.9360	0.9216	0.9370	0.9373
	0.61		0.80	0.88	0.49	0.64	0.74	0.43	0.42
$R=5, Pseudo$	32.13		32.33	31.60	33.77	33.52	32.39	34.69	34.50
Radial	0.8852	---	0.8766	0.8628	0.9180	0.9105	0.9077	0.9241	0.9231
	0.83		0.97	1.08	0.63	0.93	0.91	0.56	0.56

$$HFEN = ||\text{LoG}(x) - \text{LoG}(\hat{x})||_F^2 / ||\text{LoG}(\hat{x})||_F^2 \quad [22]$$

To better understand these algorithms, running environments of different algorithms are listed in Table 2, including the operating system, programming language and machine learning framework.

3.3 Comparison on Real-Valued SIAT Dataset

For the sake of evaluating the robustness and reconstruction accuracy among these methods, three image quality metrics are listed in Table 3. It can be seen that DAE, VAE, PixelCNN and Glow based methods performance better than the traditional method DLMRI. This implies that the USL algorithms achieved better evaluation indicators, such as EDAEPRec, MEDMSPRec and DDP. Although the reconstruction results of PixelCNN and Glow are inferior to other algorithms in the evaluation index, their reconstruction results have improved significantly under higher acceleration factors. It is noticed that DDP achieves

the highest values of SSIM under 2D random sampling. Furthermore, due to the multi-channel and multi-model strategy, EDAEPRec and MEDMSPRec significantly improve the performance than the other methods under pseudo radial trajectory.

Additionally, to further exhibit the superiority of these algorithms, some visual results are depicted in Figures 10 and 11. First, by using DLMRI, there is excessive loss of detail in the reconstructed image. Although DAEPRC and DMSPRec partially alleviate the deficiency, smoothed effect still exists. Generative models have the advantage of generating detailed information. Unfortunately, PixelCNN and Glow may introduce unnatural "noise-like" patterns in smooth-region and along edge contour. However, DDP, EDAEPRec and MEDMSPRec fill in details to match the original image.

Furthermore, we evaluate the computational costs of a 256×256 MR image in Table 4, where the run time is recorded in order to facilitate the comparison for this type

Fig. 10. Visual comparisons under 2D Random sampling at an acceleration factor $R = 4$. (Top) From left to right: Reference image, reconstruction images obtained by DLMRI, DAEPRec, and DMSPRec. (Middle) From left to right: Reconstruction images obtained by DDP, PixelCNN, Glow, and EDAEPRec. (Bottom): Enlarged reconstructions are exhibited.

Fig. 11. Visual comparisons under Pseudo Radial sampling at an acceleration factor $R = 5$. (Top) From left to right: Reference image, reconstruction images obtained by DLMRI, DAEPRec, and DMSPRec. (Middle) From left to right in the second row: Reconstruction images obtained by DDP, PixelCNN, Glow, and EDAEPRec. (Bottom) Enlarged reconstruction regions are exhibited.

Table 4. Computational Cost Evaluations of Various Reconstruction Methods. The Original Image is a 256×256 MR Image. Notice that DLMRI is Executed with CPU while Others are Executed with GPU

	DLMRI	DDP	PixelCNN	Glow	EDAEPRec	MEDMSPRec
Cost time (S)	363.50	178.80	141.60	228.60	190.00	41.50
	14.54/iter	2.98/iter	1.18/iter	7.62/iter	0.96/iter	0.83/iter

Table 5. Average Reconstruction PSNR, SSIM, and HFEN Values of All Test Images with Various Sampling Trajectories and Undersampling Percentages. The Best Value of Each Row is Highlighted with Bold

(a)	DLMRI	DC-CNN	DDP	PixelCNN	Glow	DAEPRec	MEDMSPRec
$R = 3.3$, Pseudo	33.43	35.71	33.50	33.92		35.62	34.76
Radial	0.9054	0.9234	0.9109	0.9008	---	0.9279	0.9193
	0.63	0.44	0.78	0.55		0.42	0.58
$R = 4$, Pseudo	32.41	34.07	32.77	33.40		34.49	34.11
Radial	0.8866	0.8992	0.8991	0.8920	---	0.9151	0.9098
	0.84	0.69	0.91	0.64		0.64	0.68
$R = 5$, Pseudo	31.21	32.68	31.50	32.51		33.49	33.00
Radial	0.8602	0.8791	0.8751	0.8754	---	0.8990	0.8908
	1.10	0.95	1.14	0.81		0.79	0.86
$R = 10$, Pseudo	27.39	28.39	27.83	29.70		30.30	29.91
Radial	0.7444	0.7710	0.7741	0.8088	---	0.8319	0.8206
	2.18	1.93	2.06	1.45		1.40	1.48
(b)	DLMRI	DC-CNN	DDP	PixelCNN	Glow	EDAEPRec	MEDMSPRec
$R = 6.7$, 2D	27.63	28.78	27.37	305.53		30.68	30.36
Random	0.7518	0.7873	0.7543	0.8303	---	0.8433	0.8350
	2.02	1.83	2.02	1.29		1.31	1.38
$R = 6.7$, Pseudo	29.36	30.57	29.90	31.26		32.00	31.54
Radial	0.8103	0.8348	0.8381	0.8491	---	0.8716	0.8495
	1.58	1.38	1.50	1.68		1.05	1.25
$R = 6.7$, 1D	26.50	27.05	26.70	28.28		28.85	28.45
Cartesian	0.7390	0.7506	0.7622	0.7837	---	0.8041	0.7935
	2.51	2.44	2.46	1.89		1.81	1.91

of algorithms. It is worth noting that DLMRI is implemented on CPU while the others are executed with GPU. As can be observed, the total computational costs of Glow, DDP, PixelCNN and EDAEPRec are almost same, while the cost time of each iteration in Glow locates the largest. This is due to the large number of reversible operators in Glow leads to more reconstruction time than other algorithms. Finally, the running time of EDMSPRec is the shortest because it just calculates the gradient of the log-density function. Overall, as the complexity of model increases, the computational time at each reconstruction iteration will definitely increase.

3.4 Comparison of Complex-valued SIAT Dataset

Results of objective assessment for complex-valued dataset are exhibited in Table 5, where average PSNR,

SSIM, and HFEN values of these methods are recorded. Among all compared algorithms, only DC-CNN is a SL method which needs to be trained separately with different sampling trajectories and acceleration factors. Although DC-CNN incorporates the data-fidelity term to enhance the ability of end-to-end mapping, it is inferior to EDAEPRec, particularly when R is 6.7 and 10. Except for pseudo radial sampling at acceleration factor $R = 3.3$ that the SL method DC-CNN achieved the best performance, the USL method showed obvious advantages for other sampling patterns. Particularly, the generative model can effectively recover missing data when less data are available in the k -space. Finally, EDAEPRec achieved better performance than other USL methods such as MEDMSPRec, PixelCNN, and DDP.

Figures 12 and 13 show visual qualities of reconstructions with different methods. As can be seen, DLMRI

Fig. 12. Reconstruction results and zoomed regions of various methods under Pseudo Radial sampling at an acceleration factor $R = 4$. (Top) From left to right: Reference image, degrade image, reconstruction images obtained by DLMRI and DC-CNN. (Middle) From left to right: Reconstruction images obtained by DDP, PixelCNN, EDAEPRec, and MEDMSPRec. (Bottom): Enlarged reconstruction regions are exhibited.

Fig. 13. Reconstruction results and enlarged regions of various methods under 2D Random sampling at an acceleration factor $R = 6.7$. (Top) From left to right: Reference image, degrade image, reconstruction images obtained by DLMRI and DC-CNN. (Middle) From left to right: Reconstruction images obtained by DDP, PixelCNN, EDAEPRec, and MEDMSPRec. (Bottom): Enlarged reconstruction regions are exhibited.

reconstructions have limitations in recovering the structure and texture. Reconstruction with unsupervised methods exhibited higher resolution than the dictionary learning method and SL methods. DC-CNN was obviously almost devoid of aliasing artifacts when R was 3.3 or 4. By contrast, some USL methods such as DDP and PixelCNN suffered from over-smoothing deficiency. Moreover, an enlargement area is presented to reveal structures and fine details preserved by each algorithm. As can be observed in zoom-in regions enclosed by green box, unsupervised methods successfully preserved the vertical line-like pattern, producing more pleasing visual effects. Additionally, EDAEPRec achieved more satisfying results with clearer contours and finer image details than PixCNN.

Acknowledgments

The authors sincerely thank anonymous reviewers for their valuable comments and constructive suggestions that are very helpful in the improvement of this paper. This work was supported in part by the National Natural Science Foundation of China under 61871206 and 81830056. It was also supported by a project of innovative special funds for graduate students (CX2019075, CX2019092).

REFERENCES

1. Heidemann RM, Ozsarlak O, Parizel PM, et al. A brief review of parallel magnetic resonance imaging. *Eur Radiol* 2003;13:2323-2337
2. Donoho DL. Compressed sensing. *IEEE Trans Inf Theory* 2006;52:1289-1306
3. Lustig M, Donoho DL, Santos JM, Pauly JM. Compressed sensing MRI. *IEEE Signal Proc Mag* 2008;25:72-82
4. Liang D, Liu B, Wang J, Ying L. Accelerating SENSE using compressed sensing. *Magn Reson Med* 2009;62:1574-1584
5. Huang J, Zhang S, Metaxas D. Efficient MR image reconstruction for compressed MR imaging. *Med Image Anal* 2011;15:670-679
6. Lin FH, Kwong KK, Belliveau JW, Wald LL. Parallel imaging reconstruction using automatic regularization. *Magn Reson Med* 2004;51:559-567
7. Dong W, Shi G, Li X, Ma Y, Huang F. Compressive sensing via nonlocal low-rank regularization. *IEEE Trans Image Process* 2014;23:3618-3632
8. Eksioğlu EM. Decoupled algorithm for MRI reconstruction using nonlocal block matching model: BM3D-MRI. *J Math Imaging Vis* 2016;56:430-440
9. Ravishankar S, Bresler Y. MR image reconstruction from highly undersampled k-space data by dictionary learning. *IEEE Trans Med Imaging* 2011;30:1028-1041
10. Huang Y, Paisley J, Lin Q, Ding X, Fu X, Zhang XP. Bayesian nonparametric dictionary learning for compressed sensing MRI. *IEEE Trans Image Process* 2014;23:5007-5019
11. Wang S, Su Z, Ying L, et al. Accelerating magnetic resonance imaging via deep learning. *Proc IEEE Int Symp Biomed Imaging*, 2016:514-517
12. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. *MICCAI 2015: Medical Image Computing and Computer-Assisted Intervention*, 2015:234-241
13. Lee D, Yoo J, Ye JC. Deep residual learning for compressed sensing MRI. *Proc IEEE Int Symp Biomed Imaging*, 2017:15-18
14. Schlemper J, Caballero J, Hajnal JV, Price A, Rueckert D. A deep cascade of convolutional neural networks for MR image reconstruction. *IMPI 2017: Information Processing in Medical Imaging*, 2017:647-658
15. Aggarwal HK, Mani MP, Jacob M. MoDL: model-based deep learning architecture for inverse problems. *IEEE Trans Med Imaging* 2019;38:394-405
16. Liu Q, Yang Q, Cheng H, Wang S, Zhang M, Liang D. Highly undersampled magnetic resonance imaging reconstruction using autoencoding priors. *Magn Reson Med* 2020;83:322-336
17. Zhang M, Li M, Zhou J, et al. High-dimensional embedding network derived prior for compressive sensing MRI reconstruction. *Med Image Anal* 2020;64:101717
18. Tezcan KC, Baumgartner CF, Luechinger R, Pruessmann KP, Konukoglu E. MR image reconstruction using deep density Priors. *IEEE Trans Med Imaging* 2019;38:1633-1642
19. Mardani M, Gong E, Cheng JY, et al. Deep generative adversarial neural networks for compressive sensing MRI. *IEEE Trans Med Imaging* 2019;38:167-179
20. Mardani M, Monajemi H, Pappas V, Vasanawala S, Donoho D, Pauly J. Recurrent generative adversarial networks for proximal learning and automated compressive image recovery. *arXiv preprint arXiv* 2017:1711.10046
21. Luo G, Zhao N, Jiang W, Hui ES, Cao P. MRI reconstruction using deep Bayesian estimation. *Magn Reson Med* 2020;84:2246-2261
22. Hussein S, Kandel P, Bolan CW, Wallace MB, Bagci U. Lung and Pancreatic Tumor Characterization in the deep learning era: novel supervised and unsupervised learning approaches. *IEEE Trans Med Imaging* 2019;38:1777-1787
23. Antun V, Renna F, Poon C, Adcock B, Hansen AC. On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proc Natl Acad Sci U S A* & 2020;117:30088-30095

24. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media, 2009
25. Bengio Y, Courville AC, Vincent P. Unsupervised feature learning and deep learning: a review and new perspectives. CoRR 2012;abs/1206.5538
26. Erhan D, Courville A, Bengio Y, Vincent P. Why does unsupervised pre-training help deep learning? JMLR Workshop and Conference Proceedings, 2010:201-208
27. Bengio Y, LeCun Y. Scaling learning algorithms towards AI. Large-scale Kernel Machines 2007;34:1-41
28. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 2010;11:3371-3408
29. Vincent P, Larochelle H, Bengio Y, Manzagol P. Extracting and composing robust features with denoising autoencoders. In Proceedings of International Conference on Machine Learning, 2008:1096-1103
30. Rifai S, Vincent P, Muller X, Glorot X, Bengio Y. Contractive auto-encoders: explicit invariance during feature extraction. In Proceeding International Conference on Machine Learning (ICML), 2011:833-840
31. Lee H, Grosse R, Ranganath R, Ng AY. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In Proceeding International Conference on Machine Learning (ICML), 2009:609-616
32. Bengio Y. Deep learning of representations: looking forward. International Conference on Statistical Language and Speech Processing, 2013:1-37
33. Rolfe JT. Discrete variational autoencoders. arXiv preprint arXiv:1609.02200, 2016
34. Dilokthanakul N, Mediano PAM, Garnelo M, et al. Deep unsupervised clustering with gaussian mixture variational autoencoders. arXiv preprint arXiv:1611.02648, 2016
35. Kipf TN, Welling M. Variational graph auto-encoders. arXiv preprint arXiv:1611.07308, 2016
36. Sutskever I, Jozefowicz R, Gregor K, Rezende D, Lillicrap T, Vinyals O. Towards principled unsupervised learning. arXiv preprint arXiv:1511.06440, 2015
37. Yi Z, Zhang H, Tan P, Gong M. Dualgan: Unsupervised dual learning for image-to-image translation. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017:2849-2857
38. Yuan Y, Liu S, Zhang J, et al. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018:701-710
39. Wang C, Macnaught G, Papanastasiou G, ManGillcray T, Newby D. Unsupervised learning for cross-domain medical image synthesis using deformation invariant cycle consistency networks. International Workshop on Simulation and Synthesis in Medical Imaging, 2018:52-60
40. Lugmayr A, Danelljan M, Timofte R. Unsupervised learning for real-world super-resolution. IEEE/CVF International Conference on Computer Vision Workshop (ICCV Workshops), 2019:3408-3416
41. Song Y, Ermon S. Generative modeling by estimating gradients of the data distribution. 33rd Conference on Neural Information Processing Systems, 2019:11918-11930
42. Kingma DP, Dhariwal P. Glow: Generative flow with invertible 1x1 convolution. Advances in Neural Information Processing Systems 31 (NeurIPS 2018), 2018:10215-10224
43. Oord A, Kalchbrenner N, Kavukcuoglu K. Pixel recurrent neural networks. arXiv preprint arXiv:1601.06759, 2016
44. Quan TM, Nguyen-Duc T, Jeong WK. Compressed sensing MRI reconstruction using a generative adversarial network with a cyclic loss. IEEE Trans Med Imaging 2018;37:1488-1497
45. Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv: 1511.06434, 2015
46. Lustig M, Donoho DL, Santos JM, Pauly JM. Compressed sensing MRI. IEEE Signal Proc Mag 2008;25:72-82
47. Lin FH, Wang FN, Ahlfors SP, Hamalainen MS, Belliveau JW. Parallel MRI reconstruction using variance partitioning regularization. Magn Reson Med 2007;58:735-744
48. Akcakaya M, Nam S, Hu P, et al. Compressed sensing with wavelet domain dependencies for coronary MRI: a retrospective study. IEEE Trans Med Imaging 2011;30:1090-1099
49. Liu Q, Leung H. Synthesis-analysis deconvolutional network for compressed sensing. 2017 IEEE International Conference on Image Processing (ICIP), 2017:1940-1944
50. Liu Q, Wang S, Yang K, Luo J, Zhu Y, Liang D. Highly undersampled magnetic resonance image reconstruction using two-level Bregman method with dictionary updating. IEEE Trans Med Imaging 2013;32:1290-1301
51. Qu X, Hou Y, Lam F, Guo D, Zhong J, Chen Z. Magnetic resonance image reconstruction from undersampled measurements using a patch-based nonlocal operator. Med Image Anal 2014;18:843-856
52. Xiong J, Liu Q, Wang Y, Xu X. A two-stage convolutional sparse prior model for image restoration. J Vis Commun Image R 2017;48:268-280
53. He J, Liu Q, Christodoulou AG, Ma C, Lam F, Liang ZP. Accelerated High-Dimensional MR Imaging With Sparse Sampling Using Low-Rank Tensors. IEEE Trans Med Imaging 2016;35:2119-2129

54. Hammernik K, Knoll F, Sodickson D, Pock T. Learning a variational model for compressed sensing MRI reconstruction. *Proceedings of the International Society of Magnetic Resonance in Medicine (ISMRM)*, 2016:1088
55. Mardani M, Gong E, Cheng JY, et al. Deep generative adversarial networks for compressed sensing automates MRI. *arXiv preprint arXiv:1706.00051*, 2017
56. Ye JC, Han Y, Cha E. Deep convolutional framelets: a general deep learning framework for inverse problems. *SIAM J Imaging Sci* 2018;11:991-1048
57. Alain G, Bengio Y. What regularized auto-encoders learn from the data-generating distribution. *J Mach Learn Res* 2014;15:3743-3773
58. Nguyen A, Clune J, Bengio Y, Dosovitskiy A, Yosinski J. Plug & play generative networks: conditional iterative generation of images in latent space. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017: 4467-4477
59. Bigdeli SA, Zwicker M. Image restoration using autoencoding priors. *arXiv preprint arXiv:1703.09964*, 2017
60. Bigdeli SA, Zwicker M, Favaro P, Jin M. Deep mean-shift priors for image restoration. *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017:763-772
61. Kingma DP, Welling M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013
62. Rezende DJ, Mohamed S, Wierstra D. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014
63. Bishop CM. *Pattern recognition and machine learning*. New York, NY: Springer, 2006:738
64. Salimans T, Karpathy A, Chen X, Kingma DP. Pixelcnn++: improving the PixelCNN with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017
65. Larochelle H, Murray I. The neural autoregressive distribution estimator. *Proceeding International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings*, 2011:29-37
66. Grover A, Dhar M, Ermon S. Flow-GAN: combining maximum likelihood and adversarial learning in generative models. *arXiv preprint arXiv:1705.08868*, 2017
67. Dinh L, Krueger D, Bengio Y. Nice: non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014
68. Dinh L, Sohl-Dickstein J, Bengio S. Density estimation using real NVP. *arXiv preprint arXiv:1605.08803*, 2016
69. Ma X, Kong X, Zhang S, Hovy E. MaCow: masked convolutional generative flow. *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, 2019:5893-5902
70. Hoogeboom E, Berg R, Welling M. Emerging convolutions for generative normalizing flows. *arXiv preprint arXiv:1901.11137*, 2019
71. Uecker M, Hohage T, Block KT, Frahm J. Image reconstruction by regularized nonlinear inversion--joint estimation of coil sensitivities and image content. *Magn Reson Med* 2008;60:674-682
72. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. *Advances in neural information processing systems 27 (NIPS 2014)*, 2014:2672-2680