Sparse synthesis regularization with deep neural networks

Daniel Obmann
Department of Mathematics
University of Innsbruck
Technikerstraße 13
6020 Innsbruck, Austria
daniel.obmann@uibk.ac.at

Johannes Schwab
Department of Mathematics
University of Innsbruck
Technikerstraße 13
6020 Innsbruck, Austria
johannes.schwab@uibk.ac.at

Markus Haltmeier
Department of Mathematics
University of Innsbruck
Technikerstraße 13
6020 Innsbruck, Austria
markus.haltmeier@uibk.ac.at

Abstract—**We propose a sparse reconstruction framework for solving inverse problems. Opposed to existing sparse regularization techniques that are based on frame representations, we train an encoder-decoder network by including an ℓ^1-penalty. We demonstrate that the trained decoder network allows sparse signal reconstruction using thresholded encoded coefficients without losing much quality of the original image. Using the sparse synthesis prior, we propose minimizing the ℓ^1-Tikhonov functional, which is the sum of a data fitting term and the ℓ^1-norm of the synthesis coefficients, and show that it provides a regularization method.**

I. INTRODUCTION

Various applications in medical imaging, remote sensing and elsewhere require solving an inverse problems of the form

$$y = \mathbf{A}x + z \,, \qquad (\text{I.1})$$

where $\mathbf{A} \colon \mathbb{X} \to \mathbb{Y}$ is a linear operator between Hilbert spaces $\mathbb{X}$, $\mathbb{Y}$, and z is the data distortion. Inverse problems are well analyzed and several established approaches for its solution exist, including filter-based methods or variational regularization [1], [2]. In the very recent years, neural networks (NNs) and deep learning appeared as new paradigms for solving inverse problems, and demonstrate impressive performance. Several approaches have been developed, including two-step [3]–[5], variational [6], iterative [7], [8] and regularizing networks [9].

Standard deep learning approaches may lack data consistency for unknowns very different from the training images. To address this issue, in [10] a deep learning approach has been introduced where minimizers

$$x_\alpha \in \arg\min_x \|\mathbf{A}(x) - y\|_{\mathbb{Y}}^2 + \alpha\phi(\mathbf{\Psi}(x)) \qquad (\text{I.2})$$

are investigated. Here $\mathbf{\Psi} \colon \mathbb{X} \to \Xi$ is a trained NN, Ξ a Hilbert space, $\phi \colon \Xi \to [0,\infty]$, $\alpha > 0$ a regularization parameter and $\mathbf{A} \colon \mathbb{X} \to \mathbb{Y}$. The resulting reconstruction approach has been named NETT (for network Tikhonov regularization), as it is a generalized form of Tikhonov regularization using a NN as trained regularizer. For a related approach see [11]. In [10] it is shown that under reasonable conditions, the NETT yields a convergent regularization method.

In this paper, we introduce a novel deep learning approach for inverse problems that is somehow dual to (I.2). We define approximate solutions of (I.1) as $x_\mu = \mathbf{\Phi}(\xi_\mu)$, where

$$\xi_\mu \in \arg\min_\xi \|\mathbf{A}\mathbf{\Phi}(\xi) - y\|_{\mathbb{Y}}^2 + \mu\phi(\xi) \,. \qquad (\text{I.3})$$

Here $\mathbf{\Phi} \colon \Xi \to \mathbb{X}$ is a trained network, $\phi \colon \Xi \to [0,\infty]$ a penalty functional and $\mu > 0$ a regularization parameter. The NETT functional in (I.2) uses an analysis approach where the analysis coefficients $\mathbf{\Psi}(x_\alpha)$ are regular with regularity measured in smallness of ϕ. Opposed to that, (I.3) assumes regularity of the synthesis coefficients ξ_μ and is therefore a synthesis version of NETT.

In particular, we investigate the case where $\Xi = \ell^2(\Lambda)$ for some index set Λ and ϕ is a weighted ℓ^1-norm used as a sparsity prior. To construct an appropriate network, we train a (modified) tight frame U-net [12] of the form $\mathbf{\Phi} \circ \mathbf{\Psi}$ using an ℓ^1-penalty, and take the decoder part as synthesis network. We show numerically that the decoder $\mathbf{\Phi}$ allows to reconstruct the signal using sparse representations. Note that we train the network independent of any measurement-operator. As in [7] this allows one to solve any inverse problem with the same (or similar) prior assumptions in the same way without having to retrain the network. As the main theoretical result, in this paper we show that (I.3) is a convergent regularization method. Performing numerical reconstructions and comparing (I.3) with existing approaches for solving inverse problems is subject of future research.

II. PRELIMINARIES

In this section, we give some theoretical background of inverse problems. Moreover, we describe the tight frame U-net that will be used for the trained regularizer.

A. Regularization of inverse problem

The characteristic property of inverse problems is its ill-posedness, which means that the solution of $\mathbf{A}x = y$ is not unique or highly unstable with respect to data

perturbations. In order to make the signal reconstruction process stable and accurate, regularization methods have to be applied, which use a-priori knowledge about the true unknown in order to construct estimates from data (I.1) that are close to the true solution.

Variational regularization is one of the most established methods for solving inverse problems. These methods incorporate prior knowledge by choosing solutions with small value of a regularization functional. In the synthesis approach, this amounts solving (I.3), where $\mathbf{\Phi} \colon \Xi \to \mathbb{X}$ is a prescribed synthesis operator. The minimizers of (I.3) are designed to approximate ϕ-minimizing solutions of the equation $\mathbf{A}\mathbf{\Phi}(\xi) = y$, defined by

$$\begin{cases} \min & \phi(\xi) \\ \text{s.t.} & \mathbf{A}\mathbf{\Phi}(\xi) = y \,. \end{cases} \tag{II.1}$$

A frequently chosen regularizer is a weighted ℓ^1-norm, which has been proven to be useful for solving compressed sensing and other inverse problems [13]–[15]. This is the form for the regularizer we will be using in this paper.

The synthesis approach is commonly used with $\mathbf{\Phi}(\xi) = \sum_{\lambda \in \Lambda} \xi_\lambda u_\lambda$ being the synthesis operator of a frame $(u_\lambda)_\lambda$ of $\mathbb{X}$, such as a wavelet or curvelet frame or a trained dictionary [16]–[19]. In this case, $\mathbf{A}\mathbf{\Phi}$ is linear, which allows the application of the standard sparse recovery theory [2], [14]. Opposed to that, in this paper we take the synthesis operator as a trained network in which case $\mathbf{A}\mathbf{\Phi}$ is non-linear. In particular, we take the synthesis operator as decoder part of an encoder-decoder network that is trained to satisfy $\mathbf{\Phi}(\mathbf{\Psi}(x)) \simeq x$. As encoder-decoder network we use the tight frame U-net [12] which is a modification of the U-net [20] with improved reproducing capabilities.

B. Tight frame U-net

We consider the case of 2D images and denote by $\mathbb{X}_0 = \mathbb{R}^{n_0 \times c_0}$ the space at the coarsest resolution of the signal with size n_0 and c_0 channels. The tight frame U-net uses a hierarchical multi-scale representation defined recursively by

$$\mathcal{N}_{\ell+1} = \mathbf{G}_\ell \circ \left(\begin{bmatrix} \mathbf{H}_h \circ \mathbf{H}_h^\mathsf{T} \\ \mathbf{H}_d \circ \mathbf{H}_d^\mathsf{T} \\ \mathbf{H}_v \circ \mathbf{H}_v^\mathsf{T} \\ \mathbf{L} \circ \mathcal{N}_\ell \circ \mathbf{L}^\mathsf{T} \end{bmatrix} \circ \mathbf{F}_\ell, \mathrm{id} \right), \tag{II.2}$$

for $\ell \in \mathbb{N}$ and with $\mathcal{N}_0 = \mathrm{id}$. Here $\mathbf{F}_\ell \colon \mathbb{R}^{n_\ell \times c_\ell} \to \mathbb{R}^{n_\ell \times d_\ell}$ and $\mathbf{G}_\ell \colon \mathbb{R}^{n_\ell \times d_\ell} \to \colon \mathbb{R}^{n_\ell \times c_\ell}$ are convolutional layers followed by a non-linearity and id is the identity used for the bypass-connection. $\mathbf{H}_h, \mathbf{H}_v, \mathbf{H}_d$ are horizontal, vertical and diagonal high-pass filters and $\mathbf{L}$ is a low-pass filter such that the tight frame property

$$\mathbf{H}_h \mathbf{H}_h^\mathsf{T} + \mathbf{H}_v \mathbf{H}_v^\mathsf{T} + \mathbf{H}_d \mathbf{H}_d^\mathsf{T} + \mathbf{L}\mathbf{L}^\mathsf{T} = c \cdot \mathrm{id} \tag{II.3}$$

is satisfied for some $c > 0$. We define the filters by applying the tensor products HH^T, HL^T, LH^T and LL^T of

the Haar wavelet low-pass $\mathrm{L} = 2^{-1/2} [1, 1]^\mathsf{T}$ and high-pass $\mathrm{H} = 2^{-1/2} [1, -1]^\mathsf{T}$ filters separately in each channel.
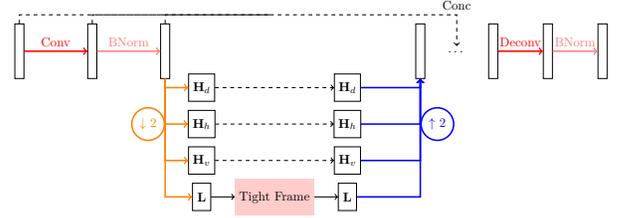
Figure II.1: **Tight frame U-net architecture.** We start by convolving the input and applying batch normalization. Then each channel is filtered using the wavelet filters, and the $\mathbf{L}$ output is recursively used as input for the next layer. After the downsampling to the coarsest resolution, we upsample by applying the transposed wavelet filters. Next we concatenate the layers and use deconvolution and batch normalization to obtain the output.

The architecture of the tight frame U-net is shown in Figure II.1. It uses standard learned convolution, batch-normalization and the fixed wavelet filters $\mathbf{H}_h, \mathbf{H}_v, \mathbf{H}_d, \mathbf{L}$ for downsampling and upsampling. To improve flexibility of the network we include an additional learned deconvolution layer after the upsampling. After every convolutional layer the ReLU activation function is applied. Similarly, we define a tight frame U-net without bypass-connection,

$$\mathcal{N}_{\ell+1} = \mathbf{G}_\ell \circ \left(\begin{bmatrix} \mathbf{H}_h \circ \mathbf{H}_h^\mathsf{T} \\ \mathbf{H}_d \circ \mathbf{H}_d^\mathsf{T} \\ \mathbf{H}_v \circ \mathbf{H}_v^\mathsf{T} \\ \mathbf{L} \circ \mathcal{N}_\ell \circ \mathbf{L}^\mathsf{T} \end{bmatrix} \circ \mathbf{F}_\ell \right), \tag{II.4}$$

for $\ell \in \mathbb{N}$ and with $\mathcal{N}_0 = \mathrm{id}$. Here $\mathbf{F}_\ell \colon \mathbb{R}^{n_\ell \times c_\ell} \to \mathbb{R}^{n_\ell \times d_\ell}$, $\mathbf{G}_\ell \colon \mathbb{R}^{n_\ell \times d_\ell} \to \colon \mathbb{R}^{n_\ell \times c_\ell}$ are convolutional layers followed by a nonlinearity, and $\mathbf{H}_h, \mathbf{H}_v, \mathbf{H}_d, \mathbf{L}$ are the wavelet filters as described above. In the rest of the paper we will refer to the network defined in (II.2) as tight frame U-net with bypass-connection, and the network defined in (II.4) as tight frame U-net without bypass-connection.

The tight frame property (II.3) allows the networks (II.2) and (II.4) to both have the perfect recovery condition which means that filters $\mathbf{F}_\ell, \mathbf{G}_\ell$ can be chosen such that any signal $x \in \mathbb{X}$ can be perfectly recovered from its frame coefficients if they are given in all layers [12]. In the following we will refer to the results after convolving an image $x_\ell \in \mathbb{X}_\ell = \mathbb{R}^{n_\ell \times c_\ell}$ with the fixed wavelet filters as filtered version of x_ℓ.

III. Nonlinear sparse synthesis regularization

To solve the inverse problem (I.1), we use the sparse synthesis NETT which considers minimizers of

$$\mathcal{S}_{\mu,y}(\xi) \triangleq \|\mathbf{A}\mathbf{\Phi}(\xi) - y\|_{\mathbb{Y}}^2 + \mu \sum_{\lambda \in \Lambda} w_\lambda |\xi_\lambda| \,. \tag{III.1}$$

Here $\mathbf{\Phi} \colon \ell^2(\Lambda) \to \mathbb{X}$ is the synthesis operator, Λ an index set and w_λ are positive parameters.

A. Theoretical analysis

The sparse synthesis NETT can be seen as weighted ℓ^1-regularization for the coefficient inverse problem $\mathbf{A}\mathbf{\Phi}(\xi) = y$. For its theoretical analysis we require the following

(A1) $\mathbf{A} \colon \mathbb{X} \to \mathbb{Y}$ is bounded linear;
(A2) $\mathbf{\Phi} \colon \ell^2(\Lambda) \to \mathbb{X}$ is weakly continuous;
(A3) $w_{\min} \triangleq \inf\{w_\lambda \mid \lambda \in \Lambda\} > 0$.

We then have the following result:

Theorem III.1 (Well-posedness). *Under assumptions (A1)-(A3) the following holds:*

- EXISTENCE: *For all $y \in Y$, $\mu > 0$, the functional in (III.1) has a minimizer*
- STABILITY: *Suppose $y_k \to y$ and $\xi_k \in \arg\min \mathcal{S}_{\mu, y_k}$. Then weak accumulation points of $(\xi_k)_{k \in \mathbb{N}}$ exist and are minimizers of $\mathcal{S}_{\mu, y}$.*

Proof. According to (A1), (A2), the operator $\mathbf{A}\mathbf{\Phi}$ is weakly continuous. Therefore, the results are a direct consequence of [2, Theorem 3.48]. □

From [2, Theorem 3.48, Theorem 3.49] we can further deduce convergence (as the noise level goes to zero) of the sparse synthesis NETT. Later we take $\mathbf{\Phi}$ as decoder part of a tight frame U-net trained as an auto-encoder, which we expect to be weakly continuous and Lipschitz continuous. In this case, we have stability and convergence for the actual reconstruction $\mathbf{\Phi}(\xi_\mu)$.

B. A trained sparse regularizer

Using a similar architecture to the one suggested in [12], we train a model for sparse regularization. To enforce sparsity in the encoded domain we will use a combination of mean-squared-error and an ℓ^1-penalty of the filtered coefficients as loss-function for training purposes. The idea is to enforce the sparsity in the high-pass filtered images. To achieve this, we will regularize these images in the encoded domain using a regularization parameter depending on the layer.

We write the tight frame U-net defined by (II.2) in the form $\mathbf{\Phi}_\eta \circ \mathbf{\Psi}_\theta$ where $\mathbf{\Psi}_\theta$ is the encoder and $\mathbf{\Phi}_\eta$ the decoder part. Moreover, we denote by $\mathbf{\Psi}_{\theta;a}^\ell(x)$ for $a \in \{h, v, d\}$ the high-pass filter coefficients of $x \in \mathbb{X}$ in the ℓth layer. Given training data $x_1, \ldots, x_N$, the loss-function used for network training is taken as

$$E(\theta, \eta) = \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{\Phi}_\eta \circ \mathbf{\Psi}_\theta(x_i) - x_i\|_2^2$$

$$+ \frac{\mu}{N} \sum_{i=1}^{N} \sum_{\ell \in \mathbb{N}} \sum_{a \in \{h, v, d\}} w_\ell \|\mathbf{\Psi}_{\theta;a}^\ell(x_i)\|_1 . \quad (\text{III.2})$$

The first term of the loss-function is supposed to enforce the network to reproduce the training images. Following the sparse regularization strategy, the second term forces the network to learn convolutions such that high-pass filtered coefficients are sparse.

IV. NUMERICAL EXPERIMENTS

The above sparse encoding strategy has been tested with the two network architectures described in (II.2) and (II.4). Both networks are tested for their reconstruction capabilities when setting parts of the frame coefficients to zero. Actual application to the solution of tomographic inverse problems is subject of future research.
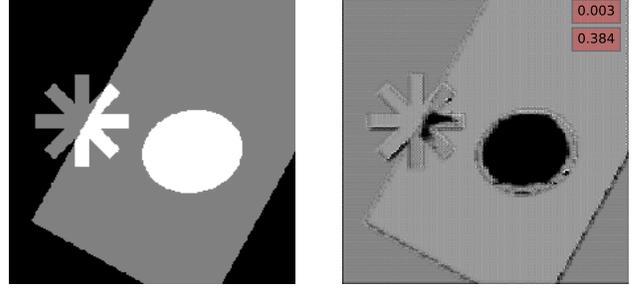
Figure IV.1: **Test phantom and influence of the bypass connection.** Top left: original image. Top right: reconstructed image using the network with bypass-connection and setting the bypass-coefficients to 0. The first number depicted in the right image is the image distance described in (IV) and the second one is the SSIM.

A. Implementation details

For the numerical experiments, we generated 256×256 grayscale images which contain an ellipse, a rectangle and a star-like shape. Each of the shapes parameter has been chosen randomly. The training dataset consists of 1500 and the validation dataset of 500 such images. One of the phantoms from the training set is shown in Figure IV.1 (top left). The top right image shows the reconstruction using the tight frame U-net trained with the bypass-connection after setting the bypass-coefficients to zero. The large difference between these two images shows that the bypass-connection significantly contributes to the image representation and reconstruction. Since the wavelet filters have not been applied to the bypass-connection, one cannot expect sparsity for this part. This is actually the reason why we expect the tight frame U-net without bypass-connection to allow much sparser approximation than the tight frame U-net with bypass-connection. This conjecture is supported by the numerical results presented below.

Each of the networks has 3 downsampling- and upsampling-layers and starts with 8 channels for the first convolution. The number of channels is then doubled in each consequent layer. For minimizing the loss-function $E(\theta, \eta)$ w.r.t θ and η we use the Adam [21] algorithm with the suggested parameters and train each network for 60 epochs. For the experiments we chose the regularization parameters $\mu = 10^{-9.5} \cdot N$ where N is the number of trainings-samples and $w_\ell = 2^{-\ell}$. The training was done using an Intel Xeon CPU E331225 @3.10 GHz processor

and 16 GB RAM. Each epoch (including the evaluation on the validation set) took about 30 min for the tight frame U-net with bypass-connection and about 20 min minutes for the tight frame U-net without bypass-connection. This results in a training-time of 30 h and 20 h, respectively. Note that the training time could be reduced significantly by using GPUs for less than 1000 € instead of the CPU.
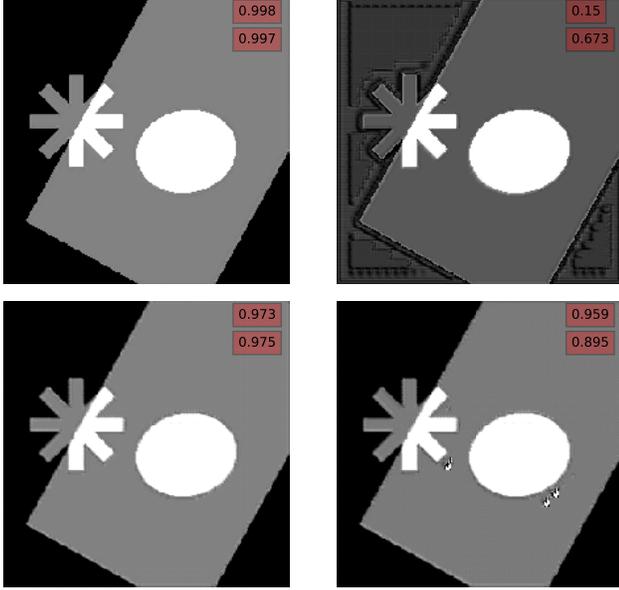
Figure IV.2: **Sparse recovery results.** Top: passing the image through the tight frame U-net with bypass-connection (left) and corresponding reconstruction after setting 85 % of the coefficients to 0 (right). Bottom: passing the image through the tight frame U-net without bypass-connection (left) and corresponding reconstruction after setting 85 % of the coefficients to 0 (right).

B. Sparse approximation results

Each of the two tight frame U-nets has been tested on its ability to reconstruct the image from a sparse approximation in the encoded domain. To this end, we calculated the frame coefficients of the test image using the encoder part of the network, and set a certain fraction $p \in [0, 1]$ of the coefficients in each channel with smallest absolute value to 0. The decoder is then applied to the thresholded coefficients to get a sparse approximation of the original image. In Figure IV.2, example reconstructions using all coefficients (left) and thresholded coefficients with a value of $p = 0.85$ (right) are shown. We observe that both tight frame U-net variants yield almost perfect recovery when using the original coefficients. However, as expected, when applied to the thresholded coefficients, the network without bypass-connection (bottom) yields significantly better results.

To quantitatively evaluate the reconstructed images, we compute the structural similarity index (SSIM), the peak-signal-to-noise-ratio (PSNR) and the image distance (ID), defined by $\mathrm{ID}_\varepsilon(x, \hat{x}) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{[0,\varepsilon]}(|x_i - \hat{x}_i|)$ with $\varepsilon = 1/256$, meaning that entries differing by less than one pixel are considered equal. To evaluate the sparse approximation capabilities of the two models we calculate ratios of the evaluation metrics between the reconstructions with the thresholded and the original coefficients, respectively. In these evaluation metrics, a high (close to 1) ratio indicates good performance.
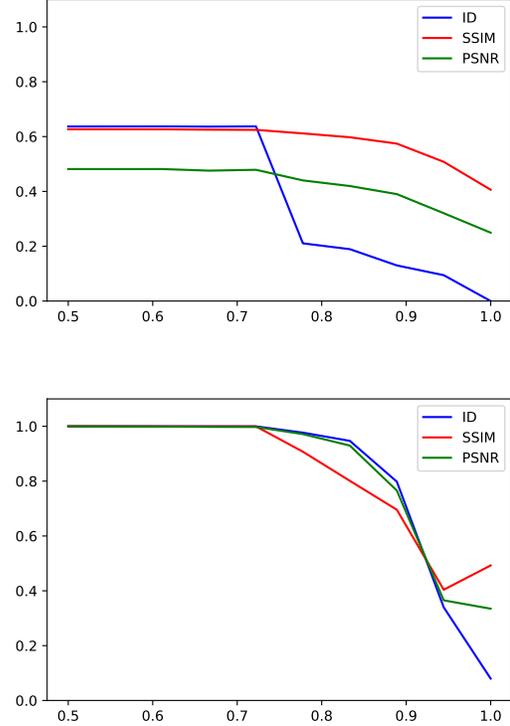
Figure IV.3: **Ratios of ID, SSIM and PSNR scores depending on the thresholding level.** Top: Network with bypass-connection. Bottom: Network without bypass-connection. Because of the inherent sparsity of the image, we decided to only measure the quality of the reconstruction for a thresholding level of $p \geq 0.5$.

C. Discussion

The reconstruction results in Figure IV.2 show the sparse approximation results using the tight frame U-net with and without bypass-connection. The network with bypass-connection is able to almost perfectly recover the image from all frame coefficients (top left). However, when thresholding 85 % of the coefficients, this is no longer the case (top right). The bottom left image shows the image passed through the network without bypass-connection. Comparing this to the pass through the network with bypass-connection we see that the network without bypass-connection, when using all coefficients,

performs slightly worse. However, when thresholding 85 % of the coefficients obtained by passing the image through the encoder part, the network without bypass-connection significantly outperforms the one with bypass-connection.

To further investigate this issue, we sample images from the validation set and plot the mean of the ratios of the metric scores when setting various percentages of coefficients to zero (Figure IV.3). As a base for this we take the metric scores obtained by passing the images through the network. Because of the inherent sparsity of the images we chose to plot these metrics only for $p \geq 0.5$. When comparing the two plots in Figure IV.3 we see that the network without bypass-connection can almost maintain the metric scores up to some point at $p \simeq 0.85$, whereas the network with bypass-connection falls off right at the beginning and tends to perform worse than the network without bypass-connection.

V. Conclusion

In this paper we proposed a sparse regularization strategy using a neural network as synthesis operator. The network is used as a nonlinear transformation between the image space and a coefficient space used for signal representation. In particular, we used an encoder-decoder pair of a tight frame U-Net trained with an ℓ^1-penalty for signal representation in the coefficient space. To numerically investigate the sparse approximation capabilities, we set some of the encoded coefficients to zero before applying the decoder. Our numerical results suggests that the tight frame U-net without bypass-connection enables sparse recovery. Actual implementation of our approach to tomographic inverse problems and detailed comparison with other established reconstruction methods is subject of future research. We point out that the learned part of our proposed regularization approach only depends on the class of images to be (re-)constructed which allows us to apply the same network to any inverse problem targeting a similar class of phantoms, without having to retrain the network.

References

[1] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of inverse problems*, ser. Mathematics and its Applications. Dordrecht: Kluwer Academic Publishers Group, 1996, vol. 375.

[2] O. Scherzer, M. Grasmair, H. Grossauer, M. Haltmeier, and F. Lenzen, *Variational methods in imaging.* Springer, 2009.

[3] D. Lee, J. Yoo, and J. C. Ye, "Deep residual learning for compressed sensing MRI," in *IEEE 14th International Symposium on Biomedical Imaging*, 2017, pp. 15–18.

[4] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Process.*, vol. 26, pp. 4509–4522, 2017.

[5] S. Antholzer, M. Haltmeier, and J. Schwab, "Deep learning for photoacoustic tomography from sparse data," *Inverse Probl. Sci. and Eng.*, vol. in press, pp. 1–19, 2018.

[6] E. Kobler, T. Klatzer, K. Hammernik, and T. Pock, "Variational networks: connecting variational methods and deep learning," in *German Conference on Pattern Recognition.* Springer, 2017, pp. 281–293.

[7] J. R. Chang, C.-L. Li, B. Poczos, and B. V. Kumar, "One network to solve them all–solving linear inverse problems using deep projection models," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5889–5898.

[8] J. Adler and O. Öktem, "Solving ill-posed inverse problems using iterative deep neural networks," *Inverse Probl.*, vol. 33, p. 124007, 2017.

[9] J. Schwab, S. Antholzer, and M. Haltmeier, "Deep null space learning for inverse problems: convergence analysis and rates," *Inverse Probl.*, vol. 35, p. 025008, 2019.

[10] H. Li, J. Schwab, S. Antholzer, and M. Haltmeier, "NETT: Solving inverse problems with deep neural networks," *arXiv:1803.00092*, 2018.

[11] S. Lunz, C. Schoenlieb, and O. Öktem, "Adversarial regularizers in inverse problems," in *Advances in Neural Information Processing Systems*, 2018, pp. 8507–8516.

[12] Y. Han and J. C. Ye, "Framing U-Net via deep convolutional framelets: Application to sparse-view CT," *IEEE Trans. Med. Imag.*, vol. 37, pp. 1418–1429, 2018.

[13] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, pp. 21–30, 2008.

[14] M. Grasmair, M. Haltmeier, and O. Scherzer, "Necessary and sufficient conditions for linear convergence of ℓ^1-regularization," *Comm. Pure Appl. Math.*, vol. 64, pp. 161–182, 2011.

[15] M. Haltmeier, "Stable signal reconstruction via ℓ_1-minimization in redundant, non-tight frames," *IEEE Trans. Signal Process.*, vol. 61, pp. 420–426, 2013.

[16] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math.*, vol. 57, pp. 1413–1457, 2004.

[17] E. J. Candès and D. Donoho, "Recovering edges in ill-posed inverse problems: Optimality of curvelet frames," *Ann. Stat.*, vol. 30, pp. 784–842, 2002.

[18] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Proc.*, vol. 54, pp. 4311–4322, 2006.

[19] R. Gribonval and K. Schnass, "Dictionary identification – sparse matrix-factorization via ℓ_1-minimization," *IEEE Trans. Inf. Theory*, vol. 56, pp. 3523–3539, 2010.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention.* Springer, 2015, pp. 234–241.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.