INFORMATICA, 2020, Vol. 31, No. 3, 561–578

© 2020 *Vilnius University*
DOI: https://doi.org/10.15388/20-INFOR421

Perceptual Autoencoder for Compressive Sensing Image Reconstruction

Ivan RALAŠIĆ*, Damir SERŠIĆ, Siniša ŠEGVIĆ

University of Zagreb, Faculty of Electrical Engineering and Computing, Unska 3, Zagreb, HR-10000, Croatia
e-mail: ivan.ralasic@fer.hr

Received: August 2019; accepted: May 2020

Abstract. This paper presents a non-iterative deep learning approach to compressive sensing (CS) image reconstruction using a convolutional autoencoder and a residual learning network. An efficient measurement design is proposed in order to enable training of the compressive sensing models on normalized and mean-centred measurements, along with a practical network initialization method based on principal component analysis (PCA). Finally, perceptual residual learning is proposed in order to obtain semantically informative image reconstructions along with high pixel-wise reconstruction accuracy at low measurement rates.

Key words: compressive sensing, convolutional autoencoder, deep learning, image reconstruction, perceptual loss, principal component analysis.

1. Introduction

Compressive sensing (CS) is a signal processing technique that enables accurate signal recovery from an incomplete set of measurements (Candes and Tao, 2006; Baraniuk, 2007; Duarte and Eldar, 2011; Duarte and Baraniuk, 2012):

$$\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\epsilon}, \tag{1}$$

where $\boldsymbol{\Phi}$ is an $M \times N$ measurement matrix, $\boldsymbol{y} \in \mathbb{R}^M$ is a set of M measurements (where M can be much smaller than the original dimensionality of the signal N), and $\boldsymbol{\epsilon}$ is measurement noise. Efficient signal recovery is possible even in the case when the number of the acquired measurements is far below the Shannon-Nyquist limit.

The CS reconstruction process can be observed as a linear inverse problem that occurs in numerous image processing tasks such as inpainting (Bertalmio *et al.*, 2000; Bugeau *et al.*, 2010), super-resolution (Yang *et al.*, 2010; Dong *et al.*, 2016), and denoising (Elad and Aharon, 2006). In order to reconstruct the signal $\boldsymbol{x}$ from a set of measurements $\boldsymbol{y}$, one has to solve the underdetermined (i.e. $M < N$) system of linear equations in Eq. (1). In the CS literature, the ratio $r = {}^{M}/_{N}$ is called the CS measurement rate. In order to recover the

*Corresponding author.

signal $\boldsymbol{x}$ from its low dimensional measurements, it is necessary to use a signal prior that enables the identification of a true solution from an infinite set of feasible solutions. This is usually done by introducing a regularization term to an existing loss function. Usually, the l_0 norm, or its convex relaxation, the l_1 norm, is used as the regularizer under the assumption that the observed signal is sparse in certain transformation domain $\boldsymbol{\Psi}$:

$$\boldsymbol{s} = \boldsymbol{\Psi}\boldsymbol{x}, \tag{2}$$

where $\boldsymbol{s}$ denotes the sparse representation of the signal $\boldsymbol{x}$. Other signal priors can be used as regularizers as well. An unconstrained optimization problem for the sparse signal recovery using l_1 regularization can be written as:

$$\min_{\boldsymbol{s}} \left\| \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{\Psi}^{-1}\boldsymbol{s} \right\|_2^2 + \lambda \|\boldsymbol{s}\|_1. \tag{3}$$

Most of the algorithms for solving sparse optimization problems are iterative and have high computational complexity (Mallat and Zhifeng, 2006; Pati *et al.*, 1993; Needell and Tropp, 2009; Beck and Teboulle, 2009; Becker *et al.*, 2011; Wright *et al.*, 2009). This presents a serious drawback when it comes to the real-world applications of CS.

After being successfully applied to numerous previously mentioned image processing tasks, machine learning methods started to gain more interest in the area of CS (Mousavi *et al.*, 2015; Mousavi and Baraniuk, 2017; Mousavi *et al.*, 2017; Kulkarni *et al.*, 2016; Hantao *et al.*, 2019; Lohit *et al.*, 2018). Novel CS reconstruction algorithms based on deep neural networks have recently been proposed, and they represent a non-iterative, fast and efficient alternative to the traditional CS reconstruction algorithms.

2. Related Work

A deep learning framework based on the stacked denoising autoencoder (SDA) has been proposed in Mousavi *et al.* (2015) and it represents pioneer work in the area of CS reconstruction using the learning-based approach. The main drawback of the SDA approach is that the network consists of fully-connected layers, which means that all units in two consecutive layers are connected to each other. Thus, as the signal size increases, so does the computational complexity of the neural network. Authors present an extension of their previous work in Mousavi and Baraniuk (2017) and Mousavi *et al.* (2017). The DeepInverse network proposed in Mousavi and Baraniuk (2017) solves the image dimensionality problem by using the adjoint operator $\boldsymbol{\Phi}^T$ to initialize the weights of the fully connected reconstruction layer. In Mousavi *et al.* (2017), a non-linear measurement operator is trained to learn a transformation from the original signal space to an undersampled measurement space. A novel class of convolutional neural networks (CNN) architectures inspired by the work of Dong *et al.* (2016) was proposed in Kulkarni *et al.* (2016). The proposed CNN takes image block CS measurements as inputs and outputs a block reconstruction obtained from low-dimensional measurements. Improved ReconNet was proposed in Lohit *et al.* (2018), where the authors use adversarial loss to further improve the CS reconstruction

results. Moreover, the authors add a linear fully connected layer to the existing ReconNet architecture and learn the optimal measurement and reconstruction matrix in a single network. Based on their initial work in Xie *et al.* (2017) and Du *et al.* (2019), the authors propose to train the neural network using perceptual loss in Du *et al.* (2018). Perceptual loss (Johnson *et al.*, 2016) is defined in the latent space of a secondary network and helps to preserve higher level information when compared to the commonly used per-pixel Euclidean loss. In Hantao *et al.* (2019), the authors propose a novel *Deep Residual Reconstruction Network* (DR^2-Net) to restore the image from its blockwise CS measurements with an additional residual layer that enhances the preliminary image reconstruction.

In this paper, we propose an efficient deep learning model for CS acquisition and reconstruction. Our model is based on a fully convolutional autoencoder with a residual network. Fully convolutional architecture alleviates the signal dimensionality problems that occur in the full-connected network design (Mousavi *et al.*, 2015). Disadvantage of using the fully convolutional architecture is that it is not directly applicable to certain imaging modalities where the measurements correspond to the whole signal, and one cannot perform measurements in a blockwise manner. In contrast to Mousavi *et al.* (2017) where the authors propose to learn a non-linear measurement operator in their *DeepCodec* network, we use a linear encoding part while the non-linearities are introduced only into the residual learning network. Motivation for this is to ensure that the learned measurement operator is implementable in the real-world CS measurement systems which are mostly linear. The residual network improves the initial image reconstruction and removes eventual reconstruction artifacts.

Although it is well known that normalization of the training data significantly speeds up the training procedure (Ioffe and Szegedy, 2015), applying the measurement normalization in the learning-based CS is not straightforward. In order to normalize and mean-centre the CS measurements, measurement process has to be redesigned. Mean values of the observed image blocks have to be known in order to perform mean-centring and normalization. Therefore, we dedicate a single measurement vector to measure the mean value of the observed image block. The rest of the measurement matrix is optimized in the training process. Input to the decoding part of the proposed model are mean-centred measurements, and the decoding process results in mean-centred image reconstruction. Mean value for the observed block is then added to the initial image estimate to obtain the final image reconstruction. Without the proposed modifications of the measurement process, performing reconstruction on normalized measurements would not be feasible. As expected, we show that the measurement normalization process speeds up the convergence of the network significantly.

Furthermore, we discuss the connection between the linear autoencoder network and principal component analysis (PCA). Based on our observations, an efficient method for initialization of the network weights is proposed. The proposed method serves as a bootstraping step in the network training procedure. Instead of initializing the model using random weights, we propose to use an educated guess for the initial weights by using the PCA initialization method.

Finally, we introduce perceptual loss in the residual network training in order to improve the reconstructions at extremely low measurement rates. Experimental results ob-

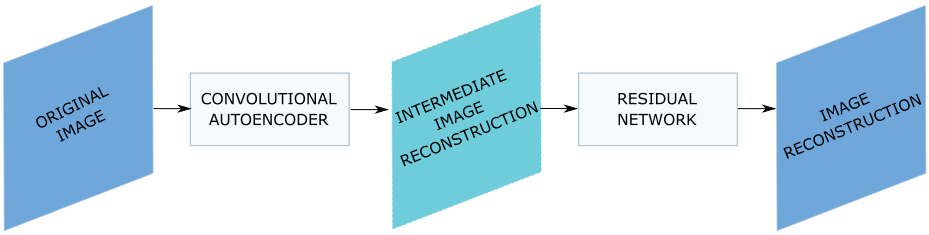

Fig. 1. Proposed design of the CS image reconstruction model. The convolutional autoencoder learns the end-to-end CS mapping. The encoder performs synthetic measurements on the input image, transforming it into the low-dimensional measurement space. The decoding part learns the optimal inverse mapping from the low-dimensional measurements into the intermediate image reconstruction. The residual network additionally improves the initial image reconstruction.

tained using the proposed model show improvements in terms of the reconstruction quality.

The paper is organized as follows: in Section 3.1, convolutional autoencoder for CS image reconstruction is proposed. Section 3.2 and Section 3.4 offer a discussion on the measurement matrix optimality and efficient network initialization. Section 3.3 introduces the normalized measurement process. Finally, perceptual residual learning is introduced in Section 3.5 in order to improve the image reconstructions obtained by the autoencoder. Section 4 presents the main results with discussion, while Section 6 offers the conclusion.

3. Proposed Architecture for the CS Model

3.1. *Convolutional Autoencoder*

The encoding part of the proposed shallow autoencoder network performs the CS measurement process on an input image, while the decoding part models the CS reconstruction process and reconstructs the input image from the low-dimensional measurement space (Fig. 1).

In the traditional CS measurement process, an image is vectorized to form a one-dimensional vector $\boldsymbol{x} \in \mathbb{R}^N$ and is projected into a low-dimensional measurement vector $\boldsymbol{y} \in \mathbb{R}^M$ using an inner product with a collection of measurement vectors $\{\boldsymbol{\phi}_m\}_{m=1}^M$:

$$y_m = \langle \boldsymbol{\phi}_m, \boldsymbol{x} \rangle = \sum_{i=1}^{N} \phi_{m,i} x_i . \tag{4}$$

The measurement matrix $\boldsymbol{\Phi}$ is created by arranging the measurement vectors $\boldsymbol{\phi}_m^T$ as rows. Signal dimensionality (i.e. image dimensions) determines the number of columns in the measurement matrix. Consequently, when image dimensions are large, a block-based CS approach is suitable since it operates on local image patches (Du *et al.*, 2012). The block-based CS results in a lower computational complexity and requires less memory to store the measurement matrix.

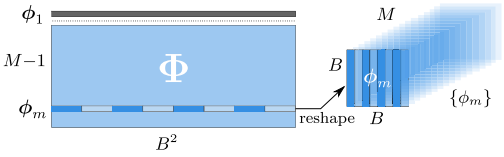

Fig. 2. Creating a set of measurement filters from the measurement matrix. Row vector $\boldsymbol{\phi}_m$ is reshaped column-wise to create a measurement filter ϕ_m. The first row vector $\boldsymbol{\phi}_1$ of the measurement matrix is kept fixed during the training and corresponds to the measurement vector that calculates the mean value of the observed block. The measurement matrix $\boldsymbol{\Phi}$ has $M-1$ rows that are optimized. The collection of measurement filters $\{\phi_m\}$ has a depth size of M (i.e. $M-1$ trainable filters and one fixed filter).

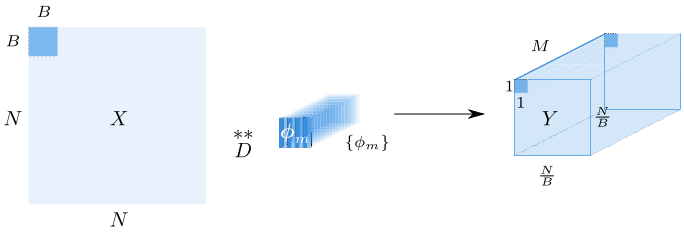

Fig. 3. Visualization of the measurement process using decimated 2D convolution. Block x of size $B \times B$ from the whole image X of size $N \times N$ is convolved with a collection of measurement filters $\{\phi_m\}$ of size $B \times B \times M$. This results in a measurement tensor $\boldsymbol{y}$ of size $1 \times 1 \times M$. A set of measurement tensors is denoted by Y and has a size of $\frac{N}{B} \times \frac{N}{B} \times M$.

In this paper, a linear convolutional layer performs decimated convolution, as in Eq. (5), in order to obtain the measurements. Convolution can be used as an extension of the inner product in which the inner product is computed repeatedly over the image space.

$$\begin{aligned} Y &= X \underset{D}{**} \{\phi_m\}, \\ Y_m[i,\, j] &= \sum_k \sum_l X[Di + k,\, Dj + k]\phi_m[k, l]. \end{aligned} \tag{5}$$

In Eq. (5), decimation factor D equals the size of the block B and the double asterisk $(\underset{D}{**})$ denotes a 2D convolutional operator decimated with the same factor. A two-dimensional measurement filter $\boldsymbol{\phi}_m$ is created column-wise from the measurement vector $\boldsymbol{\phi}_m$ as shown in Fig. 2. In Eq. (5), Y denotes all the measurements obtained using decimated convolution over the whole input image X with the collection of measurement filters $\{\phi_m\}$. A visualization of the measurement process modelled using 2D convolution is shown in Fig. 3.

The CS reconstruction process is modelled using a transposed convolution (Dumoulin and Visin, 2016), and the decoding part of the autoencoder is trained to learn the optimal pseudo-inverse linear mapping operator $\boldsymbol{\Phi}^+$ from the measurement data.

3.2. *Predefined vs. Adaptive Measurement Matrix*

There are two basic approaches for the measurement matrix design. An arbitrary measurement matrix $\mathbf{\Phi}$ can be used in the measurement process to obtain measurements $\boldsymbol{y}$ from the input images. In the traditional CS, the measurement matrix with independent and identically distributed (i.i.d.) Gaussian measurement vectors is often used. In that case, the encoding layer of the autoencoder is initialized using the weights defined by the vectors from the measurement matrix $\mathbf{\Phi}$ and is kept fixed during the training process. A signal dimensionality reduction using a predefined (e.q. random Gaussian, Hadamard, DCT) measurement matrix $\mathbf{\Phi}$ is sub-optimal due to the fact that it does not exploit the underlying structure of the observed signal.

Alternatively, the optimal measurement matrix can be inferred from the training data. Such a matrix better adapts to the dataset and preserves more information in the measurements, resulting in better reconstruction results. In our proposal, we optimize the encoding part of the autoencoder to learn the optimal linear measurement matrix $\mathbf{\Phi}$ from the training dataset. In the experimental section, we show the effect of the measurement matrix choice on the reconstruction results.

3.3. *Network Training Using Normalized Measurements*

Training neural networks on normalized, mean-centred data became standard in all areas of machine learning (Ioffe and Szegedy, 2015). It is well known that such practice significantly reduces the training time, but the application to the learning based CS is not straightforward. The measurement process needs to be redesigned in order to obtain normalized and mean-centred measurements, since the mean value of the observed signal has to be measured during the CS acquisition process. In this section, we present an efficient measurement process which enables the direct application of data normalization techniques, which is in contrast with the previous work in this area.

In order to measure the mean value y_1 of the observed block (Fig. 2), we fix the first row of the measurement matrix $\mathbf{\Phi}$, so that it corresponds to a row vector containing all ones:

$$y_1 = \frac{1}{B^2}\sum_{i=1}^{B^2}\phi_{1,i}x_i = \frac{1}{B^2}\sum_{i=1}^{B^2}x_i. \tag{6}$$

The rest of the matrix ($M-1$ rows) is left to be optimized in the training procedure. We mean-centre the normalized measurements y_m using the obtained mean measurement y_1:

$$\hat{y}_m = \frac{1}{\sum_{i=1}^{B^2}\phi_{m,i}}y_m - y_1, \quad m \in [2, M]. \tag{7}$$

The decoding part of the network is trained using the mean-centred measurement vector $\hat{\boldsymbol{y}}$ as its input, and it results in the mean-centred image reconstruction. In order to obtain the

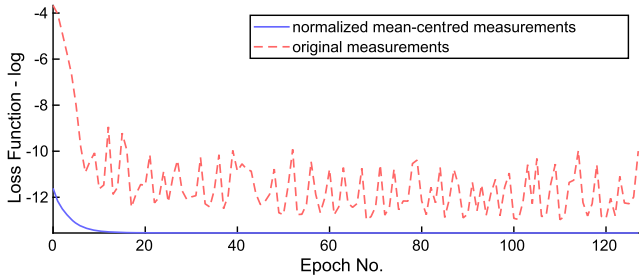

Fig. 4. Training loss function. Normalized mean-centred measurements vs. original measurements. Notice the zig-zagging in the loss function when using non-centred measurement data. Loss functions are visualized on the log scale.

final image reconstruction, the mean value for each image block is restored by adding y_1 to each block.

Training the neural network on non-mean-centred data has undesirable consequences. If the data coming into a neuron is always positive (e.q. $x > 0$ element-wise in $f = w^T x + b$), then the gradient on the weights w becomes either all-positive, or all-negative (depending on the gradient of the whole expression f) during the back-propagation step. In return, this could introduce the undesirable zig-zagging dynamics in the gradient updates of the weights (Karpathy, 2017). As shown in Fig. 4, zig-zagging is also manifested in the loss function. The training loss function for the unnormalized measurements (red dashed line) and normalized measurements (blue solid line) are shown in the *log* scale simultaneously. Notice that the loss function for the proposed network that is trained on mean-centred data converges significantly faster than the network trained on non-centred measurements.

3.4. *Efficient Method for Network Initialization*

In Lohit *et al.* (2018) and Du *et al.* (2019), the authors optimize the linear encoder in order to infer the optimal measurement matrix for each measurement rate r. In Baldi and Hornik (1989), it has been shown that the linear autoencoder with the mean squared error (MSE) loss converges to a unique minimum corresponding to the projection onto the subspace generated by the first principal component vectors of the covariance matrix obtained using the principal component analysis (PCA). Thus, it is sub-optimal to retrain the model for each measurement rate r.

Instead, we propose an efficient initialization method for the deep learning CS models based on the observation from Baldi and Hornik (1989). Principal component analysis (PCA) is an analytic method that has a widespread use in dimensionality reduction. The PCA is performed on the covariance matrix of the data vector $\boldsymbol{x}$:

$$C(\boldsymbol{x}) = E\left[\boldsymbol{x}\boldsymbol{x}^T\right] - E[\boldsymbol{x}]E\left[\boldsymbol{x}^T\right], \tag{8}$$

where E denotes the expectation operator. In the case when images are the signals of interest, PCA is performed by calculating an unbiased estimate of the covariance matrix

$C(\boldsymbol{x})$ for the vectorized images, where $\boldsymbol{x}$ is a flattened image vector, and $\bar{\boldsymbol{x}}$ is its mean value:

$$C(\boldsymbol{x}) = \frac{1}{N-1}\sum_{i=1}^{N}(\boldsymbol{x}_n - \bar{\boldsymbol{x}})(\boldsymbol{x}_n - \bar{\boldsymbol{x}})^T. \tag{9}$$

After applying the eigendecomposition (Eq. (10)) to the estimate of the covariance matrix $C(\boldsymbol{x})$ for the observed images, an eigenvalue matrix $\boldsymbol{\Sigma}$ contains positive eigenvalues λ sorted in a descending order. The eigenvalues explain the variance in the direction of corresponding eigenvector in the orthonormal matrix $\boldsymbol{U}$. Under the assumption that the variance reflects the informational content, a subset of M eigenvectors with the largest eigenvalues (i.e. principal components) optimally describes the observed signal in terms of the mean squared error:

$$C(\boldsymbol{x}) = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}^T \approx \boldsymbol{U}_{1:M}\boldsymbol{\Sigma}_{1:M}(\boldsymbol{U}_{1:M})^T. \tag{10}$$

If the training dataset is formed to faithfully represent the image statistics, the reduced eigenvector matrix $\boldsymbol{U}_{1:M}^T$ optimally preserves the informational content of the observed image blocks.

Thus, we propose to use the reduced eigenvector matrix $\boldsymbol{U}_{1:M}^T$ to initialize the weights of the encoding part of the CS model:

$$\boldsymbol{\Phi} = \boldsymbol{U}_{1:M}^T. \tag{11}$$

Furthermore, we propose to initialize the reconstruction part of the network using the PCA as well. The eigenvector matrix $\boldsymbol{U}$ is a unitary matrix. If the measurement matrix $\boldsymbol{\Phi}$ is equal to the reduced eigenvector matrix $\boldsymbol{U}_{1:M}^T$ as in our proposal, we can write:

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{x} = \boldsymbol{U}_{1:M}^T\boldsymbol{x}. \tag{12}$$

The original image $\boldsymbol{x}$ can be reconstructed using the pseudo-inverse of the measurement matrix:

$$\begin{aligned}\boldsymbol{x} &= \boldsymbol{\Phi}^+\boldsymbol{y}\\ &= \left(\boldsymbol{\Phi}\boldsymbol{\Phi}^T\right)^{-1}\boldsymbol{\Phi}\boldsymbol{y}\\ &= \left[(\boldsymbol{U}_{1:M})^T\boldsymbol{U}_{\mathbf{1:M}}\right]^{-1}(\boldsymbol{U}_{1:M})^T\boldsymbol{y}\\ &= \boldsymbol{U}_{1:M}\boldsymbol{y}.\end{aligned} \tag{13}$$

Since $\boldsymbol{U}^T$ is a unitary matrix, the pseudo-inverse matrix $\boldsymbol{\Phi}^+$ for the CS reconstruction becomes just a transposition of the measurement matrix. This results in an efficient method for initialization of neural network weights for both the encoding and decoding part of the learning based CS models.

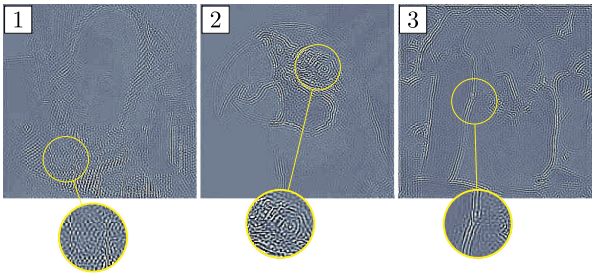

Fig. 5. Contrast-adjusted visualization of the learned residual for several test images and for the measurement ratio $r = 0.25$: 1) *Barbara*, 2) *Parrot*, 3) *Peppers*. Notice that the residual network improves the preliminary reconstructions in aspects of blocking artifacts, high frequency content restoration and edge preservation.

The proposed initialization method for the network weights has several advantages. While a neural network has to be retrained in order to obtain the measurement matrix $\mathbf{\Phi}$ for a different sub-rate r, the PCA approach outputs the whole eigenvector matrix $\boldsymbol{U}$. Thus, for any measurement rate the initial measurement matrix $\mathbf{\Phi}$ can be formed by selecting a subset of M largest eigenvectors and one can use them in order to initialize the model. The learning based approach is significantly slower since it is extremely hard to learn the optimal measurement operator and the network might not fully converge. Contrary, in the case of linear autoencoder, we obtain the exact solution for optimal measurement and reconstruction operator in a fraction of time needed to train the neural network. Using the PCA initialization for the autoencoder might be beneficial even when the loss function in the training procedure is not pixel-wise Euclidean and when additional regularization is introduced in the training procedure.

3.5. *Residual Network*

As previously mentioned, the first part of the proposed network consists of a linear autoencoder. Non-linearities can be easily introduced into the measurement and reconstruction part of the network to further improve the initial reconstruction obtained by the autoencoder. In our proposal, non-linearities are only introduced into the decoding part of the network. Although there are some methods that learn a non-linear measurement operator from the data (Mousavi *et al.*, 2017), linearity is an important property of measurement systems and we want our CS model to be realizable in real physical measurement setups like Takhar *et al.* (2006) and Ralašić *et al.* (2018).

The output of the proposed convolutional autoencoder represents a preliminary reconstruction of the input image from its low-dimensional measurements. We feed the preliminary reconstruction to a residual network (He *et al.*, 2015) that induces non-linearity and reduces potential reconstruction and blocking artifacts, and eliminates the need for an off-the-shelf denoiser such as BM3D (Dabov *et al.*, 2009) used in the competitive methods. Figure 5 shows several examples of the estimated residual. Residual learning compensates for some of the high-frequency loss and improves the initial image reconstruction.

Figure 6 shows the architecture of the residual learning block used in our proposal. The residual network consists of two residual learning blocks and each residual learning

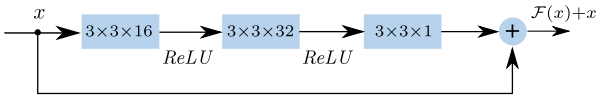

Fig. 6. Residual learning block. The residual learning block consists of 3 convolutional layers.

block has three layers. The first layer consists of 16 convolutional filters of size 3×3 with stride 1, followed by a ReLU non-linearity. The second layer has $3 \times 3 \times 32$ filters with stride 1, also followed by a ReLU non-linearity. The final layer consists of a single filter of size 3×3, which outputs the inferred residual image. Image dimensions are preserved in each layer by the appropriate zero-padding. Identity shortcuts are added to each residual block and are used to propagate the intermediate image reconstructions.

3.6. *Choice of the Loss Function*

Reconstructing the high-frequency content in the original image (i.e. edges, texture) is problematic for the linear autoencoder, and the residual network helps to alleviate this problem. Problems occur partly due to the fact that the lower frequency content is dominant in natural images and the learned measurement filters have a low-pass character, and partly due to the choice of the loss function used for training the network. It is known that the MSE loss function yields blurry images (Kristiadi, 2019). Thus, some papers suggest using a different loss function for the network training. As an example, Lohit *et al.* (2018) uses the adversarial loss function in addition to Euclidean loss to obtain better and sharper reconstructions. Furthermore, Du *et al.* (2018) uses perceptual loss in order to achieve better reconstruction results. The authors train their model using the Euclidean loss in the latent space of the VGG_{19} neural network (Simonyan and Zisserman, 2014).

In this paper, we fuse the per-pixel reconstruction loss in the autoencoder with the perceptual loss in latent space in the residual network. This is in contrast with Du *et al.* (2018), where the authors optimize the whole network using the Euclidean loss in the latent space. As a consequence, their method results in semantically informative reconstructions, but with low per-pixel accuracy. By using a combination of Euclidean and perceptual loss, we obtain semantically informative reconstructions that have high accuracy of per-pixel reconstruction resulting in higher PSNR compared to Du *et al.* (2018).

Pixel-wise Euclidean loss function for the autoencoder is defined as:

$$\mathcal{L}_1\left(\left\{\mathbf{\Phi}, \mathbf{\Phi}^+\right\}\right) = \left\|x - f\left\{x, \left\{\mathbf{\Phi}, \mathbf{\Phi}^+\right\}\right\}\right\|_2^2, \tag{14}$$

where $\mathbf{\Phi}$ denotes the weights of the measurement operator, $\mathbf{\Phi}^+$ are the weights of the reconstruction operator, x is the original image and $f\{x, \{\mathbf{\Phi}, \mathbf{\Phi}^+\}\}$ is the image reconstruction obtained by the autoencoder.

The residual part of the proposed network is trained separately from the autoencoder part using perceptual loss function $\mathcal{L}_2$ (Eq. (15)) in the latent space of the VGG_{19} network similarly to Du *et al.* (2018). In contrast with Du *et al.* (2018), we use a linear combination of Euclidean losses defined on the features of second and third max-pooling layer of the

VGG_{19} network instead of the Euclidean loss on individual feature map. The motivation for this is to simultaneously reconstruct both the low-level information contained in the bottom layers, as well as the high-level semantic features contained in the top layers of the VGG_{19} network.

$$\mathcal{L}_2\big(\{\boldsymbol{W}\}\big) = \frac{1}{2}\sum_{j=2}^{3}\big\|\phi_j(x) - \phi_j\big(f\{x, \boldsymbol{W}\}\big)\big\|_2^2. \tag{15}$$

In Eq. (15), ϕ_j denotes the feature map of the j-th layer of the VGG_{19} with input x. Furthermore, $\boldsymbol{W}$ denotes filter weights in the residual network and $f\{x, \boldsymbol{W}\}$ is the final image reconstruction.

4. Experiments

4.1. *Network Training*

In this section, we discuss the details of our network training procedure. We use *tensorflow* (Abadi *et al.*, 2015) deep learning framework for training and testing purposes. The training dataset is formed using uncalibrated JPEG images from the publicly available *Barcelona Calibrated Images Database* (Párraga *et al.*, 2010). Our training dataset is created by extracting 1676 image patches of size 256×256, taken from different parts of the original high-resolution (2268×1512) images. This corresponds to 107264 unique image blocks for training.

Adam optimizer (Kingma and Ba, 2015) ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1\text{e}^{-8}$) is used for the network training. The learning rate for the loss function $\mathcal{L}_1$ is set to 0.001 and the learning rate for the loss function $\mathcal{L}_2$ is set to 0.0001. The number of epochs in the training stage is set to 256. The training was performed on an Intel i7-4770K@3.50 GHz computer with NVIDIA GeForce GTX780 (GK110) graphic card.

We perform series of experiments to corroborate previous discussions and observations. In order to achieve a fair comparison framework, a set of 11 images (*Monarch, Fingerprint, Flintstones, House, Parrot, Barbara, Boats, Cameraman, Foreman, Lena, Peppers* – see TestDataset), which were used in the evaluation of the competitive methods are used for testing purposes with four different measurement sub-rates $r = \frac{M}{N}$, where $r \in \{0.25, 0.1, 0.04, 0.01\}$. In our experiments, block size of 32×32 is used.

4.2. *Measurement Matrix*

In Section 3, we have discussed the connection between the measurement matrix learned by the linear encoder and the one obtained by performing the PCA analysis. In addition, we proposed an efficient initialization method for the network weights. In this section, we perform an experiment to show that the performance of the trained linear autoencoder is limited by the performance of the PCA method network in terms of image reconstruction quality.

Table 1
Comparison of linear autoencoder and PCA in terms of reconstruction PSNR [dB].

PSNR [dB]	$r = 0.25$	$r = 0.10$	$r = 0.04$	$r = 0.01$
PCA	31.45	27.11	23.95	20.56
Linear autoencoder	31.39	27.06	23.92	20.55

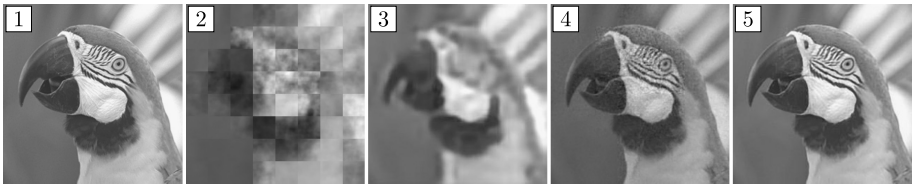

Fig. 7. Reconstruction results obtained using linear autoencoder for "*Parrot*" test image (1) and for two measurement ratios $r = 0.01$ (2, 3) and $r = 0.25$ (4, 5). Reconstructions labelled with (2) and (4) are obtained using the random Gaussian measurement matrix, while (3) and (5) are obtained using the adaptive measurement matrix.

Table 1 shows the mean reconstruction results in terms of PSNR for the standard test images. Notice that the reconstruction results are comparable. The slightly lower reconstruction performance of the linear encoder is due to the network not fully converging to the global minimum. Reconstruction results obtained by using the PCA method represent an upper boundary for the performance of the linear autoencoder network for CS image reconstruction.

In Fig. 7, reconstruction results obtained using random Gaussian and adaptive measurement matrix for *Parrot* test image are shown. The reconstructions are presented for measurement rates $r = \{0.01, 0.25\}$. Notice that the adaptive measurement matrix preserves more information compared to the random Gaussian matrix.

4.3. *Comparison to Other Methods*

In this section, we compare the proposed CS model to other state-of-the-art learning-based CS methods. To provide a fair comparison, we compare our method only to similar methods which use an adaptive linear encoding part.

We compare our method to the ImpReconNet (Lohit *et al.*, 2018), Adp-Rec (Xie *et al.*, 2017), FCMN (Du *et al.*, 2019) and two variants of PCS (Du *et al.*, 2018), namely PCS_{conv22} and PCS_{conv34}. In Table 2, mean PSNR reconstruction results (on the same test dataset) for the proposed method and for the competitive methods are shown. ImpReconNet (Euc) denotes a variant of a ReconNet model that uses Euclidean loss function for the network training, while the ImpReconNet (Euc+Adv) denotes a variant which uses a combination of Euclidean and adversarial loss. The competitive PNSR values are shown as reported in the original papers or reproduced using the available algorithms and models. In Fig. 8, "*Fingerprint*" test image reconstructions are shown compared to the ground-truth.

On one hand, FCMN and ImpReconNet yield similar results in terms of PSNR compared to our method (see Table 2), while on the other hand the aforementioned methods

Table 2
Reconstruction results obtained using the learned measurement matrix. Table contains mean PSNR reconstruction results for the standard test images at different measurement rates r. Although, FCMN achieves better results in terms of PSNR, it is clearly visible from Fig. 8 that it does not preserve structural information. This is due to the fact that PSNR measures image quality on per pixel basis, which is not a relevant measure for the preservation of high-level image features.

Mean PSNR [dB] for different methods	$r = 0.25$	$r = 0.10$	$r = 0.04$	$r = 0.01$
ImpReconNet (Euc) (Lohit *et al.*, 2018)	26.59	25.51	23.14	19.44
ImpReconNet (Euc + Adv) (Lohit *et al.*, 2018)	30.53	26.47	22.98	19.06
Adp-Rec (Xie *et al.*, 2017)	30.80	27.53	–	20.33
FCMN (Du *et al.*, 2019)	32.67	28.30	23.87	21.27
PCS_{conv22} (Du *et al.*, 2018)	–	–	19.38	18.30
PCS_{conv34} (Du *et al.*, 2018)	–	–	16.72	16.80
Proposed method	32.00	26.36	23.67	20.51

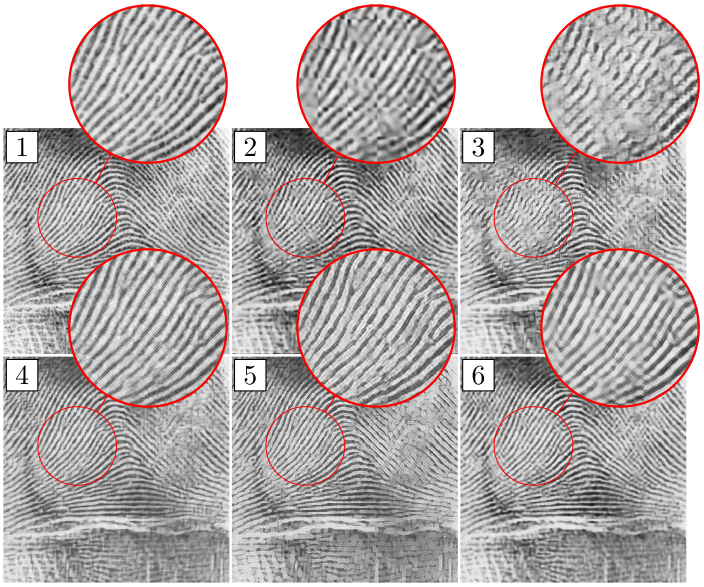

Fig. 8. Reconstruction results for "*Fingerprint*" test image and for measurement rate $r = 0.04$: (1) original, (2) ImpReconNet (Euc + Adv), $PSNR = 16.97$ dB, (3) FCMN, $PSNR = 19.05$ dB, (4) PCS_{conv22}, $PSNR = 14.83$ dB, (5) PCS_{conv34}, $PSNR = 14.35$ dB, (6) proposed method, $PSNR = 20.31$ dB. Our method results in better structure preservation compared to the ImpReconNet and FCMN methods, while we achieve significantly higher PSNR compared to the *PCS* methods by a margin of around 5 dB in PSNR.

do not preserve structural and high level semantic information. The two PCS methods preserve structural information, but yield images that contain significant amount of noise when observed on pixel-wise level. Our method benefits from the combination of pixel-wise Euclidean loss in image space and the Euclidean loss in the latent space of the VGG_{19} network resulting in high pixel-wise accuracy as well as good preservation of structural information. Similar observation holds for the "*Monarch*" reconstructions in Fig. 9 where a comparison between the competitive perceptual CS methods and the proposed method

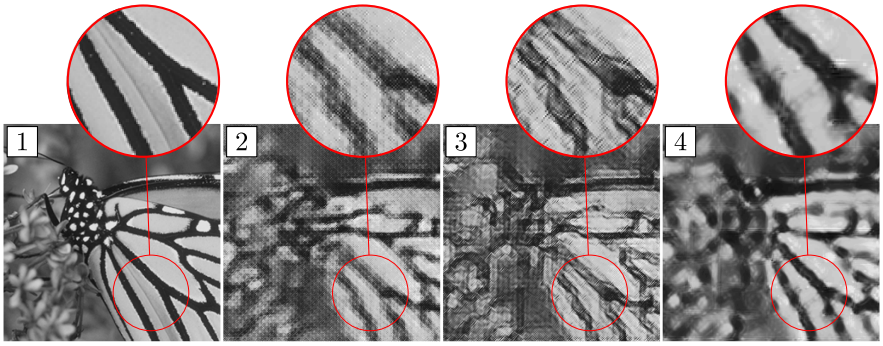

Fig. 9. Reconstruction results for "*Monarch*" test image and for measurement rate $r = 0.01$: (1) original, (2) PCS_{conv22}, $PSNR = 16.28$ dB, (3) PCS_{conv34}, $PSNR = 14.87$ dB, (4) proposed method, $PSNR = 18.04$ dB. Although PCS method successfully reconstructs higher semantic information, it suffers from significant amount of noise. Contrary, our method reconstructs the same amount of information with less noise and visual artifacts.

at extremely low measurement rate $r = 0.01$ is presented. Notice the high level of noise in the PCS reconstructions compared to the reconstruction obtained using the proposed method.

5. Discussion

Iterative nature and high computational complexity present the main drawbacks of the traditional CS reconstruction algorithms. Learning based methods for the CS image reconstruction present an efficient alternative to the traditional approach. Average per-image reconstruction time for a set of images with size 512×512 using traditional l_1 reconstruction method from the Sparse Modelling Software (SPAMS, 2010) optimization toolbox and a block-based approach with a subsampling rate of $r = 0.04$ is around 0.6 s, while the learning based method reduces the reconstruction time to around 0.025 s. An example of a real-world application of the learning-based approach is (Ralašić and Seršić, 2019), where the authors propose a real-time motion detection system in CS video which operates at extremely low measurement rates.

Better performance of the learning based methods in the reconstruction phase comes at an increased cost in the training phase. In order to learn the optimal measurement and reconstruction operators, learning based methods require an offline training procedure with a relatively large training dataset. Since learning based methods are data driven, they are also data dependent. Thus, if the statistical distribution of the training dataset significantly differs from the testing data, the performance of the learning based methods will be influenced. Finally, convolutional block image processing is not applicable in imaging modalities where the measurements correspond to the whole signal, and one cannot divide the signal into smaller blocks.

6. Conclusion

In this paper, we proposed a convolutional autoencoder architecture for the image compressive sensing reconstruction, which represents a non-iterative and extremely fast alternative to the traditional sparse optimization algorithms. In contrast with other learning based methods, we designed a measurement process which enables the model to be trained on normalized, mean-centred measurements which results in a significant speedup of the neural network convergence. Moreover, we proposed an efficient initialization method for the autoencoder network weights based on the connection between the learning-based CS approach and the principal component analysis. The residual learning network was used to further improve the initial reconstruction obtained by the autoencoder.

A combination of a pixel-wise Euclidean loss function for the autoencoder network training along with a Euclidean loss function in the latent space of the VGG_{19} network for the residual network training was proposed. It results in image reconstructions with higher pixel-wise reconstruction accuracy and more semantic information preserved at low measurement rates. In our future work, we will explore different loss functions that correspond to the notion of the perceptual loss.

Funding

This work was supported in part by the Croatian Science Foundation under Projects IP-2014-09-2625 and IP-2019-04-6703, and in part by the European Regional Development Fund under Grant KK.01.1.1.01.0009 (DATACROSS).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat I, S., Goodfellow, Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Online: https://www.tensorflow.org/.

Baldi, P., Hornik, K. (1989). Neural networks and principal component analysis: learning from examples without local minima. *Neural Networks*, 2, 53–58.

Baraniuk, R.G. (2007). Compressive sensing [lecture notes]. *IEEE Signal Processing Magazine*, 24, 118–121.

Beck, A., Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2, 183–202.

Becker, S., Bobin, J., Candès, E.J. (2011). NESTA: a fast and accurate first-order method for sparse recovery. *SIAM Journal on Imaging Sciences*, 4, 1–39.

Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C. (2000). Image inpainting. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniqus*. ACM Press/Addison-Wesley Publishing Co., pp. 417–424.

Bugeau, A., Bertalmio, M., Caselles, V., Sapiro, G. (2010). A comprehensive framework for image inpainting. *IEEE Transactions on Image Processing*, 19, 2634–2645.

Candes, E.J., Tao, T. (2006). Near-optimal signal recovery from random projections: universal encoding strategies? *IEEE Transactions on Information Theory*, 52, 5406–5425.

Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K. (2009). BM3D image denoising with shape-adaptive principal component analysis. In: *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*.

Dong, C., Loy, C.C., He, K., Tang, X. (2016). Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38, 295–307.

Du, J., Xie, X., Wang, C., Shi, G. (2012). Block-based compressed sensing of images and video. *Foundations and Trends in Signal Processing*, 4, 297–416.

Du, J., Xie, X., Wang, C., Shi, G. (2018). Perceptual compressive sensing. *Elsevier Neurocomputing*, 328, 105–112.

Du, J., Xie, X., Wang, C., Shi, G., Xu, X., Wang, Y. (2019). Fully convolutional measurement network for compressive sensing image reconstruction. *Elsevier Neurocomputing*, 328, 105–112.

Duarte, M.F., Eldar, Y.C. (2011). Structured compressed sensing: from theory to applications. *IEEE Transactions on Signal Processing*, 59, 4053–4085.

Duarte, M.F., Baraniuk, R.G. (2012). Kronecker compressive sensing. *IEEE Transactions on Image Processing*, 21, 494–504.

Dumoulin, V., Visin, F. (2016). *A guide to convolution arithmetic for deep learning*. ArXiv preprint arXiv:1603.07285, pp. 1–13.

Elad, M., Aharon, M. (2006). Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15, 3736–3745.

Hantao, Y., Feng, D., Shiliang, Z., Yongdong, Z., Tian, Q., Xu, C. (2019). DR2-Net: Deep Residual Reconstruction Network for Image Compressive Sensing. *Neurocomputing*. abs/1702.05743.

He, K., Zhang, X., Ren, S., Sun, J. (2015). Deep residual learning for image recognition. *Multimedia Tools and Applications*, 1–17. arXiv:1512.03385.

Ioffe, S., Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *International Conference on Machine Learning, 2015*, pp. 448–456.

Johnson, J., Alahi, A., Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In: *Springer Chinese Conference on Pattern Recognition and Computer Vision (PRCV), 2018*, pp. 268–279.

Karpathy, A. (2017). CS231n: Convolutional Neural Networks for Visual Recognition, Spring 2017. Online: http://cs231n.github.io/neural-networks-1/, accessed: June 2019.

Kingma, D.P., Ba, J. (2015). Adam: a method for stochastic optimization. In: *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*.

Kristiadi, A. (2019). Why does L2 reconstruction loss yield blurry images? Online: https://wiseodd.github.io/techblog/2017/02/09/why-l2-blurry/, accessed: June 2019.

Kulkarni, K., Lohit, S., Turaga, P., Kerviche, R., Ashok, A. (2016). ReconNet: non-iterative reconstruction of images from compressively sensed measurements. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lohit, S., Kulkarni, K., Kerviche, R., Turaga, P., Ashok, A. (2018). Convolutional neural networks for non-iterative reconstruction of compressively sensed images. *IEEE Transactions on Computational Imaging*, 4, 326–340.

Mallat, S.G., Zhifeng, Z. (2006). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41, 3397–3415.

Mousavi, A., Baraniuk, R.G. (2017). Learning to invert: signal recovery via deep convolutional networks. In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017*, pp. 2272–2276.

Mousavi, A., Patel, A.B., Baraniuk, R.G. (2015). A deep learning approach to structured signal recovery. In: *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1336–1343.

Mousavi, A., Dasarathy, G., Baraniuk, R.G. (2017). DeepCodec: adaptive sensing and recovery via deep convolutional neural networks. In: *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 744–744.

Needell, D., Tropp, J.A. (2009). CoSaMP: iterative signal recovery from incomplete and inaccurate samples. *Elsevier Applied and Computational Harmonic Analysis*, 26, 301–321.

Párraga, C.A., Baldrich, R., Vanrell, M. (2010). Accurate mapping of natural scenes radiance to cone activation space: a new image dataset. In: *Conference on Colour in Graphics, Imaging, and Vision, 2010*. Society for Imaging Science and Technology, pp. 50–57.

Pati, Y.C., Yagyensh, C., Rezaiifar, R., Krishnaprasad, P.S. (1993). Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In: *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, 1993*. IEEE, pp. 40–44.

Ralašić, I., Seršić, D. (2019). Real-time motion detection in extremely subsampled compressive sensing video. In: *2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 198–203.

Ralašić, I., Seršić, D., Petrinović, D. (2018). Off-the-shelf measurement setup for compressive imaging. *IEEE Transactions on Instrumentation and Measurement*, 68, 502–512.

Simonyan, K., Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556.

Sparse Modeling Software – optimization toolbox (2010). Online: http://spams-devel.gforge.inria.fr/index.html.

Takhar, D., Laska, J.N., Wakin, M.B., Duarte, M.F., Baron, D., Sarvotham, S., Kelly, K.F., Baraniuk, R.G. (2006). A new compressive imaging camera architecture using optical-domain compression. In: *Computational Imaging IV, 2006*. International Society for Optics and Photonics.

Testing dataset for learning-based compressive sensing reconstruction (2019). Available on: https://github.com/KuldeepKulkarni/ReconNet/tree/master/test/test_images.

Wright, S.J., Nowak, R.D., Figueiredo, M.A.T. (2009). Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57, 2479–2493.

Xie, X., Wang, Y., Shi, G., Wang, C., Du, J., Han, X. (2017). Adaptive measurement network for CS image reconstruction. In: *CCF Chinese Conference on Computer Vision 2017*. Springer.

Yang, J., Wright, J., Huang, T.S., Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19, 2861–2873.

I. Ralasic received his BSc degree in computing and MSc degree in information and communication technology from the University of Zagreb, in 2014 and 2016, respectively. He is currently pursuing the PhD degree at the University of Zagreb, Faculty of Electrical Engineering and Computing. His current research interests include signal processing, compressive sensing, sparse modelling and machine learning. He is a Student Member of IEEE.

D. Seršić received the diploma degree and the MS and PhD degrees in electrical engineering from the University of Zagreb, Zagreb, Croatia, in 1986, 1993, and 1999, respectively. Since 1987, he has been with the Faculty of Electrical Engineering and Computing, University of Zagreb, where he is currently a full professor. His current research interests include theory and applications of wavelets, advanced signal and image processing, adaptive systems, blind source separation, and compressive sensing. Dr. Seršić is a member of the European Association for Signal Processing. From 2006 to 2008, he served as the chair for the Croatian IEEE Signal Processing Chapter.

S. Šegvić received his PhD degree in computer science, in 2004. He spent one year as a post-doctoral researcher at IRISA/INRIA, Rennes, France, and also at TU Graz, Austria. He is currently a full professor at UniZg-FER. His research and professional interests focus on lightweight convolutional architectures for semantic segmentation, detection, re-identification, outlier detection, and semantic forecasting.