

Convolutional Sparse Support Estimator Network (CSEN): From Energy-Efficient Support Estimation to Learning-Aided Compressive Sensing

Mehmet Yamaç¹, Mete Ahishali¹, Serkan Kiranyaz¹, *Senior Member, IEEE*,
and Moncef Gabbouj², *Fellow, IEEE*

Abstract—Support estimation (SE) of a sparse signal refers to finding the location indices of the nonzero elements in a sparse representation. Most of the traditional approaches dealing with SE problems are iterative algorithms based on greedy methods or optimization techniques. Indeed, a vast majority of them use sparse signal recovery (SR) techniques to obtain support sets instead of directly mapping the nonzero locations from denser measurements (e.g., compressively sensed measurements). This study proposes a novel approach for learning such a mapping from a training set. To accomplish this objective, the convolutional sparse support estimator networks (CSEs), each with a compact configuration, are designed. The proposed CSEN can be a crucial tool for the following scenarios: 1) real-time and low-cost SE can be applied in any mobile and low-power edge device for anomaly localization, simultaneous face recognition, and so on and 2) CSEN’s output can directly be used as “prior information,” which improves the performance of sparse SR algorithms. The results over the benchmark datasets show that state-of-the-art performance levels can be achieved by the proposed approach with a significantly reduced computational complexity.

Index Terms—Learned compressive sensing (CS), sparse signal representation, support recovery.

I. INTRODUCTION

SPARSE representation or sparse coding (SC) denotes representing a signal as a linear combination of only a small subset of a predefined set of waveforms. Compressive sensing (CS) [1], [2] can be seen as a special form of SC, while a signal, $\mathbf{s} \in \mathbb{R}^d$ that has a sparse representation, $\mathbf{x} \in \mathbb{R}^n$ in a dictionary or basis $\Phi \in \mathbb{R}^{d \times n}$, can be acquired in a compressed manner using a linear dimensional reduction matrix, $\mathbf{A} \in \mathbb{R}^{m \times d}$. Therefore, this signal can also be

represented in a sparse manner in the dictionary, $\mathbf{D} \in \mathbb{R}^{m \times n}$ (which can be called equivalent dictionary [3], where $m \ll n$, and typically assumed to be full-row rank), which is the matrix multiplication of the measurement matrix, \mathbf{A} , and predefined dictionary, Φ , i.e., $\mathbf{D} = \mathbf{A}\Phi$. In the SC literature, *signal synthesis* refers to producing a signal, $\mathbf{y} = \mathbf{D}\mathbf{x} \in \mathbb{R}^m$, using a sparse code, $\mathbf{x} \in \mathbb{R}^n$ and a prespecified dictionary, \mathbf{D} . On the other hand, the *signal analysis* deals with finding the sparse codes, \mathbf{x} from the given measurements, \mathbf{y} , with respect to the dictionary \mathbf{D} [4]. Sparse support estimation (SE) [5]–[7] refers to finding the location indices of nonzero elements in SCs. In other words, it is the localization of the smallest subset of the atoms, which are the basis waveforms in the dictionary, whose linear combination represents the given signal well enough. On the other hand, *sparse signal recovery* (SR) refers to finding the values of these nonzero elements of SCs. SE and SR are intimately linked in such a way that the SE of a sparse signal is first performed; then, an SR will be trivial using the ordinary least-squares optimization. In fact, this is the main principle of most greedy algorithms [8], [9]

The literature that purely targets SE is relatively short compared to extensive studies on sparse SR [10]. Many existing works, first, apply a coarse SR using existing SR methods, and then, SE can be easily performed if SE is the main objective. Indeed, there are many applications where computing the support set is more important than computing the magnitudes of SCs. For instance, in an SR-based classification (SRC) [11], such as face recognition [12], the training samples are stacked in the dictionary in such a way that a subset of the columns consists of the samples of a specific class. As another example, in cognitive radio systems, only a small ratio of all spectrum is occupied for a given time interval. Therefore, finding the occupied spectrum (i.e., the support set) is the primary concern [13], [14]. Similarly, in a ground-penetrating radar imaging system, finding the location of the target is more important than predicting the actual signal magnitudes [15].

In this study, a novel convolutional sparse support estimator networks (CSEs) is proposed with two primary objectives, as shown in Fig. 1. First, this approach enables learning-based noniterative SE with minimal computational complexity. To accomplish this, we use two compact convolutional neural

Manuscript received April 2, 2020; revised October 23, 2020 and April 16, 2021; accepted June 25, 2021. This work was supported in part by the NSF CVDI Program under Project AMALIA funded by the Business Finland and Mad@Work and Stroke-Data projects funded by Haltian and Business Finland. (*Corresponding author: Mehmet Yamaç.*)

Mehmet Yamaç, Mete Ahishali, and Moncef Gabbouj are with the Faculty of Information Technology and Communication Sciences, Tampere University, 33720 Tampere, Finland (e-mail: mehmet.yamac@tuni.fi).

Serkan Kiranyaz is with the Department of Electrical Engineering, Qatar University, Doha, Qatar.

This article has supplementary material provided by the authors and color versions of one or more figures available at <https://doi.org/10.1109/TNNLS.2021.3093818>.

Digital Object Identifier 10.1109/TNNLS.2021.3093818

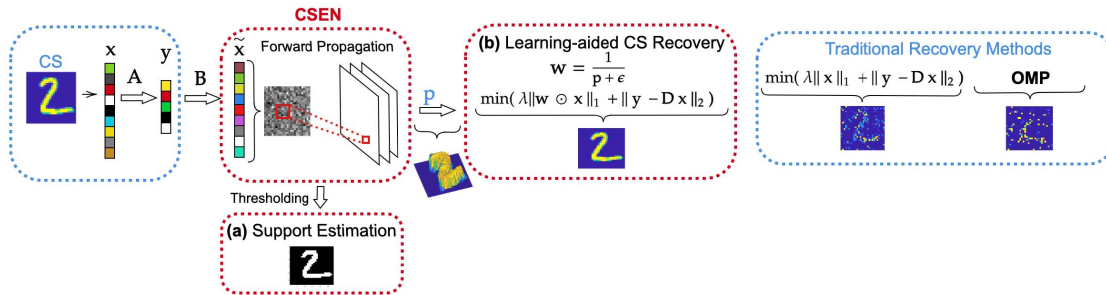


Fig. 1. Proposed CSEN with two potential applications. (a) (bottom-left) sparse SE. (b) (top-middle) Learned-aided CS-sparse signal reconstruction with CSEN versus (top-right) traditional recovery methods. 1) OMP [8] and 2) ℓ_1 -minimization.

network (CNN) configurations, both of which are designed without the dense layers [16]. The proposed CSENs are trained to optimize the SEs. To the best of our knowledge, this is the first study that proposes a learning-based approach for noniterative SE. Hence, in order to perform comparative evaluations, we train the following state-of-the-art CS signal reconstruction deep neural networks as the support estimators: 1) ReconNet [17] that originally works on the spatial domain; 2) the learned AMP (LAMP) [18] that is the deep version of AMP [19], which is the state-of-the-art optimization scheme working on the sparse domain; and 3) learned ISTA (LISTA) [20] is the deep learning version of well-known SR algorithm, iterative soft-thresholding algorithm (ISTA) [21], which is the first attempt to unfold an optimization-based SR algorithm in a neural network manner. An extensive set of experiments over four benchmark datasets has demonstrated that the proposed CSEN approach outperforms deep counterparts, especially dealing with a structural sparse signal. In the first experimental setup, we simulate a CS system making data acquisition from the MNIST dataset in different measurement rates (MRs). Moreover, the proposed SE system is shown to improve the SE performance compared to its deep counterparts, especially in low MRs and imperfect sparsity (in the case of CS of approximate sparse signal or noisy environment). Furthermore, CSEN is tested on a well-known support recovery problem, where face recognition is performed based on sparse codes [11]. We use two benchmark datasets, Yale-B [22] and CelebA [23], in our experiments. Comparative evaluations performed against the two state-of-the-art dictionary-based (representation-based) face recognition methods in the literature, SR-based face recognition [11], and collaborative learning [24] have demonstrated that the proposed CSEN approach outperformed both methods. Furthermore, we develop a CSEN-based Coronavirus disease (COVID-19) recognition system from X-Ray images [25]. In this problem, CSEN shows its superiority over other representation-based classifiers and traditional approaches on classification tasks when the training size is small/moderate.

As for the second objective, we focus on an alternative usage of CSENs. Instead of using them as support estimators, which naturally requires the hard-thresholding of the network outputs, these outputs can be directly used as prior information about the sparse signals. It is a well-known fact that having prior information about the nonzero locations, such as the

probability map, $p(x)$ (or simply \mathbf{p}), on the support set, could improve the conventional SR algorithms [26]. However, in many cases, it is not clear how to obtain such prior information in advance. The most common usage of such a system appears in dynamical sparse recovery [27], where previous SEs can be used as priors for the next estimation. In this study, we have demonstrated that CSEN outputs can be a better alternative for the prior information of the nonzero locations. Therefore, CSEN is now used as a *learning-aided CS reconstruction* scheme, where the prior information comes directly from the CSEN outputs. A wide range of experiments shows that this approach has great potential to improve the SR performance of traditional approaches for sparse SR problems. As mentioned above, we used CS imaging simulation, but this time signal reconstruction error is compared with state-of-the-art conventional SR approaches. Fig. 1 illustrates a representative graph of two different applications of CSENs: 1) performing SE from CS measurement vector, \mathbf{y} and 2) the output of CSEN is used as the side information, \mathbf{p} , which gives the estimated probability of being nonzero for each index. In this simple illustration, we assume that the hand-writing signal “2” is sparse in the spatial domain such that $\Phi = \mathbf{I}$; therefore, $\mathbf{D} = \mathbf{A}\mathbf{I} = \mathbf{A}$, and \mathbf{B} is a denoiser matrix such as \mathbf{D}^T , or $(\mathbf{D}^T\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{D}^T$, where λ is the regularization parameter. Moreover, we also show the possibility of using the learning-aided CS reconstruction scheme when the signal is not sparse in the spatial domain but in a proper domain. In this respect, the sparsity of natural images in the gradient domain is used to have a CSEN-aided total variation minimization system.

The rest of this article is organized as follows. In Section II, we start by giving mathematical notation that is used in this article. A brief overview of sparse representation and CS theory, with an emphasis on state-of-the-art sparse SR and SE techniques, will be given in Section III. In the same section, we also introduce case studies of SE that are chosen for this work. Then, we discuss the limitations of existing support estimator techniques. In Section IV, we will present the proposed learned-based SE scheme and the two compact CSEN models. Experimental evaluations of the study will also be given at the end of this section, which we can divide into five main categories according to the case studies: 1) basic SE performance evaluation on the MNIST dataset that is performed to compare CSENs with the aforementioned state-of-the-art deep networks; 2) SE-based face recognition performance evolution

of proposed SE with an emphasis on how CSEN-based SE has the ability to improve the classical representation-based approaches; 3) a CSEN-based COVID-19 recognition system; 4) performance comparison of classical compressing sensing reconstruction techniques and proposed learned-aided SR in terms of both speed and reconstruction accuracy in the MNIST dataset; and 5) CSEN-aided total variation system for recovery of compressively sensed natural images. Having theoretical and experimental analysis, in Section VI, we will present a more detailed discussion on how the proposed scheme differs from the state-of-the-art SR and SE techniques, pros and cons, and possible usage scenarios with an emphasis on the flexibility of proposed CSEN in different scenarios. Finally, the conclusions are drawn in Section VII.

II. NOTATIONS

In this work, we define the ℓ_p -norm of any vector $\mathbf{x} \in \mathbb{R}^n$ as $\|\mathbf{x}\|_{\ell_p} = (\sum_{i=1}^n |x_i|^p)^{1/p}$ for $p \geq 1$. The ℓ_0 -norm of the vector $\mathbf{x} \in \mathbb{R}^n$ is given as $\|\mathbf{x}\|_{\ell_0} = \lim_{p \rightarrow 0} \sum_{i=1}^n |x_i|^p = \#\{j : x_j \neq 0\}$, and the ℓ_∞ is defined as $\|\mathbf{x}\|_{\ell_\infty} = \max_{i=1, \dots, n} (|x_i|)$. A signal \mathbf{s} can be defined as a strictly k -sparse signal if it can be represented with less than $k + 1$ nonzero coefficients in a proper basis Φ , i.e., $\|\mathbf{x}\|_0 \leq k$, where $\mathbf{s} = \Phi \mathbf{x}$. We also define a sparse support set or simply support set, $\Lambda \subset \{1, 2, 3, \dots, n\}$, as the set of indices that represent the nonzero coefficients, i.e., $\Lambda := \{i : x_i \neq 0\}$. The complement of support set, Λ , with respect to $\{1, 2, 3, \dots, n\}$ is given as $\Lambda^c = \{1, 2, 3, \dots, n\} \setminus \Lambda$. In this manner, $\mathbf{x}_\Lambda \in \mathbb{R}^{|\Lambda|}$ is a vector consisting of nonzero elements of $\mathbf{x} \in \mathbb{R}^n$, where $|\Lambda|$ refers to the number of the nonzero coefficients. Similarly, $\mathbf{M}_\Lambda \in \mathbb{R}^{m \times |\Lambda|}$ denotes a matrix that consists of the columns of a matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ indexed by support Λ .

III. RELATED WORK

The CS theory claims that a signal \mathbf{s} can be sensed using far fewer linear measurements m than Nyquist-/Shannon-based traditional methods' use, d , i.e.,

$$\mathbf{y} = \mathbf{A}\mathbf{s} = \mathbf{A}\Phi\mathbf{x} = \mathbf{D}\mathbf{x} \quad (1)$$

where $\mathbf{A} \in \mathbb{R}^{m \times d}$ is the measurement matrix and $\mathbf{D} \in \mathbb{R}^{m \times n}$ is called the equivalent dictionary. It can be demonstrated that sparse representation

$$\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t. } \mathbf{D}\mathbf{x} = \mathbf{y} \quad (2)$$

is unique if $m \geq 2k$ [28] and $\|\mathbf{x}\|_0 \leq k$. In brief, the uniqueness of the sparse representation in (2) shows that any k -sparse signal pair can still be distinguished in the equivalent dictionary, \mathbf{D} . However, the problem in (2) is that this is a nonconvex problem and known to be NP-hard. The most common approach is the relaxation of the ℓ_0 -norm to the closest convex norm, which is ℓ_1 -norm

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t. } \mathbf{x} \in \mathcal{U}(\mathbf{y}) \quad (3)$$

where $\mathcal{U}(\mathbf{y}) = \{\mathbf{x} : \mathbf{D}\mathbf{x} = \mathbf{y}\}$, which is known as basis pursuit [29]. The surprising result of the CS theory is that, even if the exact recovery of the signal, \mathbf{s} , was not possible

by using the minimum norm solution, a tractable solution is possible using (3), when \mathbf{D} satisfies some properties, such as restricted isometry property [30] and $m > k(\log(n/k))$.

However, the signal of interest, \mathbf{x} , is not perfectly k -sparse but approximately sparse in most of the cases. In addition, CS measurements, most probably, are corrupted by an additive noise during data acquisition, quantization, and so on. As a result, we handle $\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{z}$, where \mathbf{z} is the additive noise. In this case, the constraint can be relaxed by setting $\mathcal{U}(\mathbf{y}) = \{\mathbf{x} : \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2 \leq \epsilon\}$, which is known as basis pursuit denoising (BPDN) [29] or the Dantzig selector [31], if we set $\mathcal{U}(\mathbf{y}) = \{\mathbf{x} : \|\mathbf{D}^T(\mathbf{y} - \mathbf{D}\mathbf{x})\|_\infty \leq \lambda\}$. In the noisy case, even exact recovery of sparse signal is not possible, stable recovery is well studied in the literature for BPDN [32] and the Dantzig selector [33], [34]. We mean by stable recovery is that a stable solution $\hat{\mathbf{x}}$ obeys $\|\mathbf{x} - \hat{\mathbf{x}}\| \leq \kappa\|\mathbf{z}\|$, where the κ is small constant. Another related formulation is

$$\min_{\mathbf{x}} \{\|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{x}\|_1\} \quad (4)$$

which is known as Lasso [35] formulation, which is also known to produce stable solution in noisy case and exact solution in noise free case [36].

A. Generic Sparse Support Estimation

In many application scenarios, detecting the indices of the nonzero coefficients' location, Λ , is more important than computing these coefficients. To list a few, in a sparse anomaly (either from CS [37] or uniform sampled measurements) detection problem [38], where a group of users initiates a flooding attack to a communication network (specifically for a VoIP network), detecting the malicious user group (a subset of all users) is more critical. Among others, CS-based active user detection in the downlink of a CDMA system [39] and for the uplink of an NOMA [40], [41] system can be counted. Such systems are believed to play an important role in 5G communication technology. As discussed in Section I, other examples may be listed as sparse representation-based classifications [11], [12] and radar imaging [15], [42].

Mathematically speaking, for the linear measurement model given in (1) and with additive noise, $\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{z}$, we define the following support estimator $\mathcal{E}(\cdot, \cdot)$:

$$\hat{\Lambda} = \mathcal{E}(\mathbf{y}, \mathbf{D}) \quad (5)$$

where $\hat{\Lambda}$ is the estimated support. For the noise-free case, \mathbf{x} is exactly k -sparse, and the exact Λ recovery performance of an algorithm coincides with the sparse SR performance. This an expected outcome since the unique representation is satisfied when $m > 2k$. In the noisy case, even if the exact SR is not possible, it is still possible to recover the support set exactly. In the literature, several studies have proposed to provide information-theoretical (i.e., the optimal decoder, \mathcal{E} 's performance) guarantee conditions for exact [5], [10], [43], [44] and partial SE [7], [10], [45]. However, in most of the practical applications, a tractable SR method is applied first to find an estimation $\hat{\mathbf{x}}$ of the sparse signal \mathbf{x} ; then, a componentwise thresholding is applied to $\hat{\mathbf{x}}$ to compute the estimated support, as illustrated in Fig. 2.

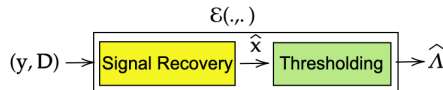


Fig. 2. Most common model for a practical support estimator.

A common approach is to follow an iterative sparse SR method from the CS literature. For instance, it is proven in [46] that, if $\min_{i \in \Lambda} |x_i| > 8\sigma(2 * \log(n))^{1/2}$, then one can recover the support set exactly using Lasso with $\lambda = 2(2 * \log(n))^{1/2}$, where σ^2 is variance of the measurement noise. This theorem is valid in the case that the equivalent dictionary satisfies the mutual coherence property defined in [46]. One may clearly deduce from their results that accurate SE is possible via Lasso if the nonzero coefficients' magnitudes are above a certain level determined by the noise. Similarly, the conditions of exact support recovery under noise using OMP are given in [47], and partial support recovery performance bounds of AMP are in [48]. Along with these SR algorithms in the CS literature, which are iterative methods, traditional linear decoders, such as maximum correlation (MC) [49], $\hat{\mathbf{x}}^{MC} = \mathbf{D}^T \mathbf{y}$, and LMSEE [48], $\hat{\mathbf{x}}^{LMSEE} = (\mathbf{D}^T \mathbf{D} + \sigma_z^2 \mathbf{I}_{n \times n})^{-1} \mathbf{D}^T \mathbf{y}$, are also used in many applications. The theoretical performance bounds of these methods are also given in [48].

B. Case Study of SE: Representation-Based Classification

Consider an image from a particular class is queried. It can be expected from the estimated SCs, $\hat{\mathbf{x}}$, to have significant (nonzero) entries that are located in a specific location so that the corresponding columns in the dictionary matrix, \mathbf{D} , are the samples from the actual class of the image. This problem is also known as the representation-based classification, which is a typical example where the support set location is the main information that we are seeking.

In [11], ℓ_1 -minimization is used to obtain such a sparse code to determine the identity of face images. However, in reality, such an ideal decomposition is not accomplished in general because face images show a high correlation among different classes. This is why, instead of using the estimated sparse codes, $\hat{\mathbf{x}}$ obtained by an SR technique, such as (4), the authors propose a four steps solution.

- 1) *Normalization*: Normalize all the atoms in \mathbf{D} and \mathbf{y} to have unit ℓ_2 -norm.
- 2) *SR*: $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1$ s.t. $\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2$.
- 3) *Residual Finding*: $\mathbf{e}_i = \|\mathbf{y} - \mathbf{D}_i \hat{\mathbf{x}}_i\|_2$, where $\hat{\mathbf{x}}_i$ is the estimated coefficients corresponding the class i .
- 4) *Class Determination*: $\text{Class}(\mathbf{y}) = \arg \min(\mathbf{e}_i)$.

This technique and its similar variants have been reported to perform well not only in face recognition but many other classification problems [50], [51]. Later, Zhang *et al.* [24] propose to change the second step, from ℓ_1 -minimization to the classical ℓ_2 -minimization; $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \{\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_2^2\}$, which has a closed-form solution, $\hat{\mathbf{x}} = (\mathbf{D}^T \mathbf{D} + \lambda \mathbf{I}_{n \times n})^{-1} \mathbf{D}^T \mathbf{y}$. This collaborative representation-based classification (CRC) was reported to achieve a comparable classification performance for different classification problems. For face recognition problems, in particular, the authors reported that high classification accuracies were obtained especially for high MRs.

C. Sparse Signal Reconstruction With Side Information of Support Set

Consider the case where SE is not the main concern but SR is. In case side information is available about the support set, an improvement to ℓ_1 -minimization can be achieved in sparse SR as follows:

$$\min_{\mathbf{x}} \{\|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{w} \odot \mathbf{x}\|_1\} \quad (6)$$

where \odot is elementwise multiplication operator and \mathbf{w} is the predefined cost that imposes the prior information about each element's values. In the concept of modified CS [52] and CS with prior information literature, the cost function, \mathbf{w} , generally appears in the form of $w_i = (1/(p_i + \epsilon))$, where $\epsilon > 0$ is a predefined constant and p_i is the i th element of the vector \mathbf{p} , which is a type of a measure, such as prior likelihood [26] of the support set, which could represent the probability of the (i)th element being nonzero.

D. Limitations of Existing Support Estimators

Both SE and SR algorithms guarantee to perform well if the equivalent dictionary \mathbf{D} satisfies certain properties, such as mutual incoherence [53]. However, in many practical scenarios, \mathbf{D} fails to satisfy these properties, e.g., in the face recognition problem, the atoms of \mathbf{D} , vectorized faces, are highly correlated. The second limitation of traditional sparse recovery algorithms is that they are iterative methods and computationally costly. Therefore, the support estimators relying on these sparse recovery algorithms may not be feasible, especially in real-time applications. The third limitation of state-of-the-art SR techniques, such as ℓ_1 -minimization, is that there is a lower limit for MR (see phase transition [54]); below this limit, the SR algorithms start to fail completely. This limit generally depends on the wellness of \mathbf{D} (defined by properties such as mutual incoherence [53]). Therefore, SE techniques that build upon an SR algorithm tend to fail if \mathbf{D} does not satisfy the required properties, e.g., if the atoms of \mathbf{D} are highly correlated.

On the other hand, when it comes to SR techniques leveraging SE as prior information, despite the fact that a good improvement can be achieved using such prior information, most of the works assume that the information is available in advance; however, they do not mention how to obtain such a \mathbf{p} .

IV. CONVOLUTIONAL SPARSE SUPPORT ESTIMATOR NETWORK

Recent advance in deep neural networks [18], [20] enables a noniterative solution for the sparse SR. It is often reported that they produce a solution $\hat{\mathbf{x}}$, which is closer to \mathbf{x} than the ones obtained by an iterative approach. They can still work under those MRs where classical CS recovery algorithms fail. Nevertheless, their complex configuration with millions of parameters causes certain computational complexity issues, such as speed and memory problems, especially when they are used in edge devices with limited power, speed, and memory.

If one may wish to find only support Λ instead of the sign and amplitude of \mathbf{x} , a traditional machine learning approach would be sufficient. In this study, we propose a support estimator, $\mathcal{E}(\cdot)$, which can be performed by a compact

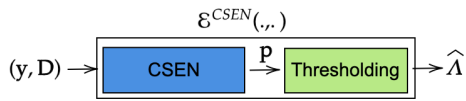


Fig. 3. Proposed model for an efficient support estimator.

CSEN network. Another crucial objective is to have the ability to learn from a minimal training set with a limited number of labeled data. A typical application where this approach can benefit from is face recognition via sparse representations, where only a few samples of each identity are available.

Let us define a binary mask $\alpha \in \{0, 1\}^n$, as follows:

$$\alpha_i = \begin{cases} 1 & \text{if } i \in \Lambda \\ 0 & \text{else.} \end{cases} \quad (7a)$$

$$(7b)$$

Consequently, the problem of finding an estimation $\hat{\alpha}$ of this binary mask will be equivalent to producing an SE $\hat{\Lambda}$, i.e., $\hat{\Lambda} = \{i \in \{1, 2, \dots, n\} : \hat{\alpha}_i = 1\}$.

To accomplish this objective, first, the CSEN network with input and output, $\mathcal{P}(\mathbf{y}, \mathbf{D}) : \mathbb{R}^n \mapsto [0, 1]^n$, produces a vector \mathbf{p} that gives the information about the probability of each index to be in support set such that $p_i \in [0, 1]$. Then, the final support estimator, $\mathcal{E}(\mathbf{y}, \mathbf{D})$, will produce an SE such that $\hat{\Lambda} = \{i \in \{1, 2, \dots, n\} : p_i > \tau\}$, by thresholding \mathbf{p} with τ , where τ is a fixed threshold.

As shown in Fig. 3, the proposed SE approach is different from the conventional SR-based methods, which directly thresholds $\hat{\mathbf{x}}$ for SE. Moreover, the input–output pair is different. The proposed CSEN learns over $(y^{\text{train}}, v^{\text{train}})$ to compute \mathbf{p} , while the conventional SR methods work with $(y^{\text{train}}, x^{\text{train}})$ to first make the sparse signal estimation and then compute SE by thresholding it. As evident in Fig. 1, the application of direct SR may cause noisy estimation of the support codes, while the proposed CSEN has the advantage of learning the pattern of the support codes and, therefore, can predict their most-likely location with proper training.

In this study, the proposed CSEN models consist of only convolutional layers in the type of fully convolutional networks [16] that are trained by optimizing the SEs. Since the SE problem involves one-to-one mapping, other network types, such as multilayer perceptrons (MLPs), can also be used as in [18]. However, this brings two limitations compared to CSENs: high computational complexity and overfitting due to the limited training data and number of parameters in the network. In Section V, it will be shown that such an approach yields a poor generalization and is not robust to noise.

When a CSEN is trained, it learns the following transformation: $\hat{\alpha} \leftarrow \mathcal{P}(\tilde{\mathbf{x}})$, where $\hat{\alpha}$ is the estimation of binary mask representing the estimated support for the signal \mathbf{x} , and the proxy $\tilde{\mathbf{x}} = \mathbf{B}\mathbf{y}$ with $\mathbf{B} = \mathbf{D}^T$, or $(\mathbf{D}^T\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{D}^T$, i.e., the MC and LMMSE formula in [48]; hence, $\mathbf{x}, \tilde{\mathbf{x}} \in \mathbb{R}^N$. First, the proxy $\tilde{\mathbf{x}}$ is reshaped to 2-D plane (e.g., the original size of the image or predefined search grid). Correspondingly, the proxy $\tilde{\mathbf{X}}$ (the matrix version of $\tilde{\mathbf{x}}$) is convolved with \mathbf{w}_1 , the set of weight kernels connecting the input layer to the next layer with N_1 filters to form the input of the next layer with the summation of weight biases \mathbf{b}_1 as follows:

$$\mathbf{f}_1 = \left\{ S_1(\text{ReLu}(b_1^i + \mathbf{w}_1^i * \tilde{\mathbf{X}})) \right\}_{i=1}^{N_1} \quad (8)$$

where $S_1(\cdot)$ is the down-sampling or identity operator, \mathbf{w}_1^i is the i th kernel weight, and b_1^i is its corresponding bias term and $\text{ReLu}(x) = \max(0, x)$. In more general form, the k th feature map of layer l can be expressed as

$$\mathbf{f}_l^k = S_l \left(\text{ReLu} \left(b_l^k + \sum_i^{N_{l-1}} \mathbf{w}_l^{ik} * \mathbf{f}_{l-1}^i \right) \right) \quad (9)$$

where \mathbf{w}_l^i is the i th kernel weight of the l th layer, b_l^i is its corresponding bias term, N_l is the number of filter in this layer, and $S_l(\cdot)$ is either the down- or up-sampling or identity operator depending on the CSEN structure. The trainable parameters of the network would be

$$\Theta_{\text{CSEN}} = \left\{ \{\mathbf{w}_1^i, b_1^i\}_{i=1}^{N_1}, \{\mathbf{w}_2^i, b_2^i\}_{i=1}^{N_2}, \dots, \{\mathbf{w}_L^i, b_L^i\}_{i=1}^{N_L} \right\}$$

for a L layer CSEN.

In the proposed approach, the mean square error (mse) is computed between its binary mask, α , and CSEN's actual output, $\mathcal{P}_\Theta(\mathbf{x})_p$, as follows:

$$E(\mathbf{x}) = \sum_p (\mathcal{P}_\Theta(\mathbf{x})_p - \alpha_p)^2 \quad (10)$$

where v_p is the p th pixel of α . The CSEN network is trained using samples in the train data, $D_{\text{train}} = \{(\tilde{\mathbf{x}}^{(1)}, \alpha^{(1)}), (\tilde{\mathbf{x}}^{(2)}, \alpha^{(2)}), \dots, (\tilde{\mathbf{x}}^{(s)}, \alpha^{(s)})\}$. Please note that, even if we use mse as the loss function in the original CSEN design, depending on the application, any other regularization function (e.g., ℓ_1 -norm and mixed norm) can be added to this cost function. As an example, we present a strategy to approximate the loss function, which is group ℓ_1 -norm in addition to mse.

V. RESULTS

In order to evaluate the effect of different network configurations, in this study, we use two different CSEN configurations and perform a comprehensive analysis of each of them. Generally, each convolutional layer has a dimension reduction capability with pooling functions. However, the first proposed network architecture consists of only convolutional layers with ReLu activation functions to preserve the sparse signal (e.g., image) dimensions at the output layer. In this configuration (CSEN1), we propose to use three convolutional layers with 48 and 24 hidden neurons and 3×3 filter size, as given in Fig. 4. CSEN2 is a slight modification of CSEN1 configuration, as shown in Fig. 5, by using up- and down-sampling layers. Although this modification increases the number of parameters, in return, it yields substantial performance improvement over MNIST. While the SE performance analysis over MNIST has done using CSEN1 and CSEN2, only CSEN1 results are reported since CSEN2 produces similar recognition rates (~ 0.001 difference) for face recognition. In any case, both network configurations are compact compared to the deep CNNs that have been proposed recently. For example, the study in [17] proposes ReconNet for SR, which consists of six convolutional layers with 32 neurons or more in each layer.

Since there is no competing method for SE that is similar to the proposed method, we use the ReconNet [17] in this study on the SE problem by directly giving $\tilde{\mathbf{x}}$ as the input

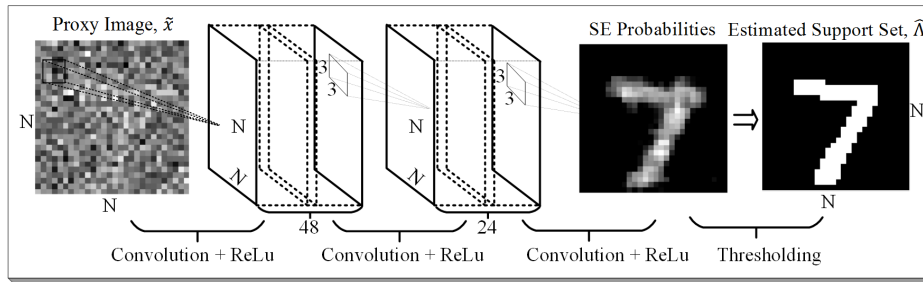


Fig. 4. Type-I CSEN (CSEN1).

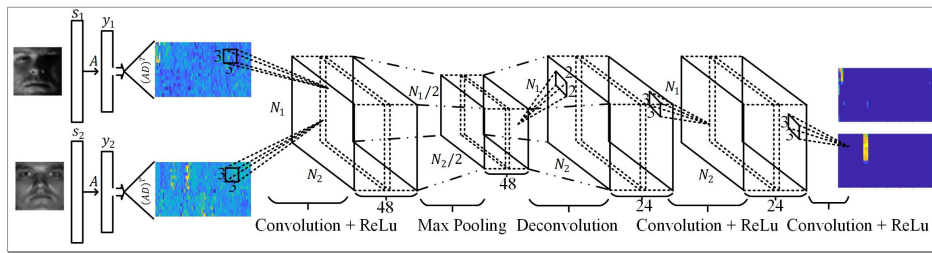


Fig. 5. Type-II CSEN (CSEN2).

and removing the denoiser block at the end for comparative evaluations. Finally, we apply thresholding over the output of ReconNet to generate SE i.e., $\hat{\Lambda}_R = \{i \in \{1, 2, \dots, n\} : \mathcal{P}_R(\tilde{\mathbf{x}}) > \tau\}$, where $\mathcal{P}_R(\cdot)$ is ReconNet with fully convolutional layers. ReconNet is originally a CS recovery algorithm working directly on spatial domain, i.e., $\hat{\mathbf{s}} \leftarrow \mathcal{P}(\mathbf{y})$ instead of solving them in the sparsifying dictionary, i.e., $\hat{\mathbf{s}} = \Phi \hat{\mathbf{x}}$ where $\hat{\mathbf{x}} \leftarrow \mathcal{P}(\mathbf{y})$. Therefore, ReconNet serves as a deep CSEN approach against which the performance of the two compact CSENs will be compared. Moreover, we also train the state-of-the-art deep SR solution, LAMP, and, first of its kind, LISTA networks, in order to use them over the SE problem. For the LAMP method, it is possible to predefine the number of layers in advance. For a fair comparison, we have tested the unfolded networks, LISTA and LAMP, for three different setups: two-, three-, and four-layer designs using their provided implementation. Next, in the experiments of face recognition based on SR, we consider both speed and recognition accuracy of the algorithms as it is performed only for the ℓ_1 -minimization toolbox in [55]. Thus, in order to perform comparative evaluations, the proposed CSEN approach is evaluated against most of the conventional state-of-the-art SR techniques along with ReconNet. Finally, CSEN2 is applied as a preprocessing step for the CS-recovery to obtain \mathbf{w} in the cost function, as illustrated in Fig. 1.

The experiments in this study have been carried out on a workstation that has four Nvidia TITAN-X GPU cards and Intel Xeon CPU E5-2637 v4 at 3.50 GHz with 128-GB memory. Tensorflow library [56] is used with Python. ADAM optimizer [57] is utilized during the training with the proposed default values of the learning parameters: learning rate: $lr = 0.001$ and moment updates: $\beta_1 = 0.9$ and $\beta_2 = 0.999$ with only 100 and 30 backpropagation iterations for MNIST and face recognition experiments, respectively.

A. Experiment I: Support Estimation From CS Measurements

The following metrics are used to report the performance of the proposed and competing methods:

$$\text{F1 Measure (F1-Score)} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (11)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (12)$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (13)$$

where true negatives (TNs), false negative (FN), true positive (TP), and false positive (FP) are calculated between the predicted binary mask $\hat{\alpha}$ and its corresponding ground truth α for each sample in test set. Then, the final reported performance metrics are the averaged ones using the macroaverage method.

For the experiments in this section, the MNIST dataset is used. This dataset contains 70 000 samples (50k/10k/10k as the sizes of the train/validation/test sets) of the handwritten digits (0–9). Each image in the dataset is a 28×28 pixel resolution with intensity values ranging from 0 (black, background) to 1 (white, foreground). Since the background covers more area than the foreground, each image can be considered as a sparse signal. Mathematically speaking, we may assume that the i th vectorized sample, $\mathbf{x}_i \in \mathbb{R}^{n=784}$, can be considered as the k_i -sparse signal. The sparsity rates of each sample are calculated as $\rho_i = (k_i/n)$, and its histogram is given in Fig. 6. We have designed an experimental setup where these sparse signals (sparse in canonical basis) \mathbf{x}_i 's are compressively sensed

$$\mathbf{y}_i = \mathbf{A}\mathbf{x}_i = \mathbf{D}\mathbf{x}_i \quad (14)$$

where $\mathbf{D} = \mathbf{A} \in \mathbb{R}^{m \times n}$ since $\Phi = \mathbf{I}$. We calculate the MR as $\text{MR} = (m/n)$. Therefore, the problem is SE from each CS measurement, i.e., finding $\hat{\Lambda}_i$ from each \mathbf{y}_i in the test dataset.

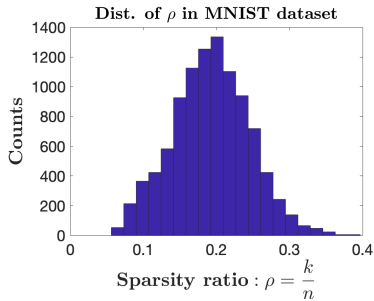


Fig. 6. Histogram of ρ_i 's obtained from the 10k samples (test set). The vectorized gray-scale images, \mathbf{x}_i , in the MNIST dataset are already sparse in the spatial domain (in canonical basis, i.e., $\Phi = I$) with $\|\mathbf{x}_i\| \leq k_i$.

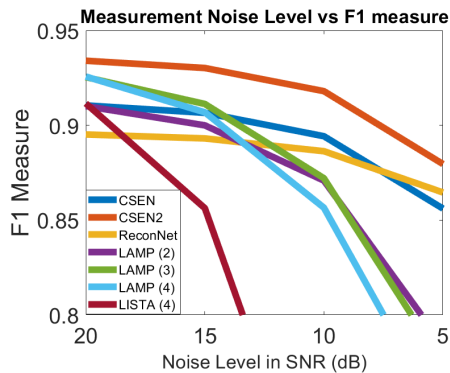


Fig. 7. F1 measure graph of CSEN and LAMP configurations in different noise levels at MR = 0.25.

For this dataset, the **MR** is varied from 0.05 to 0.25 in order to investigate the effect of MR on the SE performance. The measurement matrix is then chosen as the ‘‘Gaussian,’’ and the elements $A_{i,j}$ of the matrix are i.i.d. drawn from $\mathcal{N}(0, (1/m))$. It is worth mentioning that the approximate message passing (AMP) algorithm is a well-optimized method for the Gaussian measurement matrix, and LAMP is a learned version of this algorithm. Therefore, they are reported to be state of the art if the measurement matrix is Gaussian, but they do not even guarantee the converge for other types of measurement matrices. On the other hand, the comparative performance evaluations against LAMP, LISTA, and deep CS-SR methods are presented in Tables I and II, and the results clearly indicate that the proposed method achieves the best SE performance in terms of F1 measure for MR = 0.25 and 0.05 and comparable for MR = 0.1. The results presented in Table I indicate that, despite its deep and complex configuration, compact CSEnS achieve superior performance levels compared to ReconNet. For both LISTA and LAMP, both increasing the layer size from 2 to 4 does not improve their SE performances as it can be observed in Table I. Hence, their numbers of layers are not further increased.

Furthermore, comparative evaluations are performed when the measurements are exposed to noise in the test set, i.e., $\mathbf{y}_i = \mathbf{D}\mathbf{x}_i + \mathbf{z}_i$, where \mathbf{z}_i is an additive white Gaussian noise. The results presented in Fig. 7 show that SE performances of the LAMP and LISTA method are adversely affected by increased measurement noise. Their performance gets even worse when the number of layers is increased [i.e., see results

TABLE I
SUPPORT RECOVERY PERFORMANCE OF ALGORITHMS
FROM THE NOISE-FREE MEASUREMENTS

MR	0.25	0.1	0.05	0.25	0.1	0.05	0.25	0.1	0.05	0.25	0.1	0.05
	F1 Measure			Precision			Recall			CE		
CSEN	0.91	0.85	0.80	0.90	0.84	0.77	0.92	0.87	0.84	0.03	0.06	0.08
CSEN2	0.94	0.89	0.84	0.93	0.88	0.82	0.94	0.90	0.87	0.02	0.04	0.06
ReconNet	0.90	0.85	0.79	0.89	0.82	0.76	0.90	0.87	0.83	0.05	0.06	0.09
LAMP (2)	0.92	0.89	0.82	0.94	0.90	0.82	0.89	0.87	0.83	0.05	0.05	0.08
LAMP (3)	0.93	0.89	0.82	0.95	0.90	0.82	0.91	0.88	0.82	0.03	0.05	0.08
LAMP (4)	0.93	0.90	0.83	0.95	0.92	0.82	0.92	0.89	0.83	0.03	0.04	0.08
LISTA (2)	0.93	0.89	0.82	0.96	0.90	0.82	0.90	0.87	0.82	0.03	0.05	0.07
LISTA (3)	0.93	0.89	0.82	0.95	0.90	0.82	0.90	0.87	0.82	0.03	0.05	0.07
LISTA (4)	0.93	0.89	0.82	0.95	0.90	0.82	0.90	0.87	0.83	0.03	0.05	0.07

TABLE II
SUPPORT RECOVERY PERFORMANCE OF ALGORITHMS
UNDER 10-dB MEASUREMENT NOISE

MR	0.25	0.1	0.05	0.25	0.1	0.05	0.25	0.1	0.05	0.25	0.1	0.05
	F1 Measure			Precision			Recall			CE		
CSEN	0.89	0.82	0.77	0.89	0.82	0.75	0.89	0.82	0.79	0.04	0.07	0.09
CSEN2	0.92	0.86	0.80	0.92	0.86	0.80	0.92	0.86	0.82	0.03	0.06	0.08
ReconNet	0.89	0.83	0.78	0.89	0.81	0.74	0.89	0.85	0.81	0.04	0.07	0.09
LAMP (2)	0.87	0.85	0.79	0.90	0.86	0.78	0.84	0.83	0.80	0.08	0.08	0.10
LAMP (3)	0.87	0.84	0.77	0.91	0.87	0.78	0.84	0.81	0.77	0.06	0.08	0.12
LAMP (4)	0.86	0.85	0.77	0.87	0.87	0.78	0.85	0.82	0.77	0.08	0.07	0.12
LISTA (2)	0.72	0.82	0.79	0.58	0.77	0.76	0.93	0.88	0.82	0.18	0.08	0.10
LISTA (3)	0.68	0.83	0.79	0.54	0.78	0.76	0.94	0.87	0.82	0.22	0.08	0.10
LISTA (4)	0.68	0.82	0.78	0.53	0.77	0.75	0.94	0.88	0.82	0.22	0.08	0.10

for LAMP (2) to LAMP (4) or LISTA (2) to LISTA (4)]. CSEN2, on the other hand, achieves the highest F1 measure for all noise levels.

B. Convolutional Support Estimation-Based Classification (CSEN-C)

As explained in Section III-B, the dictionary-based (representation-based) classification could be seen as an SE problem. Therefore, CSEN presents an alternative and better approach to both CRC and SRC solutions. In this manner, the proposed CSEN approach is evaluated against both CRC and the state-of-the-art SRC techniques recently proposed. The algorithms are chosen by considering both their speed and performance on the SR problem since the speed-accuracy performance of SRC directly depends on the performance of the sparse SR algorithm [55], and there is no unique winner to achieve the top performance level for all databases. The proposed method is, of course, not limited to face recognition but can be applied in any other representation-based classification problem. In Section V-C, we will also consider a new and challenging classification task, Coronavirus disease (COVID-19) recognition from X-Ray Images.

End-to-End Learning of CSEN-Based Classifiers: In dictionary-based classification designs, the samples of a specific class are stacked in the dictionary as atoms with predefined indices, e.g., the atoms belonging to a particular class can be located in a concatenated manner. Consequently, in sparse representation-based classification, instead of using ℓ_1 -minimization in (4), group ℓ_1 -minimization can be introduced as follows:

$$\min_{\mathbf{x}} \left\{ \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \sum_{i=1}^c \|\mathbf{x}_{\mathbf{G}_i}\|_2 \right\} \quad (15)$$

where \mathbf{x}_{Gi} is the group of coefficients corresponds to class i . Hence, the mse cost function in (10) can be modified accordingly

$$E(\mathbf{x}) = \sum_p (\mathcal{P}_\Theta(\mathbf{x})_p - \alpha_p)^2 + \lambda \sum_{i=1}^c \|\mathcal{P}_\Theta(\mathbf{x})_{Gi}\|_2. \quad (16)$$

This modified cost function can be used to achieve a better estimation of the support set. Having this improved estimation, the query class can be obtained. However, having such an intermediate step is also redundant for a classification problem. In this study, we slightly modify the network to make it an end-to-end learning system: to approximate the new cost function defined in (16), a simple average pooling can be applied after the last layer of CSEN, which is then followed by the SoftMax function to produce class probabilities. Therefore, the modified cost function with the cross-entropy loss at the output would be $E(\mathbf{x}) = -\sum_i^c t_i \log(\mathcal{P}_\Theta(\mathbf{x}))$, where t_i and $\mathcal{P}_\Theta(\mathbf{x})$ are the real and predicted values by CSEN, respectively, for class $i \in C$. In this way, the modified network can directly yield the predicted class labels as the output. The pipeline of the proposed end-to-end learning is drawn in Fig. S1 in the Supplementary Material. One may question whether the proposed compact network designs (CSEN1 and CSEN2) are the optimal ones. We also replaced CSEN compact networks with the deeper fully convolutional one, ReconNet, as an alternative network design and report also its performance as a competing method.

1) *Multiclass Classification Problem: Face Recognition via CSEN-C (Experiment II)*: In the face recognition experiments, we have used Yale-B [22] and CelebA [23] databases. In the Yale-B dataset, there are 2414 face images with 38 identities; and a subset of CelebA is chosen with 5600 images and 200 identities. The face recognition experiments are repeated five times with samples randomly selected to build the dictionary, train, and test sets with 32, 16, and 16 and 8, 12, and 8 samples each for Yale-B and CelebA, respectively, for CSEN schemes, and 25% of training data is separated as validation. To have a fair comparison, for CRC and SRC methods, the training set is also included in the dictionary, which are 48 and 20 samples per identity for Yale-B and CelebA, respectively. The selected subset of the CelebA dataset is also different between each repeated run. For the Yale-B database, we use vectorized images in the dictionary. Earlier studies reported that both SRC and CRC techniques achieve a high recognition accuracy of 97%–98%, especially for high MR rate scenarios ($m/d > 0.25$ for $\mathbf{A} \in \mathbb{R}^{m \times d}$). On the other hand, for the CelebA dataset, both CRC and SRC solutions tend to fail when we use raw atoms in the dictionary without extracting descriptive features. This is why, in this study, we propose to use a more representative dictionary. Instead of using raw images, the atoms consist of more descriptive features extracted by a neural network-based face feature extractor in the library [58]. The proposed method is compared against CRC and SRC techniques with the following seven state-of-the-art SR solver: ADMM [59], Dalm [55], OMP [55], Homotopy [60], GPSR [61], L1LS [62], ℓ_1 -magic [63], and Palm [55].

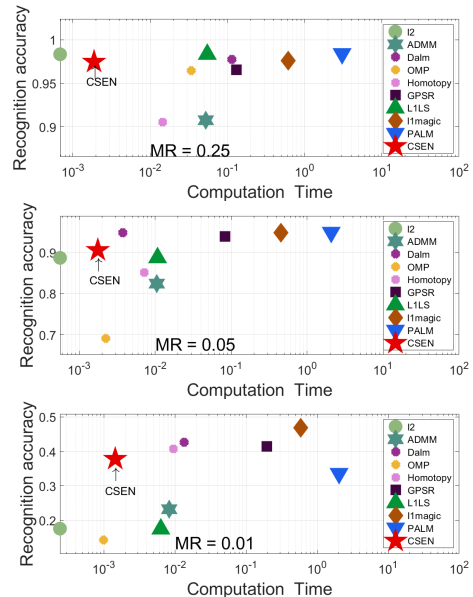


Fig. 8. Recognition accuracy versus process time comparison of algorithms in the Yale-B database.

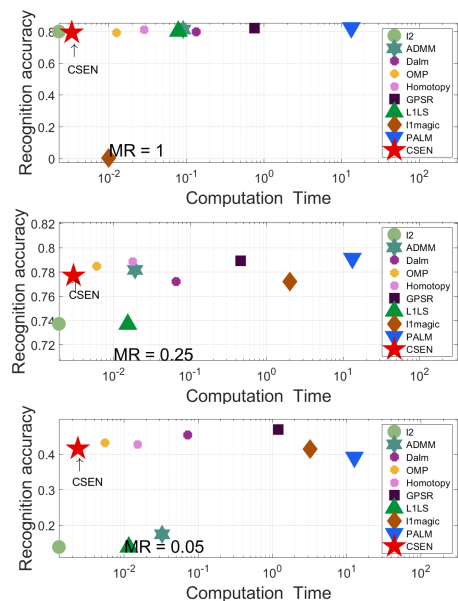


Fig. 9. Recognition accuracy versus process time comparison of algorithms in the CelebA database.

Overall, when we perform experiments in two facial image databases, Yale-B and CelebA for different MRs, the CSEN-based classification proves to be very stable; and in all MRs, it gives the highest or comparable recognition accuracy to the highest ones for all experiments, as presented in Figs. 8 and 9. Furthermore, it is significantly superior in terms of computational speed compared with SRC solutions.

To be able to use the same CSEN designs introduced in Section IV, we reorder the positions of the atoms, i.e., in the representative sparse codes corresponding nonzero coefficients remain next to each other in the 2-D plane. A simplified illustration of the comparison of conventional dictionary design and the proposed design for sparse representation-based

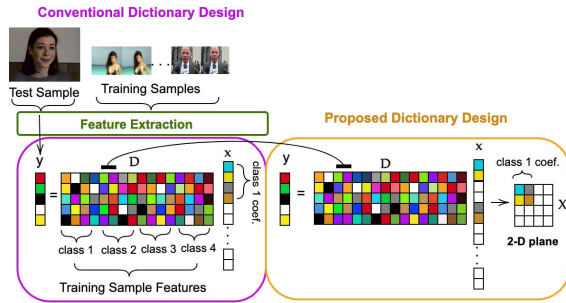


Fig. 10. Graphical representation of proposed dictionary design versus conventional design for face recognition problem.

TABLE III

FOR CSEN-BASED RECOGNITION, THE UTILIZED FACE RECOGNITION BENCHMARK DATASETS ARE GIVEN WITH THEIR CORRESPONDING MASK SIZE AND NUMBER OF SAMPLES IN DICTIONARY, TRAINING, AND TESTING PER CLASS

Dataset	Dictionary Samples	Train Samples	Test Samples	SC Size in 2D-Plane
Yale-B	32	16	16	16 x 76
CelebA	8	12	8	8 x 200

classification is shown in Fig. 10. Defined sparse code sizes and their representations in the 2-D grid for Yale-B and CelebA datasets are also given in Table III.

C. Binary Classification Problem: COVID-19 Recognition From X-Ray Images via CSEN-C (Experiment III)

The recent fast spread pandemic caused by Coronavirus disease (COVID-19) has affected millions worldwide. X-ray imaging is an easily affordable and accessible tool, which provides faster results, compared to other tests that are used in COVID-19 detection. It is well known that deep neural network models achieve state-of-the-art performance results in recognition and detection tasks. However, they require a large number of training samples to achieve a good generalization capability. On the other hand, representation-based classifiers are known to obtain reasonable classification performances with scarce data. In our previous work [25], we showed that CSEN-based classification is effective in recognizing COVID-19 among other classes when the classification problem is multiclass, i.e., COVID-19, bacterial pneumonia, viral pneumonia, and normal (healthy) classes. In the sequel, we will investigate the performance of CSEN-based classification in a binary classification task, that is, COVID-19 differentiation from other classes (control group). In such a sudden outbreak, such as COVID-19, preventing the spread should be a major concern. For this reason, we focus on minimizing FNs (while keeping the FPs as low as possible).

We used a benchmark dataset, Qata-Cov19 [25], of Chest X-Ray images from COVID-19 patients containing 462 samples. The control group (non-COVID class, a Kaggle dataset [64]) consists of 5824 X-Ray images that are 2760, 1485, and 1579 samples from Bacterial pneumonia, viral pneumonia, and normal class, respectively. We used fivefold cross-validation for evaluation, that is, for each fold, a different (20%) portion of the dataset was used as the test set, while the remaining (%80) of the data was used for training. In this

TABLE IV
COVID-19 RECOGNITION PERFORMANCES OF THE ALGORITHMS

	k-NN	SVM	MLP	CRC	SRC	ReconNet	CSENI	CSEN2
Accuracy	0.9446	0.9701	0.9613	0.9623	0.7887	0.9629	0.9618	0.9644
TN	5491	5682	5609	5602	4658	5604	5596	5610
TP	447	416	434	447	300	449	450	452
FP	333	142	215	222	1166	220	228	214
FN	15	46	28	15	162	13	12	10
Sensitivity	0.9675	0.9004	0.9394	0.9675	0.6494	0.9719	0.9740	0.9784
Specificity	0.9428	0.9756	0.9631	0.9619	0.7998	0.9622	0.9609	0.9633
F1-Score	0.7198	0.8157	0.7813	0.7905	0.3112	0.7940	0.7895	0.8014
F2-Score	0.8505	0.8645	0.8690	0.8880	0.4526	0.8919	0.8907	0.8990

way, all classifiers were evaluated over the entire dataset. Specifically, over 6286 total samples, for each fold, 5029 of them are selected for training, and 1257 (1164 samples from the control group and 63 samples from the COVID-19 class) are used as the test set. Data balancing was applied only to the training set, while the test set remained the same. The training set is augmented to have 9320 samples (4660 samples from the control group and 4660 samples from the COVID-19 class) via data balancing. The average performance over the fivefold was reported as the overall performance of each algorithm. The same experiment, with the same partitions, was conducted for all competing algorithms for a fair comparison.

In order to extract discriminative features from raw chest X-ray images, a pretrained model CheXNet [65], which was trained for other types of pneumonia detection from X-Ray images, is used. Using the pretrained CheXNet model, we extracted 1024-long vectors right after the last average pooling layer. After data normalization (zero mean and unit variance), we obtained a feature vector $\mathbf{s} \in \mathbb{R}^{d=1024}$. Then, the PCA matrix \mathbf{A} is applied to the features, i.e., $\mathbf{y} = \mathbf{A}\mathbf{s}$. As competing algorithms to CSEN-based designs, CSENI, CSEN2, and ReconNet, we selected the traditional classifiers KNN, MLP, and SVM, as well as the representation-based classifiers CRC and SRC. For SRC, we only reported the best-performed sparse recovery technique for this classification task, which is DALM. For the competing representation-based classifiers, CRC and SRC, the whole training data are used in the dictionary. On the other hand, for CSEN-based classifiers and ReconNet one, out of total training samples, 3200 samples (1600 samples for each class) are used in the dictionary, and the rest were used to train CSENs. For this smaller dictionary, the sparse code size in the 2-D plane is set as 80×40 .

As it can be observed from Table IV, SRC performance drastically drops for the binary classification task. It is an expected result because the ideal representation coefficient, \mathbf{x} , is not sparse enough (e.g., the sparsity ratio $(k/n) = 0.5$). Although the CSEN-based classifier is also sparsity-driven and favors sparser representation (e.g., multiclass) problems, it still shows superior performance over other representation-based classifiers, which are CRC and SRC. When we compare with other traditional classifiers, the proposed scheme outperforms the second-best performing one, SVM, with respect to the missclassification rate of COVID-19, sensitivity, and F2-Score. These performance metrics can be considered major indicators because we want to achieve the highest sensitivity possible for the minimization of the FNs with a tolerable false

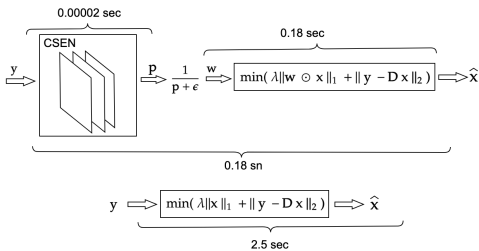


Fig. 11. Top: proposed CS reconstruction. Bottom: traditional ℓ_1 -minimization-based CS-recovery.

alarm rate. On the other hand, if one wants to compare the performance in terms of F1-Score instead of F2-Score, CSEN-based classification still achieves a comparable performance with SVM. F-2 Score is calculated as follows: $F2\text{-Score} = 5 \times (\text{Precision} \times \text{Recall}) / (4 \times \text{Precision} + \text{Recall})$. One may question whether or not the compact CSEN configuration is the optimal one. When we replace the proposed compact network configurations with a deeper well-known network, ReconNet, in the CSEN-based design, no significant performance improvement is observed. In fact, the results are even worse compared to CSEN2 configurations.

D. Learning-Aided Compressive Sensing

1) *Experiment IV: Sparse in Spatial Domain*: As the experimental setup, we randomly choose sparse signals, \mathbf{x} , in the MNIST database and use the Gaussian measurement matrix, \mathbf{A} , to simulate the CS, i.e., $\mathbf{y} = \mathbf{A}\mathbf{x}$. Then, we recover the sparse signal from \mathbf{y} by using the aforementioned state-of-the-art SR tools and the proposed weighted ℓ_1 -minimization [see (6)], where the weights \mathbf{w} are obtained using CSEN output such that $\mathbf{w} = (1/(\mathbf{p} + \epsilon))$. Fig. 11 shows an illustration of how the proposed CS reconstruction scheme differs from the traditional CS recovery setup. Using the output of CSEN as prior information not only provides more accurate SR but also faster convergence of iterative sparse SR such as ℓ_1 -minimization.

Furthermore, we draw the estimated phase transition of the algorithms in Fig. 12 using an experimental setup whose procedure is explained in [19]. Briefly summarizing the procedure, a grid of (MR, ρ) is generated for each algorithm, with 20 independent realizations of the problem: according to their sparsity ratios, ρ , randomly chosen sparse signals \mathbf{x} , among 10000 MNIST test images, are compressively sensed with the independent realization of measurement matrices. Then, they are recovered using the competing algorithms, and each realization is considered a success for the specific algorithm if $((\|\mathbf{x} - \hat{\mathbf{x}}\|_2) / \|\mathbf{x}\|) \leq \text{tol}$, where tol is a predefined parameter; we choose $\text{tol} = 10^{-1}$ in our experiments. For a specific algorithm, we draw the phase transition in the border where a 50% success rate is achieved. The procedure is similar to [19], with the exception that they repeated the experiment only once, while we repeat it 100 times for each method, except L1LS due to its infeasibly high computational cost (it took almost two weeks with an ordinary computer). With an accurate SR algorithm, we expect the transition border to be close to the left-top corner in the phase transition graph because it is a good indicator that the algorithm performs well

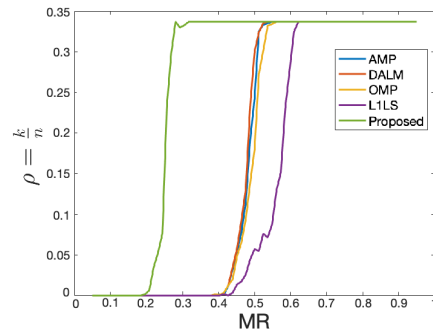


Fig. 12. Phase transition of the algorithms.

in low MRs and with a high sparsity ratio, ρ . From Fig. 12, one can easily deduce that the proposed CS-reconstruction approach clearly outperforms all competing state-of-the-art SR reconstruction methods. Moreover, the two examples where signals are compressively sensed with $MR = 0.25$ and their estimated versions by different SR methods are shown in Fig. 13. It is clear that the proposed approach recovers the sparse signal with the best quality, while the other state-of-the-art SR techniques perform poorly.

2) *Experiment V: Sparse in a Proper Domain (Total Variation for Natural Images)*: In Section V-D1, we assumed that MNIST handwriting signals are sparse in the spatial domain, i.e., $\Phi = \mathbf{I}$. Nevertheless, it is not the case for most of the real-world signals. For instance, natural images are not sparse in canonical basis but sparse in a convenient sparsifying basis, such as DCT and wavelet. In this section, we will use the gradient domain as sparsifying basis, i.e., $\Phi = \nabla$. Mathematically speaking, let we have an image $\mathbf{S} \in \mathbb{R}^{n_1 \times n_2}$ to be compressively sensed with the measurement matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ via $\mathbf{y} = \mathbf{A}\mathbf{s} \in \mathbb{R}^m$, where $\mathbf{s} \in \mathbb{R}^n$ is vectorized image, and $n = n_1 \times n_2$. The image can be sparsely represented in ∇ with sparse code pair $(\mathbf{X}_h, \mathbf{X}_v)$, which are nothing but gradients on the horizontal axis (x -axis) and the vertical axis (y -axis), respectively, i.e., $\nabla_h \mathbf{S} = \mathbf{X}_h$ and $\nabla_v \mathbf{S} = \mathbf{X}_v$. Therefore, one can recover the image from \mathbf{y} by solving the following total variation minimization problem:

$$\min_{\mathbf{S}} \lambda \|\nabla \mathbf{S}\|_{\text{TV}} + \|\mathbf{y} - \text{Avec}(\mathbf{S})\|_2^2 \quad (17)$$

where we use the following anisotropic total variation definition:

$$\|\nabla \mathbf{S}\|_{\text{TV}} = \|\nabla_h \mathbf{S}\|_1 + \|\nabla_v \mathbf{S}\|_1 \quad (18)$$

$$= \sum_{i,j} |S_{i+1,j} - S_{i,j}| + \sum_{i,j} |S_{i,j+1} - S_{i,j}|. \quad (19)$$

TV minimization-based solutions are mostly used in CS reconstruction problems and other inverse imaging systems to better preserve the edges and the boundaries compared to other sparsifying domains, such as DCT. In order to solve the optimization problem in (17), we use one of the state-of-the-art TV minimization solver, TV Minimization by Augmented Lagrangian and Alternating Direction Algorithms (TVAL3) [66]. Similar to Section V-D1, a CSEN can take the proxy of sparse code as input and produces a probability like measure that give a likelihood about the support of the sparse signal. In this TV-based problem, CSEN

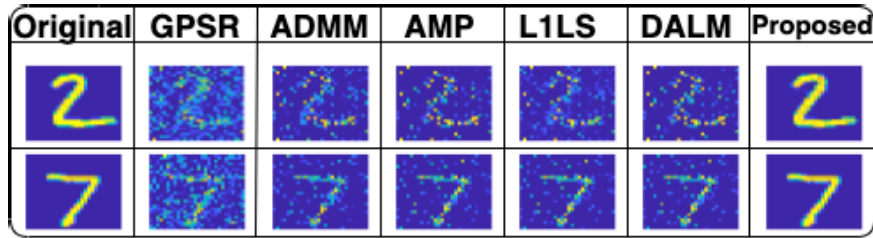


Fig. 13. Examples from MNIST that is compressively sensed and then reconstructed at $MR = 0.25$.

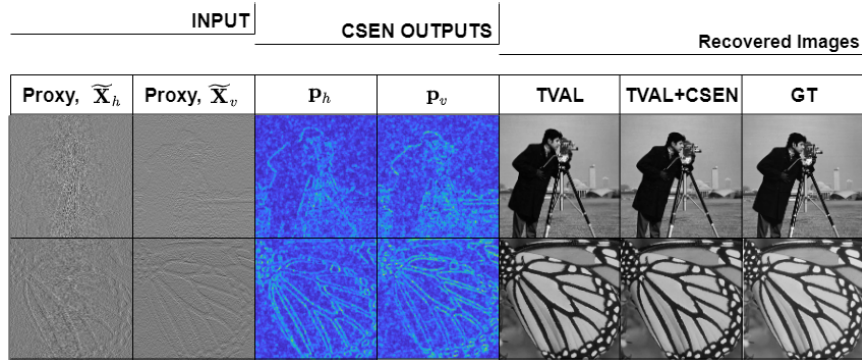


Fig. 14. Examples of the tested natural images. CSEN learns the \mathbf{p} maps from the proxy images in both axes. Then, the \mathbf{p} maps are used in solving the weighted total minimization problem. Traditional TVAL3 solutions have performance of 35.06 (dB) and 22.75 (dB) in PSNR, while CSEN-aided one achieves 36.97 (dB) and 23.68 (dB) in PSNR for butterfly and cameraman images, respectively.

takes two-channel input ($\widetilde{\mathbf{X}}_h = \widetilde{\mathbf{V}}_h \mathbf{S}$, $\widetilde{\mathbf{X}}_v = \widetilde{\mathbf{V}}_v \mathbf{S}$) and produces a two-channel \mathbf{p} -map, $(\mathbf{p}_h, \mathbf{p}_v)$. Example proxy images and CSEN outputs can be seen in Fig. 14. The proxies are obtained by having $\mathbf{A}^T \mathbf{y}$ first and then applying ∇ in both axes after reshaping $\mathbf{A}^T \mathbf{y}$ to the original image dimension, $n_1 \times n_2$. Hereafter, similar to Section V-D1, learning-aided CS recovery can be fulfilled by solving the following weighted TV minimization problem:

$$\min_{\mathbf{S}} \lambda \|\mathbf{W} \odot \nabla \mathbf{S}\|_{\text{TV}} + \|\mathbf{y} - \mathbf{A} \text{vec}(\mathbf{S})\|_2^2 \quad (20)$$

where

$$\|\mathbf{W} \odot \nabla \mathbf{S}\|_{\text{TV}} = \|\mathbf{W}_h \odot \nabla_h \mathbf{S}\|_1 + \|\mathbf{W}_v \odot \nabla_v \mathbf{S}\|_1 \quad (21)$$

and \mathbf{W}_h and \mathbf{W}_v are calculated by using the outputs of CSENs \mathbf{p}_h and \mathbf{p}_v , respectively i.e., $\mathbf{W}_h = (1/(\mathbf{p}_h + \epsilon))$ and $\mathbf{W}_v = (1/(\mathbf{p}_v + \epsilon))$.

In order to solve (20), the same solver can be utilized with the one that is used to solve (17) by only changing the soft-thresholding to the weighted soft-thresholding in the algorithm. In this manner, we use TVAL3 for both problems with the following parameter setup: $\mu = 2^{13}$, $\beta = 2^6$, $\mu_0 = 2^2$, 2^{-2} , $\text{tol} = 10^{-6}$, and $\text{maxit} = 300$.

In order to estimate p -maps, CSEN1 network was trained. The training dataset is prepared in the following manner: 89272 image patches of size 256×256 were randomly cropped from the DIV2K image dataset [67]. During data generation, data augmentation was applied to original DIV2K images with eight different rotations and three different scaling factors. The generated image patches are normalized to have values in $[0, 1]$ -scale. We applied gradient operation to each patch; then, ground-truth support sets were obtained by defining a small threshold to the gradients, i.e., $\Lambda_{GT} = \{i, j \in \{1, 2, \dots, n_1\} \times \{1, 2, \dots, n_2\} : |\nabla S_{i,j}| > \tau_1\}$. We set

$\tau_1 = 0.04$ and $n_1 = n_2 = 256$. Input images were first applied to CS, $\mathbf{y} = \mathbf{A} \text{vec}(\mathbf{S})$; then, proxies are obtained from CS images. Finally, the absolutes of the proxies are given as input and CSEN1 were trained to learn mapping to binary mask \mathbf{v} [defined in (7a) and (7b)]. During training, the batch size was chosen as 8, and CSEN was trained with 100 epoch. The learning rate was set to be 0.001 for the first 50 epochs, and then, it was scheduled to be 0.0001 and 0.00001 for the following 30 and 20 epochs, respectively. To calculate the cost matrices, \mathbf{W} , ϵ is set to be 0.2.

Since applying the Gaussian random measurement matrix, \mathbf{A} , to large-scale signals of size $n = \times 256$ is computationally infeasible, we applied structural measurement matrix. The rows of the measurement matrix can be chosen from a subset of the randomly permuted rows of a basis for which a computationally fast implementation is feasible. We used the Walsh Hadamard transform whose fast implementation is available in the TVAL3 toolbox. By using such a structural \mathbf{A} , the computational complexity of matrix multiplications, e.g., $\mathbf{A} \mathbf{s}$ and $\mathbf{A}^T \mathbf{y}$, can be reduced to $O(n \log n)$ from $O(m \times n)$ compared to using fully random matrices.

In the experimental setup, we tested the possibility to have an improvement in CR recovery with such a learned-aided weighted total variation minimization. The test is conducted on the Set5 [68] image dataset and Barbara and Cameraman images. All the images were resized to be 256×256 size. Along with the PSNR performance metric, the relative error performance metric that was used in the TVAL3 study was also used during the test. The performance metric relative error is calculated as $\text{Relative Error} = (\|\mathbf{S} - \hat{\mathbf{S}}\|_F / \|\mathbf{S}\|_F)$, where $\hat{\mathbf{S}}$ is the estimated image and $\|\cdot\|_F$ is the Frobenius norm. Table V shows the performance comparison of traditional TV minimization and CSEN-aided one. On average, the learning-aided

TABLE V

CS RECOVERY PERFORMANCE OF TV-BASED ALGORITHMS (MR = 0.6)

	TVAL		TVAL + CSEN			
	PSNR	Relative Error	Epoch 50		Epoch 100	
			PSNR	Relative Error	PSNR	Relative Error
Cameraman	35.06	0.034	38.20	0.023	36.97	0.027
Barbara	22.84	0.142	22.56	0.1467	23.33	0.134
Set 5	33.08	0.060	33.73	0.053	33.84	0.0523

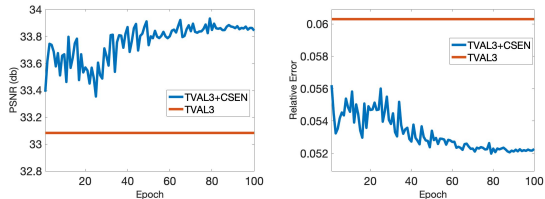


Fig. 15. Performance metrics on the SET 5 image dataset for conventional total variations minimization versus CSEN-aided one. Both techniques use a TVAL3 solver to solve the problem. The results also show when the CSEN trained with 50 epochs and 100 epochs and MR = 0.6.

scheme increases the recovery performance in the Set5 dataset compared to the conventional one and for Barbara and Cameraman Images. To further investigate whether this performance improvement is gained due to arbitrary changes in the threshold values caused by the usage of weighted soft thresholding instead of using soft thresholding, we repeated the same CS recovery tests by using the CSEN1 weights learned with the different number of epochs from 1 to 100. Fig. 15 shows the behavior of the recovery performance of the CSEN-aided solver and the conventional one. The results illustrate that, when CSEN is trained more, the recovery performance of the algorithm increases until convergences. This behavior proves that CSEN output, $(\mathbf{p}_h, \mathbf{p}_v)$, carries information (more activation on nonzero values, e.g., edges and boundaries) to be used in model-based recovery algorithm, TV minimization in that specific case, and the quality of this output determines the image recovery quality.

All in all, this proof-of-the-concept study illustrates that the proposed learned-aided CS recovery scheme has the potential to help model-based solutions for CS imaging systems and worth further investigation.

VI. DISCUSSION

A. Sparse Modeling Versus Structural Sparse Modeling

The first generation CS-recovery or sparse representation methods only use the information that the signal, which we encounter in real life, is sparse in a proper domain or dictionary. These models do not utilize any further assumptions about the sparse signal, \mathbf{x} , in SR or SE. Therefore, they only impose sparsity to the signal to have support set with elements in arbitrary location, i.e., $\min \|\mathbf{x}\|_0$ s.t. $\mathbf{D}\mathbf{x} = \mathbf{y}$. However, most sparse signals that we face in practical applications exhibit a kind of structure. In second-generation sparse representation models, researchers realized that, in addition to arbitrary sparsity, any prior information about the sparse code can be used in modeling more advanced recovery schemes [69], [70]. For instance, the indices of the nonzero wavelet coefficients of an

image mostly exhibit grouping effect [71]. This kind of group sparsity pattern can be imposed by designing the optimization problem involving mixed norm minimization problems [72] instead of simple ℓ_1 -norm. On the other hand, more complex sparsity structures require a more complex model design.

This work proposes an alternative solution to the hand-crafted model-based sparse SR approaches, to be able to learn the pattern inside sparse code (or structural sparse signals), \mathbf{x} by a machine learning technique. This proof of the concept work in which the performance is tested over three real datasets, MNIST, Yale, and CelebA, validates the possibility of such learning and deserves further investigation in different sparse representation problems.

B. Unrolling Deep Models Versus CSEN

The most common approaches to reconstruct sparse signals, \mathbf{x} , from the given measurements, \mathbf{y} , with a fixed dictionary \mathbf{D} can be listed as follows.

- 1) *Convex Relaxation (or ℓ_1 Minimization) Such as Basis Pursuit [29]:* $\min_{\mathbf{x}} \|\mathbf{x}\|$ s.t. $\mathbf{y} = \mathbf{D}\mathbf{x}$ or BPDN [29]: $\min_{\mathbf{x}} \|\mathbf{x}\|$ s.t. $\|\mathbf{y} - \mathbf{D}\mathbf{x}\| \leq \epsilon$, where ϵ is a small constant.
- 2) Greedy algorithms such as matching pursuit (MP) [73], orthogonal MP (OMP) [8], and compressive sampling matched pursuit (CoSaMP) [9].
- 3) Bayesian framework [74].

These conventional algorithms dealing with sparse inverse problems work in an iterative manner; for instance, most convex relaxation techniques, such as BPDN, minimize the data fidelity term (e.g., ℓ_2 -norm) and sparsifying term (e.g., ℓ_1 -norm) in an alternating manner in each iteration. Therefore, these schemes suffer from computational complexity and not suitable for real-time applications.

Along with the traditional approaches listed above, deep learning methods used in this domain have recently become very popular: $\hat{\mathbf{x}} \leftarrow \mathcal{P}(\mathbf{y})$, where \mathcal{P} is a learned mapping from m -dimensional compressed domain to n -dimensional sparse domain. These techniques are built on the idea that the performance of existing convex relaxation can further be improved by reducing the number of iterations and enhancing the reconstruction accuracy. The key idea is that both the possible denoiser matrices, \mathbf{B} (responsible for dealing with data fidelity term), such as \mathbf{D}^T , or $(\mathbf{D}^T\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{D}^T$, where λ is the regularization parameter, and the thresholding values (responsible from sparsifying) can be learned from the training data using a deep network generally with dense layers. For instance, the first example of this type is LISTA [20], which is built upon ISTA [21]. These categories of methods, also called unrolled deep models, design networks in an iterative manner, which are powerful tools for sparse SR.

However, in many practical applications, we may either no need to estimate the sparse signal itself or not have a large amount of training data for deep unrolling networks. In that manner, CSEN provides a third approach, by directly estimating the support set via a compact design, which requires less computational power, memory, and training set. It exhibits very good performance, especially in the problems that include sparse representation with sparse codes having

TABLE VI
1-D VERSUS 2-D CSENS IN THE CELEBA DATASET

MR=0.25 (CelebA)	1-D CSEN1	1-D CSEN2	2-D CSEN1	2-D CSEN2
Accuracy	0.776	0.776	0.777	0.777

structural patterns. The other advantage of the compact design with convolutional layers is that it is more stable against noise compared to unrolled deep models that include dense layers.

C. Proxy Signal Versus Measurement Vector as Input to CSEN

The proposed SE scheme utilizes proxy $\tilde{\mathbf{x}} = \mathbf{B}\mathbf{y}$ as input to convolutional layers. Making inference directly on proxy using the ML approach has been recently reported by several studies. For example, the study in [75] and [76] proposed to perform reconstruction-free image classification on the proxy, and the study in [77] performed signal reconstruction using proxy as an input to a deep fully convolutional network. Furthermore, proxy $\tilde{\mathbf{x}}$ can be learned by fully connected dense layers as presented in [75]. However, this brings additional complexity, and training the network may cause overfitting with a limited number of training data. As in [75], they had to adapt by first training the fully connected layers or try to freeze the other layers during the training.

On the other hand, choosing the denoiser matrix, \mathbf{B} , is another design problem. For example, Değerli *et al.* [75] and Lohit *et al.* [76] use $\mathbf{B} = \mathbf{D}^T$ as denoiser to obtain proxy. We reported the results in this article for denoiser matrix, $\mathbf{B} = (\mathbf{D}^T\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{D}^T$, because it gives slightly more stable performance over \mathbf{B} .

D. 1-D Versus 2-D Representation of the Proxy Signal

In order to use the same CSEN networks, we reshaped the 1-D sparse codes into 2-D for representation-based classification tasks. Nevertheless, a 1-D CNN network structure can also be used. To test this claim, we created 1-D CNN versions of CSEN 1 and CSEN2 networks (the same number of hidden layers, nodes, and kernel sizes). In the CelebA dataset, they were tested, and as it can be seen in Table VI, 1-D versions can also achieve very similar classification performance.

E. Equal Size Dictionary Versus Equal Size Training Samples

In a representation-based classification scheme when dictionary size getting bigger (when the number of training samples is increased), the computational complexity of the method drastically increases. For instance, for the COVID-19 dataset, dictionary size reaches 512×9320 ; in that case, even CRC computational time drastically increases. Fortunately, the computational time of CSEN does not increase that much because only a subset of the training set is used in the dictionary i.e., 512×3200 and the rest to train CSEN. This phenomenon can be seen in Table VII. On the other hand, the recognition performance does not necessarily improve with increased dictionary size; on the contrary, it may even start deteriorating when the dictionary size reaches an impracticable level. SRC is computationally heavier when the dictionary size increases, the computational complexity becomes cumbersome, and the recognition performance does not necessarily

TABLE VII
COMPUTATION TIMES (S) OF EACH METHOD OVER 1257 TEST IMAGES IN THE COVID-19 DATASET

	CRC (light)	CRC	CSEN1	CSEN2	ReconNet	MLP
Computation Time (in sec.)	14.9806	40.3234	0.2439	0.2681	0.3393	0.2848

increase. Furthermore, SRC can completely fail when the representation is not sparse enough, e.g., binary classification (see Table IV). On the other hand, the proposed SE-based classifiers perform very stable for both multiclass or binary classification problems and varying sizes of training datasets.

In the representation-based classification experimental results, the dictionary sizes are always higher than the dictionary size in the CSEN-based scheme, as mentioned above. The other fair comparison is using the same dictionaries for all competing methods. For COVID-19 recognition experiments, as computational times are reported in Table VII, the number of parameters of the networks is given in Table S4 in the Supplementary Material. The CRC algorithm's performance when the dictionary is the same as the one used in the CSEN-C approach is reported in Table S5 in the Supplementary Material. For face recognition, task performance versus computational time is reported in Figs. S2 and S3 in the Supplementary Material. CSEN-C has clear advantages compared to other dictionary-based classifiers by reducing computation complexity and increasing classification accuracy.

VII. CONCLUSION

Sparse support estimators that work based on traditional sparse SR techniques suffer from computational complexity and noise. Moreover, they tend to fail at low MRs completely. The proposed CSENS can be considered as reconstruction-free and noniterative support estimators. Of course, despite their high computational complexity, recent state-of-the-art deep signal reconstruction algorithms may be a cure to sparse recovery methods. However, they are still redundant if SR is not the main concern. In addition, such deep networks often require a large amount of training data that are not available in many practical applications. To address these drawbacks and limitations, in this study, we introduce novel learning-based support estimators that have compact network designs. The highlights of the proposed system are as follows: 1) signal reconstruction-free SE where sparse estimation can be done in a feed-forward manner, noniteratively at a low cost; 2) compact network designs enabling efficient learning even from a small-size training set; and 3) the proposed solution is generic; it could be used in any SE task, such as SE-based classification.

REFERENCES

- [1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [2] E. J. Candès *et al.*, "Compressive sampling," in *Proc. Int. Congr. Mathematicians*, vol. 3, 2006, pp. 1433–1452.
- [3] G. Li, Z. Zhu, D. Yang, L. Chang, and H. Bai, "On projection matrix optimization for compressive sensing systems," *IEEE Trans. Signal Process.*, vol. 61, no. 11, pp. 2887–2898, Jun. 2013.
- [4] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY, USA: Springer, 2010.

- [5] W. Wang, M. J. Wainwright, and K. Ramchandran, "Information-theoretic limits on sparse support recovery: Dense versus sparse measurements," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 2197–2201.
- [6] J. Haupt and R. Baraniuk, "Robust support recovery using sparse compressive sensing matrices," in *Proc. 45th Annu. Conf. Inf. Sci. Syst.*, Mar. 2011, pp. 1–6.
- [7] G. Reeves and M. Gastpar, "Sampling bounds for sparse support recovery in the presence of noise," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 2187–2191.
- [8] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.
- [9] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, May 2009.
- [10] J. Scarlett and V. Cevher, "Limits on support recovery with probabilistic models: An information-theoretic framework," *IEEE Trans. Inf. Theory*, vol. 63, no. 1, pp. 593–620, Jan. 2017.
- [11] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.
- [12] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
- [13] B. Khalfi, B. Hamdaoui, M. Guizani, and N. Zorba, "Efficient spectrum availability information recovery for wideband DSA networks: A weighted compressive sampling approach," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2162–2172, Apr. 2018.
- [14] B. Hamdaoui, B. Khalfi, and M. Guizani, "Compressed wideband spectrum sensing: Concept, challenges, and enablers," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 136–141, Apr. 2018.
- [15] A. C. Gurbuz, J. H. McClellan, and W. R. Scott, "A compressive sensing data acquisition and imaging method for stepped frequency GPRs," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2640–2650, Jul. 2009.
- [16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
- [17] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "ReconNet: Non-iterative reconstruction of images from compressively sensed measurements," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 449–458.
- [18] M. Borgerding, P. Schniter, and S. Rangan, "AMP-inspired deep networks for sparse linear inverse problems," *IEEE Trans. Signal Process.*, vol. 65, no. 16, pp. 4293–4308, Aug. 2017.
- [19] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 45, pp. 18914–18919, Nov. 2009.
- [20] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Haifa, Israel, Madison, WI, USA: Omnipress, 2010, pp. 399–406.
- [21] A. Chambolle, R. A. DeVore, N.-Y. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 319–335, Mar. 1998.
- [22] A. S. Georghiadis, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.
- [23] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.
- [24] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 471–478.
- [25] M. Yamac, M. Ahishali, A. Degerli, S. Kiranyaz, M. E. H. Chowdhury, and M. Gabbouj, "Convolutional sparse support estimator based covid-19 recognition from X-ray images," 2020, *arXiv:2005.04014*. [Online]. Available: <http://arxiv.org/abs/2005.04014>
- [26] O. D. Escoda, L. Granai, and P. Vanderghenst, "On the use of a priori information for sparse signal approximations," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3468–3482, Sep. 2006.
- [27] N. Vaswani and J. Zhan, "Recursive recovery of sparse signal sequences from compressive measurements: A review," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3523–3549, Jul. 2016.
- [28] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization," *Proc. Nat. Acad. Sci. USA*, vol. 100, no. 5, pp. 2197–2202, Mar. 2003.
- [29] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, Jan. 2001.
- [30] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *Comp. Rendus Mathématique*, vol. 346, nos. 9–10, pp. 589–592, May 2008.
- [31] E. Candès and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n ," *Ann. Statist.*, vol. 35, no. 6, pp. 2313–2351, Dec. 2007.
- [32] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [33] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
- [34] M. S. Asif and J. Romberg, "On the lasso and dantzig selector equivalence," in *Proc. 44th Annu. Conf. Inf. Sci. Syst. (CISS)*, 2010, pp. 1–6.
- [35] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. B, Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
- [36] E. J. Candès and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 11, pp. 7235–7254, Nov. 2011.
- [37] N. Durgin et al., "Compressed anomaly detection with multiple mixed observations," in *Research in Data Science*. Cham, Switzerland: Springer, 2019, pp. 211–237.
- [38] M. Yamacc, B. Sankur, and A. T. Cemgil, "Malicious users discrimination in organized attacks using structured sparsity," in *Proc. 25th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2017, pp. 266–270.
- [39] B. Shim and B. Song, "Multiuser detection via compressive sensing," *IEEE Commun. Lett.*, vol. 16, no. 7, pp. 972–974, Jul. 2012.
- [40] O. O. Oyerinde, "Multiuser detector for uplink grant free NOMA systems based on modified subspace pursuit algorithm," in *Proc. 12th Int. Conf. Signal Process. Commun. Syst. (ICSPCS)*, Dec. 2018, pp. 1–6.
- [41] B. Wang, L. Dai, Y. Zhang, T. Mir, and J. Li, "Dynamic compressive sensing-based multi-user detection for uplink grant-free NOMA," *IEEE Commun. Lett.*, vol. 20, no. 11, pp. 2320–2323, Nov. 2016.
- [42] M. Yamaç, M. Orhan, B. Sankur, A. S. Turk, and M. Gabbouj, "Through the wall target detection/monitoring from compressively sensed signals via structural sparsity," in *Proc. 5th Int. Workshop Compressed Sens. Appl. Radar, Multimodal Sensing, Imag.*, 2018, pp. 1–5.
- [43] K. R. Rad, "Nearly sharp sufficient conditions on exact sparsity pattern recovery," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4672–4679, Jul. 2011.
- [44] M. Wainwright, "Information-theoretic bounds on sparsity recovery in the high-dimensional and noisy setting," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2007, pp. 961–965.
- [45] G. Reeves and M. C. Gastpar, "Approximate sparsity pattern recovery: Information-theoretic lower bounds," *IEEE Trans. Inf. Theory*, vol. 59, no. 6, pp. 3451–3465, Jun. 2013.
- [46] E. J. Candès and Y. Plan, "Near-ideal model selection by ℓ_1 minimization," *Ann. Statist.*, vol. 37, no. 5A, pp. 2145–2177, Oct. 2009.
- [47] T. T. Cai and L. Wang, "Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4680–4688, 2011.
- [48] G. Reeves and M. Gastpar, "The sampling rate-distortion tradeoff for sparsity pattern recovery in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 3065–3092, May 2012.
- [49] A. K. Fletcher, S. Rangan, and V. K. Goyal, "Necessary and sufficient conditions for sparsity pattern recovery," *IEEE Trans. Inf. Theory*, vol. 55, no. 12, pp. 5758–5772, Dec. 2009.
- [50] T. Guha and R. K. Ward, "Learning sparse representations for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 8, pp. 1576–1588, Aug. 2012.
- [51] W. Li and Q. Du, "A survey on representation-based classification and detection in hyperspectral remote sensing imagery," *Pattern Recognit. Lett.*, vol. 83, pp. 115–123, Nov. 2016.
- [52] N. Vaswani and W. Lu, "Modified-CS: Modifying compressive sensing for problems with partially known support," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4595–4607, Sep. 2010.
- [53] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, p. 969, 2007.
- [54] D. Donoho and J. Tanner, "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing," *Phil. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 367, no. 1906, pp. 4273–4293, Nov. 2009.
- [55] A. Y. Yang, Z. Zhou, A. G. Balasubramanian, S. S. Sastry, and Y. Ma, "Fast ℓ_1 -minimization algorithms for robust face recognition," *IEEE Trans. Image Process.*, vol. 22, no. 8, pp. 3234–3246, May 2013.
- [56] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: <http://arxiv.org/abs/1603.04467>
- [57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: <http://arxiv.org/abs/1412.6980>

- [58] D. E. King, "Dlib-ml: A machine learning toolkit," *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, Jan. 2009.
- [59] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.
- [60] D. M. Malioutov, M. Cetin, and A. S. Willsky, "Homotopy continuation for sparse signal representation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 5, Mar. 2005, pp. 5–733.
- [61] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 586–597, Dec. 2007.
- [62] K. Koh, S.-J. Kim, and S. Boyd, "An interior-point method for large-scale ℓ_1 -regularized logistic regression," *J. Mach. Learn. Res.*, vol. 8, pp. 1519–1555, Jul. 2007.
- [63] E. Candes and J. Romberg. (2005). *LI-Magic: Recovery of Sparse Signals via Convex Programming*. [Online]. Available: <http://www.acm.caltech.edu/l1magic/downloads/l1magic.pdf>
- [64] P. Mooney. (2018). *Chest X-Ray Images (Pneumonia)*. Kaggle, Marzo. [Online]. Available: <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>
- [65] P. Rajpurkar *et al.*, "CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning," 2017, *arXiv:1711.05225*. [Online]. Available: <http://arxiv.org/abs/1711.05225>
- [66] C. Li, W. Yin, H. Jiang, and Y. Zhang, "An efficient augmented lagrangian method with applications to total variation minimization," *Comput. Optim. Appl.*, vol. 56, no. 3, pp. 507–530, Dec. 2013.
- [67] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 126–135.
- [68] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. Brit. Mach. Vis. Conf.*, R. Bowden, J. Collomosse, and K. Mikolajczyk, Eds. BMVA Press, 2012, pp. 135.1–135.10, doi: [10.5244/C.26.135](https://doi.org/10.5244/C.26.135).
- [69] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2481–2499, May 2012.
- [70] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.
- [71] D. Donoho and G. Kutyniok, "Microlocal analysis of the geometric separation problem," *Commun. Pure Appl. Math.*, vol. 66, no. 1, pp. 1–47, Jan. 2013.
- [72] M. Kowalski and B. Torrèsani, "Structured sparsity: From mixed norms to structured shrinkage," in *Proc. Signal Process. Adapt. Sparse Structured Represent. (SPARS)*, 2009, pp. 1–6.
- [73] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
- [74] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, p. 2346, May 2008.
- [75] A. Değerli, S. Aslan, M. Yamac, B. Sankur, and M. Gabbouj, "Compressively sensed image recognition," in *Proc. 7th Eur. Workshop Vis. Inf. Process. (EUVIP)*, Nov. 2018, pp. 1–6.
- [76] S. Lohit, K. Kulkarni, and P. Turaga, "Direct inference on compressive measurements using convolutional neural networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 1913–1917.
- [77] A. Mousavi and R. G. Baraniuk, "Learning to invert: Signal recovery via deep convolutional networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 2272–2276.



Mehmet Yamaç received the B.S. degree in electrical and electronics engineering from Anadolu University, Eskişehir, Turkey, in 2009, and the M.S. degree in electrical and electronics engineering from Boğaziçi University, Istanbul, Turkey, in 2014. He is currently pursuing the Ph.D. degree with the Department of Computing Sciences, Tampere University, Tampere, Finland.

He was a Research and Teaching Assistant with Boğaziçi University from 2012 to 2017 and a Researcher with Tampere University from 2017 to 2020. He is currently a Senior Researcher with Huawei Technologies Oy, Tampere. He has coauthored the papers nominated for the "Best Paper Award" or the "Student Best Paper Award" in European Workshop on Visual Information Processing (EUVIP) 2018 and European Signal Processing Conference (EUSIPCO) 2019. His research interests are computer and machine vision, machine learning, and compressive sensing.



Mete Ahishali received the B.Sc. degree (Hons.) in electrical and electronics engineering from the İzmir University of Economics, İzmir, Turkey, in 2017, and the M.Sc. degree (Hons.) in data engineering and machine learning from Tampere University, Tampere, Finland, in 2019, where he is currently pursuing the Ph.D. degree in computing and electrical engineering.

Since 2017, he has been a Researcher with the Signal Analysis and Machine Intelligence Research Group, Tampere University, under the supervision of Prof. Gabbouj. His research interests are pattern recognition, machine learning, and semantic segmentation with applications in computer vision, remote sensing, and biomedical images.



Serkan Kiranyaz (Senior Member, IEEE) is currently a Professor with Qatar University, Doha, Qatar. He has coauthored the papers that have nominated or received the "Best Paper Award" in IEEE International Conference on Image Processing (ICIP) 2013, International Conference on Pattern Recognition (ICPR) 2014, ICIP 2015, and IEEE TRANSACTIONS ON SIGNAL PROCESSING (TSP) 2018. He had the Most Popular Articles in the years 2010 and 2016 and the Most Cited Article in 2018 in IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING. From 2010 to 2015, he authored the fourth most-cited article of the *Neural Networks* journal. His theoretical contributions to advance the current state of the art in modeling and representation, targeting high long-term impact, while algorithmic, system-level design, and implementation issues target medium and long-term challenges for the next five to ten years. He, in particular, aims at investigating scientific questions and inventing cutting-edge solutions in "personalized biomedicine," which is in one of the most dynamic areas where science combines with technology to produce efficient signal and information processing systems. He published two books, five book chapters, more than 80 journal articles in high-impact journals, and 100 articles in international conferences. He made contributions to evolutionary optimization, machine learning, biosignal analysis, computer vision with applications to recognition, classification, and signal processing.

Dr. Kiranyaz's research team has won the second and first places in PhysioNet Grand Challenges 2016 and 2017, among 48 and 75 international teams, respectively. In 2019, he won the "Research Excellence Award" and the "Merit Award" of Qatar University.



Moncef Gabbouj (Fellow, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1986 and 1989, respectively.

He was the Academy of Finland Professor from 2011 to 2015. He is currently a Professor of signal processing with the Department of Computing Sciences, Tampere University, Tampere, Finland. His research interests include big data analytics, multimedia content-based analysis, indexing and retrieval, artificial intelligence, machine learning,

pattern recognition, nonlinear signal and image processing and analysis, voice conversion, and video processing and coding.

Dr. Gabbouj is also a member of the Academia Europaea and the Finnish Academy of Science and Letters. He has served as associate editor and a guest editor of many IEEE and international journals and the General Chair of two IEEE Signal Processing Society (SPS) and IEEE Circuits and Systems Society (CAS) Flagship conferences: IEEE International Conference on Image Processing (ICIP) and IEEE International Symposium on Circuits and Systems (ISCAS).