Block-Sparse Coding Based Machine Learning Approach for Dependable Device-Free Localization in IoT Environment

Lingjun Zhao, Huakun Huang, Chunhua Su, Shuxue Ding, Huawei Huang, Zhiyuan Tan, and Zhenni Li

Abstract—Device-free localization (DFL) locates targets without equipping with wireless devices or tag under the Internet-of-Things (IoT) architectures. As an emerging technology, DFL has spawned extensive applications in IoT environment, such as intrusion detection, mobile robot localization, and location-based services. Current DFL-related machine learning (ML) algorithms still suffer from low localization accuracy and weak dependability/robustness because the *group structure* has not been considered in their location estimation, which leads to a undependable process. To overcome these challenges, we propose in this work a dependable block-sparse scheme by particularly considering the *group structure* of signals. An accurate and robust ML algorithm named block-sparse coding with the proximal operator (BSCPO) is proposed for DFL. In addition, a severe Gaussian noise is added in the original sensing signals for preserving network-related privacy as well as improving the dependability of model. The real-world data-driven experimental results show that the proposed BSCPO achieves robust localization and signal-recovery performance even under severely noisy conditions and outperforms state-of-the-art DFL methods. For single-target localization, BSCPO retains high accuracy when the signal-to-noise ratio exceeds -10 dB. BSCPO is also able to localize accurately under most multitarget localization test cases.

Index Terms—Device-Free Localization, Internet of Things, Machine Learning, Block, Sparse Coding, Multiple Targets.

1 Introduction

With the continuous deployment of wireless networks [1], e.g., WiFi, fifth-generation communications (5G), and satellite communications, people will inevitably be covered by wireless signals wherever they are located. The device-free localization (DFL) technology [2], a kind of wireless localization technology, has been therefore acknowledged as an emerging technology for providing high quality-of-service (QoS) in the Internet-of-Things (IoT) environment [3]. Because DFL can locate targets without carrying any attached devices or tags [4], it has spawned a variety of emerging industrial and household applications, such as intrusion detection in security safeguard [5], mobile robot localization [6] in smart factories, and healthcare monitoring of patients and the elderly [7]. Fig. 1 briefly illustrates an application under intrusion detection and tracking scenarios with the DFL-based IoT system. As illustrated in this figure, the wireless sensor networks (WSNs) are deployed in a building to sense data on a target location. After sensing, the data are sent to the edge servers for processing and analyzing. Then, the useful location information is aggregated for user access [8]. The user shown in Fig. 1 could be a security administrator or a house host for security safeguard, a caregiver for healthcare of the elderly, or other similar providers.

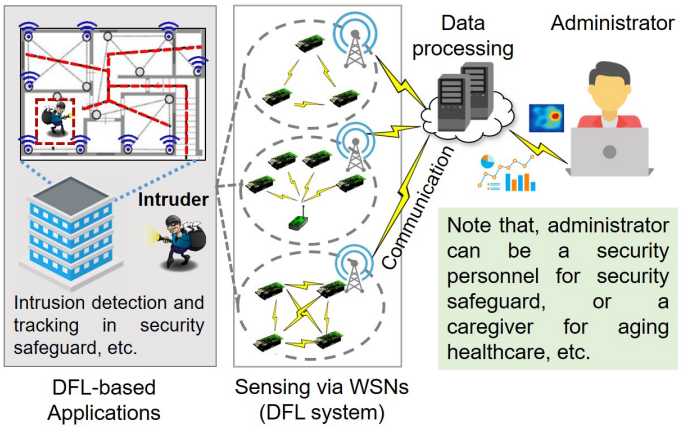

Fig. 1. DFL system-based Internet-of-Things (IoT) network for various applications

As an example, a DFL system is illustrated in step (1) of Fig. 2, in which the detection area is discretized into small grids. Wireless sensor nodes, collaboratively transmitting and receiving wireless signals, are deployed for sensing the

This work was supported by the JSPS Kiban (B) (Project number 18H03240), and in part by the JSPS Kiban (C) (Project number 18K11298), Japan, in part by the National Natural Science Foundation of China (Grant Number: 61902445, 61803096), the Fundamental Research Funds for the Central Universities of China under Grant 19lgpy222, and in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2019A1515011798. Corresponding author: Shuxue Ding (sding@guet.edu.cn); Co-corresponding author:Zhiyuan Tan (z.tan@napier.ac.uk).
Lingjun Zhao, Huakun Huang (Co-first authorship), and Shuxue Ding are with the School of Artificial Intelligence, Guilin University of Electronic Technology, Guilin-City, Guangxi 541004, China.
Chunhua Su is with the School of Computer Science and Engineering, University of Aizu, Aizu-Wakamatsu, Fukushima, Japan.
Huawei Huang is with the School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, 510006, China.
Zhiyuan Tan is with the School of Computing, Edinburgh Napier University, Edinburgh EH10 5DT, U.K.
Zhenni Li is with the Guangdong Key Laboratory of IoT Information Technology, School of Automation, Guangdong University of Technology, China.

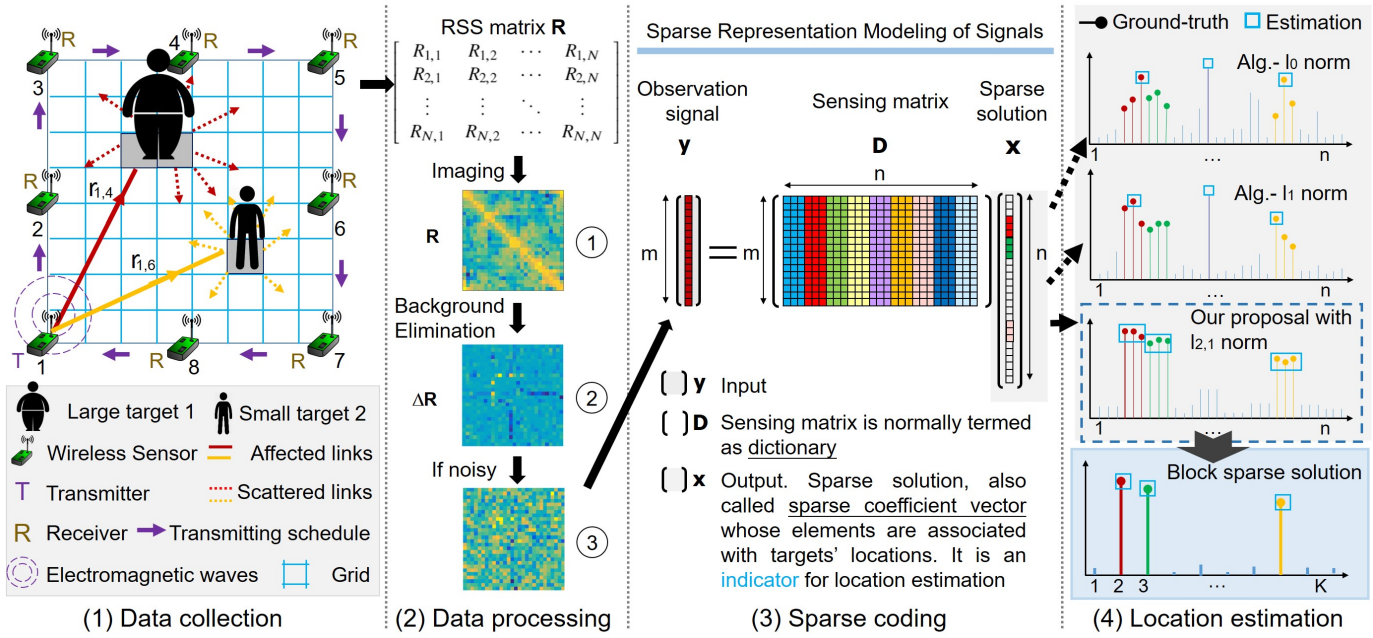

Fig. 2. Illustration of the framework of the proposed block-sparse scheme for DFL. Here, we show an example of two-target localization. K is the grid number.

data. When obstacles (e.g., intruders) appear at the detection area, the energy of the broadcasting signal is attenuated, leading to the signal variations of the transmitting–receiving correspondences. That is, the sensed data conceal the target's location information.

Because sensor nodes generate data with a specific pattern that is associated with a specific target's position and that differs from others, the pattern information can be used for localization. From this perspective, previous studies transformed the localization problem into the classification problem [8]–[10]. As a consequence, to tackle the classification problem, many popular machine learning (ML) methods, such as k-nearest-neighbor (KNN) [11], support vector machine (SVM) [12], and deep learning [13], are employed for accurate single-target localization. However, for multitarget localization problems, many current classification methods, e.g., deep autoencoder and KNN, need to configure complex decision rules and suffer from low localization accuracy.

Fortunately, sparse coding, which is based on the theory of sparse representation in ML field, can improve this situation. It is widely adopted in the DFL field because of its advantages, e.g., high accuracy, high efficiency, simple operation, and the ability to locate multitargets [10].

Specifically, in practical cases, one can assume that the number of locations of targets is far fewer than the number of all the grids of a detection area, therefore the localization problem can be further transformed into a classical *sparse* representation classification (SRC) problem, which can be well solved by the machine-learning algorithm, sparse coding. In step (2) of Fig. 2, the collected data are preprocessed to extract useful pattern information with a distinguished feature. In step (3), we briefly illustrate the basic principle of sparse coding. From step (3), once an observation signal is presented, the sparse solution, whose nonzero-element coordinations indicate the locations of targets, is obtained by sparse coding. If the elements of a sparse solution associated with the locations of targets appear to be nonzero, the locations of the target can be estimated in step (4).

Typically, the sparsity-based DFL approach consists of two stages: an offline stage for constructing the so-called sensing matrix and an online stage for locating targets. The offline stage includes steps (1) and (2) shown in Fig. 2, while the online locating stage mainly comprises steps (3)–(4). These two stages are described in subsection 2.2.2 in detail.

Based on the theory of sparse representation, a sparsest solution constrained by the linear relation $\mathbf{y} = \mathbf{D}\mathbf{x}$ corresponds to the actual true solution [14]. Thus, in an SRC problem, a sparser solution implies a more accurate result. To find a sparse solution with higher sparsity, various regularization terms are employed. From an extensive review of existing DFL-related studies, we find that the most popular regularization terms are ℓ_0 norm and ℓ_1 norm. The relevant literature is briefly summarized below.

1.1 Prior Art

Through a review of the recent literature, we find that, to accurately locate targets, the previous studies have proposed many schemes, such as deep neural networks (DNNs) [18], KNN [11], radio tomographic imaging (RTI) [19], etc., using the RSS or the channel state information (CSI) signal [20]–[22]. Subsequently, to further improve the localization accuracy and performance, many studies exploited the sparse-representation model of signals for DFL [15]–[17].

Zhang et al. [17] employed the compressed sensing (CS) method for sparse target counting and locating in WSNs. They utilized a greedy-matching pursuit (GMP) algorithm to solve the ℓ_1-norm regularized minimization problem, which accurately estimated the locations of targets from a small amount of data. Wang et al. [16] exploited the ℓ_1

TABLE 1
Comparisons with related studies.

Compared terms	Regularization	Definition	Sparse pattern	Sparsity	Localization
Previous studies [9], [10], [15]–[17]	ℓ_0 norm	$\|\mathbf{x}\|_0 = \Sigma_{i=1}^{n} \|x_i\|^0$	Element wise	Individual level	Able
	ℓ_1 norm	$\|\mathbf{x}\|_1 = \Sigma_{i=1}^{n} \|x_i\|^1$	Element wise	Individual level	Able
Proposed scheme	$\ell_{2,1}$ norm	$\|\mathbf{x}\|_{2,1} = \Sigma_{i=1}^{g} \sqrt{x_{i1}^2 + \ldots + x_{iL}^2}$	Joint elements (Block sparse)	Group level	Capable

Note: $\mathbf{x} \in \mathbb{R}^n$ is the sparse solution. $\mathbf{x}$ is divided into g groups and there are L elements for each group. $n = g \times L$.

norm in their CS modeling objective function and employed the Bayesian greedy-matching pursuit (BGMP) method for single-target localization. Ke et al. [15] utilized a basis self-calibration method to solve the problem of basis mismatch in CS-based DFL. In their work, a prior information aided OMP (POMP) algorithm with ℓ_0 norm regularization was proposed to improve reconstruction performance. Wang et al. [9] formulated DFL as an SRC problem and addressed the ℓ_1-minimization problem by sparse coding via a CVX tool to localize a single target. In previous work [10], we exploited achieving an efficient DFL process with sparse coding in the signal subspace. The localization performances of algorithms with ℓ_0 norm and ℓ_1 norm were compared. According to the experiment results of [10], the CVX tool suffered from the limited efficiency with respect to the high-dimensional data, resulting in the incapability of real-time detecting or tracking. Feng et al. [23] utilized the sparse model in a received signal strength (RSS) measurement-based indoor localization system. They recovered sparse signals in a small set of noisy signals by solving a ℓ_1 minimization problem.

In these DFL studies, the objective functions usually comprise two parts:

$$J(\mathbf{x}) = \underset{\mathbf{x}}{\text{argmin}}\ \text{Loss}\ (\mathbf{x})\ +\ \text{Regularizer}\ (\mathbf{x}) \qquad (1)$$

where the loss term is for minimizing the reconstructed error [24] between the estimation signal and the observation signal; the regularization term [25], [26] is for generating sparsity in the sparse solution $\mathbf{x}$. For sparse coding, different sparse patterns derive from different regularization terms [27], [28]. Previous DFL-related work employs ℓ_0 norm or ℓ_1 norm as sparsity regularization, which generates the element-wise sparse pattern at an individual level. Note that ℓ_0 norm and ℓ_1 norm achieves individual variable selection when searches solution at every iteration. However, the natural group-sparse nature in DFL data is not well investigated.

Also, because the element indices of a sparse solution are associated with grid IDs, the ℓ_0 norm- or ℓ_1 norm-based methods usually estimate a target's location by selecting the maximum in a sparse solution. However, in practical scenarios, the targets may have different sizes or body types, such as the two intruders in Fig. 2. A small target may capture only one grid, while a large target may capture multiple grids. Moreover, for high precision, the detection area may be divided into a large number of small grids. In this case, the individual selection of ℓ_0 norm or the ℓ_1 norm results in a low precision with localization error, while a group selection is more applicable to the DFL-based SRC problem because it leads to sparsity at the group level, as shown in the comparisons of step (4). To date, these issues have not been adequately addressed in existing studies.

Additionally, for security, the industrial manager or house host may aim to preserve network-related privacy. An effective way to achieve this is to introduce phase offset or amplitude change in the original signal so that an attacker or an intruder cannot obtain the correct sensing signal. However, this method largely induces signal fading and severely degrades the decoding signal-to-noise ratio (SNR). Therefore, developing a robust/dependable DFL algorithm that can work under such challenging situations with excellent signal-recovery performance has become necessary.

1.2 Our Contributions

In contrast to previous studies, to overcome the discussed challenges, we propose in this work a block-sparse coding scheme. Particularly, by exploiting the $\ell_{2,1}$ norm as the regularization term, the natural group structure of DFL signals will be taken full use to achieve the group selection in the sparse solution. This scheme updating each group as a block is thereby called *block sparse*. Because the $\ell_{2,1}$ norm is nondifferentiable in some places, the $\ell_{2,1}$-norm regularized problem is not straightforwardly solvable. To address this issue, we present a novel algorithm based on the proximal operator method, which is mathematically effective in solving the nonsmooth convex optimization problem in our proposed scheme. Through the proposed algorithm, named block-sparse coding with proximal operator (BSCPO), a block-sparse solution is obtained. We then use the block-sparse solution to locate targets, and this solution is expected to achieve a higher localization accuracy. In Table 1, we summarize the differences between the proposed scheme and related studies for clear comparison.

Furthermore, for privacy preservation, we add Gaussian noise in the original data. With the advantages of the sparse coding algorithm in signal reconstruction, the proposed approach can achieve a robust signal-recovery performance for DFL.

The major contributions of this article can be summarized as follows.

- We propose a dependable block-sparse representation model to extend previous machine learning frameworks for DFL. The proposed block-sparse scheme can achieve the group structure and generate the sparsity at a group level, which can locate a single target as well as multiple targets with different sizes.
- Through a well-designed objective function, the non-smooth convex optimization problem caused by the $\ell_{2,1}$-norm regularization is solved to obtain an equivalent closed-form solution. The localization performance of the proposed scheme has been evaluated on the real-world DFL dataset.
- For privacy preservation in security needs, we add severe Gaussian noise in the DFL data to validate the performance of the proposed BSCPO algorithm in terms of localization accuracy and dependability under severely noisy conditions. Our experimental results show that the proposed method is competitive with state-of-the-art DFL algorithms, especially under heavily noisy conditions.

The remainder of this paper is organized as follows. Section 2 presents the problem formulation and the system model. The design of an objective function and our proposed BSCPO algorithm are shown in Section 3. In Section 4, we carry out the performance evaluation and comparisons. We summarize our contributions in Section 5.

2 Problem Formulation

In this section, we first describe the procedure of background elimination for DFL data, transform the DFL problem into a sparse representation-based image classification problem, and finally, present the sparse model.

2.1 Preliminaries

As shown in the aforementioned step (1) of Fig. 2, for greater accessibility to DFL processing, the detection area is surrounded by several wireless transmitter–receiver nodes and is discretized into a number of grids. The wireless nodes are not required to be regularly distributed, and their positions are not necessarily be known. The target affects the signal broadcasting because of scattering, diffracting, and absorbing, thus varying the RSS measurements at each node. When a target is at a grid, the joint signal configurations comprising specific RSS measurements can be sensed. In addition, RSS signals sensed from the closed target locations usually indicate similar features. Therefore, if the grid captured by a target is treated as a class, the target positioning problem can be converted to a classification problem.

In addition, if we view RSS measurements as image pixels. The collected RSS signals can be constructed to an RSS image with a special patterns. Furthermore, because the signal basis of RSS measurements is much stronger than the signal variation caused by targets, the important characteristics are generally submerged in the raw signals. This results in adverse factors for positioning analysis. To overcome this challenge, we pre-process the raw signals by a background elimination scheme. A clearer RSS variation image is then obtained with distinct features. If the target moves in different grids of detection area, we will image different RSS signal variations with respective features. In a consequence, the target positioning problem is further formulated with the sparse-representation model, which motivates us to tackle it by a sparse coding algorithm.

2.2 Sparse Representation Model of the DFL Signal

In this subsection, the procedure of developing the sparse representation model of DFL is presented.

2.2.1 Data collection and process of background elimination

Here, let $R_{i,j}^{\text{vacant}}$ denote the RSS measurement from the i-th node to the j-th node collected without any target (but can have some static bodies that construct the background) in the detection area, i.e., a vacant area; let $R_{i,j}^{\text{target}}$ denote the measurement with a target existing in the detection area. Then, the background elimination is performed by

$$\Delta R_{i,j} = R_{i,j}^{\text{target}} - R_{i,j}^{\text{vacant}} \tag{2}$$

where $\Delta R_{i,j}$ denotes the variation of the RSS measurement caused by the target.

Assume that the total number of wireless sensor nodes is N. As the illustration in step (1) of Fig. 2, each node transmits wireless signals as a time schedule while the other nodes receive and measure signals. Then, a matrix $\Delta\mathbf{R}$, consisting of the variations of all N nodes, is constructed as

$$\Delta\mathbf{R}_{cl} = \begin{bmatrix} \Delta R_{1,1} & \Delta R_{1,2} & \cdots & \Delta R_{1,N} \\ \Delta R_{2,1} & \Delta R_{2,2} & \cdots & \Delta R_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta R_{N,1} & \Delta R_{N,2} & \cdots & \Delta R_{N,N} \end{bmatrix} \tag{3}$$

where c is the index of the grid captured by the target, and l is the sample index when a target is at the c-th grid. For $\Delta R_{i,j}$, i is the transmitter's ID and j is the receiver's ID.

2.2.2 Dataset construction

There are two stages below:

Offline stage to construct a sensing matrix (called *dictionary* in sparse model). Suppose that there are C grids in the detection area, as shown in Fig. 2, and each grid is regarded as one category. All the potential targets' locations are thereby divided into C classes in this formulation. For each class $c = 1, 2, \cdots, C$, we conduct experiment trials of $l = 1, \cdots, L$ when an object is at the corresponding grid. For each trial, by signal collection and background-elimination process, a signal-variation matrix is obtained, i.e. $\Delta\mathbf{R}_{cl} \in \mathbb{R}^{N\times N}$. We then reshape the variation matrix $\Delta\mathbf{R}_{cl}$ by merging its columns into the vector $\mathbf{d}_{cl}$. Following by stacking all the variation vectors together. We finally obtain the sensing matrix as $\mathbf{D} = [\mathbf{d}_{11}, \mathbf{d}_{12}, \cdots \mathbf{d}_{1L}, \cdots, \mathbf{d}_{c1}, \cdots, \mathbf{d}_{cL}, \cdots, \mathbf{d}_{C1}, \cdots, \mathbf{d}_{CL}]$. This sensing matrix $\mathbf{D}$ contains object's location information of all grids and trials. $\mathbf{D}$ is termed as the ***dictionary*** with size $m \times n$, where $m = N^2, n = C \times L$.

Online stage to process the test signal. We employ similar processes including data collection, background elimination, and matrix-to-vector conversion in this stage. The observed signal $\mathbf{y}$ then obtained while T targets are localized

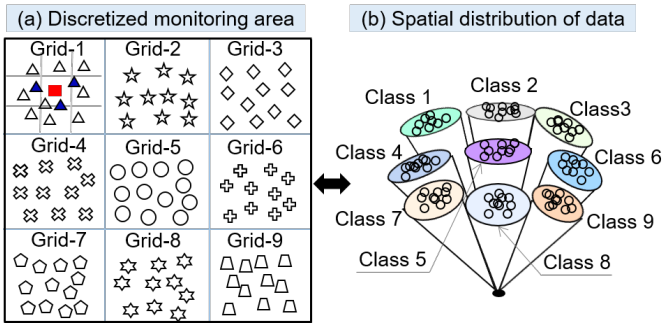

Fig. 3. Illustration of sparse representation classification (SRC) of a test signal. In (a), the empty symbols represent the bases constructed in the offline stage; the red square represents a test signal, whereas the blue triangles are the bases selected for linear representation of the test signal. In (b), each color denotes the high coherence existing among the data of the same class.

at different grids, where T must be smaller than the total grid number. If $T = 1$, it is for single-target localization while, if $T > 1$, it is for multitarget localization.

2.2.3 Sparse representation of the test signal

We now further formulate the localization problem as an SRC problem.

To easily understand the principle of the sparse representation model, let us first consider a simple example in (a) of Fig. 3, where the detection area is discretized into nine grids, i.e., $C = 9$. Thus, the dictionary consists of all samples from nine classes. Because there exists a high coherence among the data of the same class, the spatial distribution of data emerges as (b) for each class. This leads to the phenomenon that the test signal is preferred to select the base samples in the same class to represent itself. Therefore, as shown in (a), when the test signal belongs to the first class, the samples marked with blue triangles are selected to represent the test signal. Here, the red square indicates the test signal, and the other symbols are samples of the dictionary.

According to 2.2.2, in the online stage, when targets locate at the detection area, a test signal $\mathbf{y}$ is obtained. Here, we take two targets as an example. Suppose that the targets are at the p-th grid and q-th grid, respectively. If sufficient samples are given, the test signal $\mathbf{y}$ can be linearly represented with the two corresponding sample sets of dictionary $\mathbf{D}$. The linear representation is given as

$$\begin{aligned}\mathbf{y} &= \mathbf{D}\mathbf{x} + \mathbf{e} \\ &= \sum_{c=1}^{C}\sum_{l=1}^{L} \mathbf{d}_{cl}x_{cl} + \mathbf{e} \\ &= \sum_{c=p}^{}\sum_{l=1}^{L} \mathbf{d}_{pl}x_{pl} + \sum_{c=q}^{}\sum_{l=1}^{L} \mathbf{d}_{ql}x_{ql} + \sum_{o=1}^{C}\sum_{l=1}^{L} \mathbf{d}_{ol}x_{ol} + \mathbf{e} \\ &\text{for } 1 < p < C,\ 1 < q < C, o \neq p \neq q, \sum_{o=1}^{C}\sum_{l=1}^{L} \mathbf{d}_{ol}x_{ol} = 0\end{aligned} \tag{4}$$

where $\mathbf{x} = [0 \cdots 0\ x_{p1} \cdots x_{pL}\ 0 \cdots 0\ x_{q1} \cdots x_{qL}\ 0 \cdots 0]^T \in \mathbb{R}^n$ is a vector comprised of coefficients; x_{pj} and $x_{qj} \in \mathbb{R}$ (for $1 < j < L$) are the nonzero coefficients belonging to the p-th class or the q-th class, respectively; and $\mathbf{e}$ indicates noise.

In summary, the observation signal $\mathbf{y}$ can be sparsely represented in terms of the n base samples of the dictionary $\mathbf{D}$. From this perspective, Eq. (4) becomes a sparse representation problem where $\mathbf{x}$ is a sparse coefficient vector whose nonzero elements are associated with the locations of targets. Based on Eq. (4), once the sparse solution (i.e., sparse coefficient vector) $\mathbf{x}$ is effectively determined, the accurate locations of the target are estimated. Therefore, the DFL problem is essentially an SRC problem and can be well formulated with the sparse representation model.

3 Our Approach

In this section, we first summarize the challenges of sparse coding in existing work. We then propose our solution by devising a new objective function. Finally, to address such an objective function, a block-sparse coding algorithm is presented in detail.

3.1 Existing Challenges and Our Proposed Solution

3.1.1 Existing challenges in sparse coding

Sparse coding is for finding a potential vector variable that contains a small portion of non-zero valued components, termed as sparse, based on a dictionary $\mathbf{D}$ the and observed signal $\mathbf{y}$. In practical applications, to achieve an accurate or robust DFL, a large number of samples are required in the step of constructing a dictionary. This results in a high possibility that the total number of base samples n is larger than m of the dictionary. Thus, Eq. (4) is usually an undetermined system, and its solution is not unique, i.e., the problem is ill-posed.

However, by selecting the sparsest solution, we can make the problem well-posed. This inspired us to solve the optimization problem with the sparsity regularization method for finding the sparsest solution. The most popular sparse regularization terms are ℓ_0 norm and ℓ_1 norm. Unfortunately, using ℓ_0 norm to find the sparsest solution of a linear equation in the undetermined system has been proven to be NP-hard. ℓ_1 norm treats the sparse mode as a singleton independently and generates the sparsity at an individual level. In contrast, since the close locations of targets derive the inner group structure of DFL data, individual selection of ℓ_1 norm and ℓ_1 norm suffers from reduced localization precision.

3.1.2 Proposed solution – A new objective function with block-sparse mode

To determine the sparsest solution, in this paper, we exploit $\ell_{2,1}$ norm as the regularization term to achieve the group selection, generating a block-sparse mode in the sparse solution. The objective function is given as

$$J(\mathbf{x}): \quad \mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_{2,1} \tag{5}$$

where the first term $\|\cdot\|_2^2$ is a measure of the fitting error between the observation signal and estimated signal; the second term with $\ell_{2,1}$ norm is a sparsity regularizer that generates the group sparsity; and λ is a scalable parameter that trades off the fitting error and the sparsity.

3.2 Localization Algorithm with Block-Sparse Coding

3.2.1 Block-sparse coding via proximal operator

Because Eq. (5) is convex although not smooth, it admits the global optimum. Normally, the subgradient method is adopted for solving Eq. (5). By incorporating the derivative of Eq. (5) to zero, we can obtain the minimizer $\mathbf{x}^*$ as

$$\mathbf{x}^* = (\mathbf{D}^T\mathbf{D} + \lambda\frac{\partial(\|\mathbf{x}\|_{2,1})}{\partial\mathbf{x}})^{-1}\mathbf{D}^T\mathbf{y}. \tag{6}$$

However, processing Eq. (6) by the subgradient method is not easy because $(\mathbf{D}^T\mathbf{D})^{-1}$ involves high computational and memory cost as $\mathbf{D}$ is generally with high dimension. In addition, $(\mathbf{D}^T\mathbf{D})^{-1}$ may be very ill-conditioned or may not be invertible. To address this problem, we exploit the proximal operator method for searching the sparse solution. The proximal operator is an efficient method for solving the nonsmooth and nondifferentiable optimization problem. For different objective functions, a specific proximal operator can be derived. For the optimization process of nonsmooth $\|\cdot\|_{2,1}$, a normal formula is given below

$$P(\mathbf{v}): \quad \underset{\mathbf{v}}{\operatorname{argmin}}\ \eta\|\mathbf{u}-\mathbf{v}\|_2^2 + \gamma\|\mathbf{v}\|_{2,1}. \tag{7}$$

To coincide with Eq.(7), we make a modification based on (5). The processes of modification and derivation are given in the APPENDIX section in detail. Then, let us consider the minimization of an objective function simpler than Eq. (5):

$$P(\mathbf{x}): \quad \underset{\mathbf{x}}{\operatorname{argmin}}\ \frac{\mu}{2}\|\mathbf{b}-\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_{2,1} \ + \ K \tag{8}$$

where K is treated as a known constant because it does not rely on $\mathbf{x}$; μ is a scaling factor that should be set larger than the largest eigenvalue e of $(\mathbf{D}^T\mathbf{D})$, e.g., $1.01\times e$; and $\mathbf{b}$ is an intermediate variable with the same size to $\mathbf{x}$, given as:

$$\mathbf{b} = \mathbf{x}^{(k)} + \frac{1}{\mu}\mathbf{D}^T(\mathbf{y}-\mathbf{D}\mathbf{x}^{(k)}) \tag{9}$$

$$\begin{aligned}\mathbf{x}^{(k+1)} &= \operatorname{prox}_{\|\cdot\|_{2,1}}(\mathbf{b}) \\ &= \operatorname{prox}_{\|\cdot\|_{2,1}}(\mathbf{x}^{(k)} + \frac{1}{\mu}\mathbf{D}^T(\mathbf{y}-\mathbf{D}\mathbf{x}^{(k)}))\end{aligned} \tag{10}$$

where $\mathbf{x}^{(0)} = \mathbf{0}$; $\operatorname{prox}_{\|\cdot\|_{2,1}}(\cdot)$ is the proximal operator of $\ell_{2,1}$ norm, given by

$$\operatorname{prox}_{\|\cdot\|_{2,1}}(b_i) = \begin{cases}(\|\mathbf{b}_i^c\|_2 - \frac{\lambda}{\mu})\frac{b_i}{\|\mathbf{b}_i^c\|_2}, & \|\mathbf{b}_i^c\|_2 > \frac{\lambda}{\mu} \\ 0, & \text{Otherwise}\end{cases} \tag{11}$$

where b_i is the i-th element of $\mathbf{b}$ and $\mathbf{b}_i^c \in \mathbb{R}^L$ denotes the c-th group of vector $\mathbf{b}$. The optimization process is mainly based on Eq. (9) and Eq. (10) in each iteration. Until convergence or the stop rule is satisfied in the k-th iteration, the optimally sparse solution x^* can be determined as $\mathbf{x}^{(k+1)}$.

3.2.2 Target localization based on the block-sparse solution

From the presented analysis, the optimally sparse solution $\mathbf{x}^*$ can be obtained by performing Eqs (8)– (11). Here, there are two cases, i.e., if $\mathbf{x}^*$ includes only one group of nonzero elements, it is for single-target localization; if $\mathbf{x}^*$ includes multiple groups of nonzero elements, multi-target localization occurs.

Algorithm 1 Block Sparse Coding with Proximal Operator

Input: $\mathbf{y} \in \mathbb{R}^m$, $\mathbf{D} \in \mathbb{R}^{m\times n}$, μ, λ, $\mathbf{x}_0 = \mathbf{0}$
Output: $\varphi_1, \ldots, \varphi_S$

1: **for** $k = 0$ to maxiteration **do**
2: $\quad\mathbf{b} \leftarrow \mathbf{x}^{(k)} + \frac{1}{\mu}\mathbf{D}^T(\mathbf{y} - \mathbf{D}\mathbf{x}^{(k)})$
3: $\quad\mathbf{x}^{(k+1)} \leftarrow \operatorname{prox}_{\|\cdot\|_{2,1}}(\mathbf{b})$
4: $\quad$Until the convergence criterion is met or the maximum iteration number is reached.
5: **end for**
6: $\mathbf{x}^* \leftarrow \mathbf{x}^{(k+1)}$
7: $\beta^* \leftarrow \mathbf{x}^*$ according to Eq. (12)
8: Estimated the locations of targets are at the φ_1-th, φ_2-th, ..., and the φ_S-th grids by Eq. (13).
9: Return $\{\varphi_1, \ldots, \varphi_S\}$

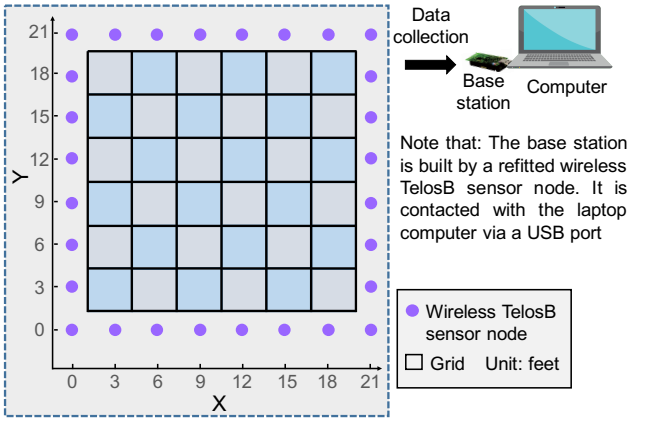

Fig. 4. Experimental setup of the DFL system illustrated according to the SPAN Lab of the University of Utah [29].

For an easy operation process of location estimation, we convert $\mathbf{x}^*$ into a block-sparse solution β^*:

$$\begin{aligned}\beta^* &= \left[\sqrt{\Sigma_{l=1}^L x_{1l}},\ \sqrt{\Sigma_{l=1}^L x_{2l}},\ \cdots,\ \sqrt{\Sigma_{l=1}^L x_{Cl}}\right] \\ &= \left[\|\mathbf{x}_{(1)}^*\|_2,\ \|\mathbf{x}_{(2)}^*\|_2,\ \cdots,\ \|x_{(C)}^*\|_2\right] \\ &= \left[\beta_1^*,\ \beta_2^*,\ \cdots,\ \beta_C^*\right]\end{aligned} \tag{12}$$

where $\mathbf{x}_{(c)}^*$ (for $1 \le c \le C$) denotes the c-th group of elements $\{x_{c1}, \cdots, x_{cL}\}$. Because the elements of β^* are associated with the grids, for the total C grids of a detection area, the locations of targets are estimated as follows. Suppose that, there are S targets. Then, the locations of S targets can be estimated at the φ_1-th grid, ..., the φ_S-th grid, where $\varphi_1, \ldots, \varphi_S$ are the subscripts of the nonzero elements with decreasing order. Then, the locations of targets are given by

$$\begin{cases}\varphi_1 = \underset{\varphi_1}{\operatorname{argmax}}\{\beta_1^*, \cdots, \beta_{\varphi_1}^*, \cdots, \beta_C^*\} \\ \varphi_2 = \underset{\varphi_2}{\operatorname{argmax}}\{\beta_1^*, \cdots, \beta_{\varphi_2}^*, \cdots, \beta_C^*\}, \text{ for} \\ \quad \{\beta_{\varphi_1}^*\} \not\subset \{\beta_1^*, \cdots, \beta_{\varphi_2}^*, \cdots, \beta_C^*\} \\ \quad \ldots \\ \varphi_S = \underset{\varphi_S}{\operatorname{argmax}}\{\beta_1^*, \cdots, \beta_{\varphi_S}^*, \cdots, \beta_C^*\}, \text{ for} \\ \quad \{\beta_{\varphi_1}^*, \ldots, \beta_{\varphi_{N-1}}^*\} \not\subset \{\beta_1^*, \cdots, \beta_{\varphi_S}^*, \cdots, \beta_C^*\}.\end{cases} \tag{13}$$

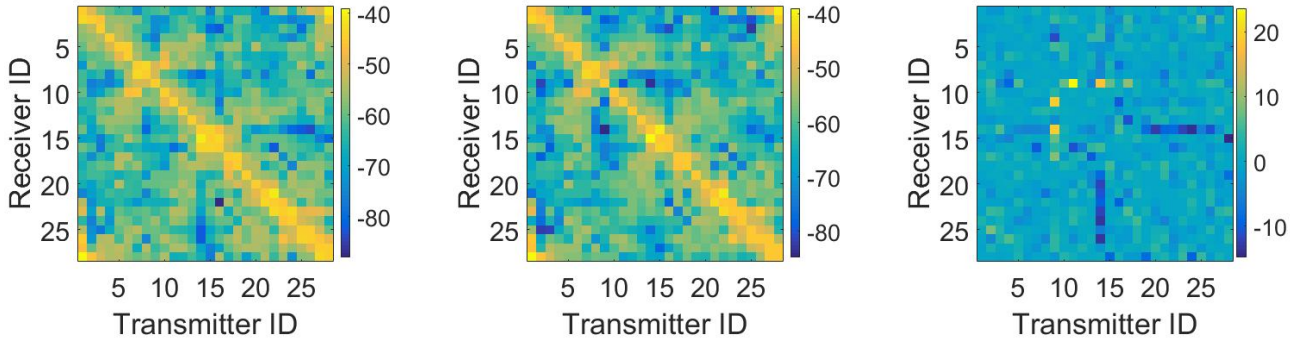

Fig. 5. Example of data preprocessing with the background elimination scheme. Note that the signal variation of (c) is calculated by subtracting the signal of (b) from (a). Here, the target locates at the 36-th grid.

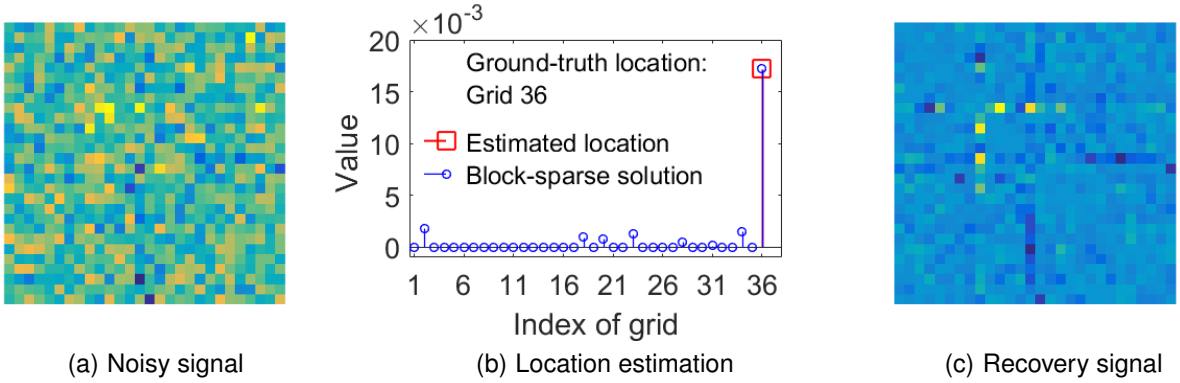

Fig. 6. Imaging the noiseless test signal and noisy test signal; location estimation with block-sparse solutions and recovery signals of the proposed BSCPO algorithm. Here, we present an example when the target is at the 36-th grid of the detection area. The noise level of (c) is SNR = -5 dB.

The proposed localization algorithm is called BSCPO, and its pseudo-code is presented in Algorithm 1.

4 Performance Evaluation

In this section, we evaluate the performance of the proposed approach on the real-world experimental dataset [29] of outdoor DFL. All of the following experiments were performed in MATLAB R2016b and executed on a computer of Windows 10 64-bit with 8GB RAM and Intel Core™ i7 CPU.

4.1 Experiment Explanation

The experiments are performed, as shown in Fig. 4, with the following settings according to [29]. The square detection area, with a 21x21 foot square, is discretized into 36 grids and surrounded by 28 TelosB sensor nodes. Each sensor node works in a frequency band of 2.4 GHz. The interval distance is 3 feet between two neighboring nodes, and each node is placed 3 feet off the ground. A base station listens to the whole network traffic for delivering the collected real-time data to the computer by a USB port. At each grid, 30 trials were conducted with a short time interval.

For the single-target localization experiments, the RSS sample matrices are partitioned into two portions, of which 25 samples are used to construct the dictionary, while the rest are test samples; for the multitarget localization experiments, it shares the same dictionary with the single-target experiments, while the test signals are from the new RSS samples of locating two targets. The dictionary size is 784× 900, while each test sample is 784× 1.

4.1.1 Data preprocessing of background elimination

Because the original RSS signal is measured directly from the environment, it may include many useless signal components that are stronger than the useful signal variation. For example, the base signal of the background does not change even if there are targets in the detection area. Therefore, preprocessing of background elimination is necessary. In the experiments of outdoor DFL, we perform data preprocessing by subtracting $\mathbf{R}^{\text{vacant}}$ from $\mathbf{R}^{\text{target}}$, as shown in Fig. 5. Then, after this preprocess, most of the useless signal components can be eliminated, and the signal variation is apparent.

4.1.2 Compared methods

ℓ_0 norm and ℓ_1 norm are the most popular sparse regularization terms in the DFL fields, and many related references have reported their effectiveness [9], [10], [30], [31]. Hence, we compare the performance of the proposed BSCPO with the algorithms structured by ℓ_0 norm and ℓ_1 norm. Furthermore, to demonstrate the enhanced performance of the proposed block-sparse scheme, we further compare it with multiple baselines and state-of-the-art machine-learning algorithms, including a deep neural network with convolutional autoencoder (CAE), one-dimensional convolutional neural network (CNN-1D), SVM, KNN, sparse coding with

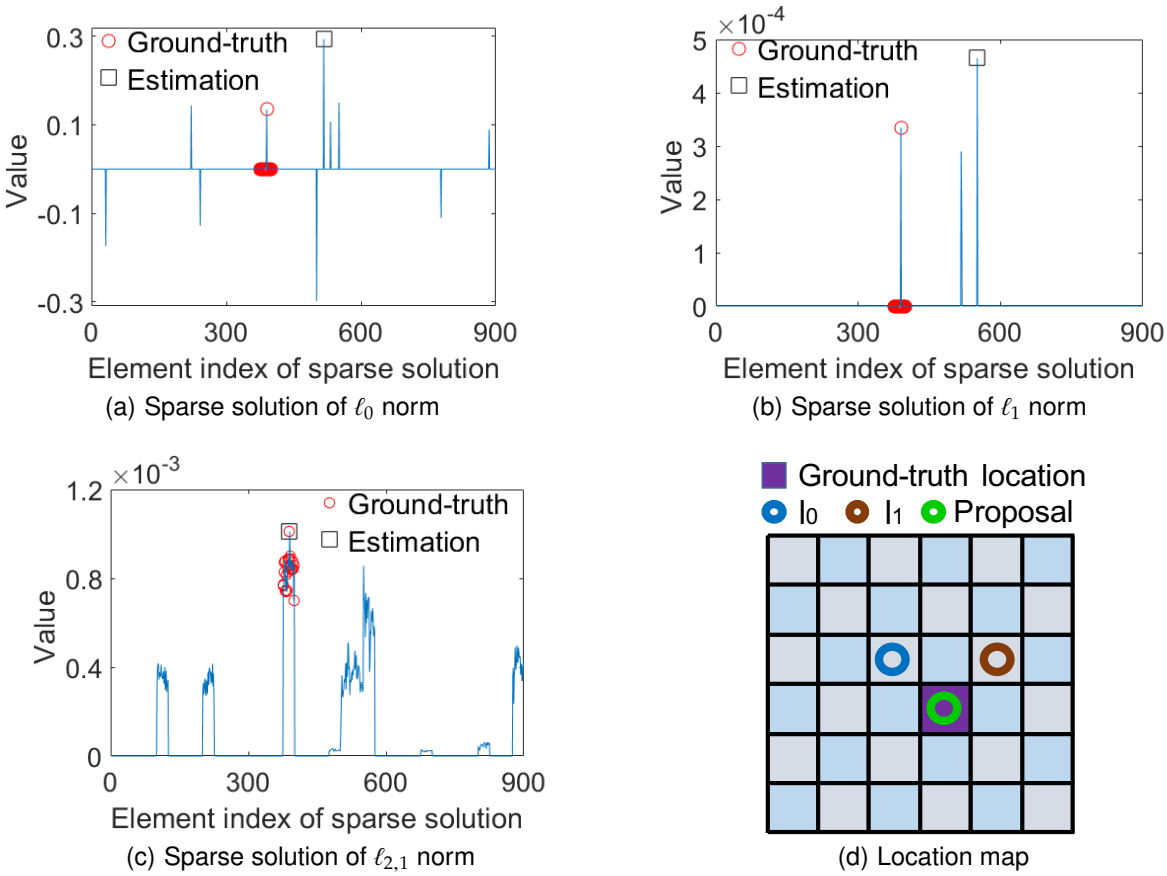

Fig. 7. Performance comparison of algorithms with $\ell_{2,1}$ norm, ℓ_1 norm, and ℓ_0 norm with a noisy test signal of SNR = -15 dB. Here, the ground-truth location of the target is at grid 16, associated to the element index number from 376 to 400, and highlighted in red in the above subfigures. The estimated location is marked with a black square. For (a), ℓ_0 norm is incorrectly located at grid 21; (b) ℓ_1 norm is incorrectly located at grid 23; (c) $\ell_{2,1}$ norm is successfully located at grid 16; and (d) results of (a)–(c).

iterative shrinkage-thresholding algorithm (SC-ISTA), and sparse coding with orthogonal matching pursuit (SC-OMP).

4.1.3 *Other settings and metrics*

To clarify the performance evaluation and comparison, the localization accuracy is used as a metric to evaluate the performance of algorithms. This refers to the percentage of the count of correctly estimated samples with respect to the count of all test samples.

For privacy preservation, we add severe Gaussian noise in the original RSS signals to prevent network-related privacy from leaking to intruders or attackers. In practical applications, the DFL system is unavoidably disturbed by environmental noise, e.g., electromagnetic interference caused by surrounding wireless devices. Therefore, the signal-recovery performance, robustness and dependability are important for DFL algorithms. We use SNR as a measurement of the signal quality. SNR is defined as $\mathrm{SNR(dB)} = 10\log_{10}(\mathrm{P_{signal}}/\mathrm{P_{noise}})$, where P_{signal} and P_{noise} represent the power of signal and the power of noise, respectively.

4.2 Experimental Result and Discussion

In this subsection, we perform validation experiments to evaluate the performance of the proposed block-sparse scheme. We 1) show the performance for locating a single target, 2) show the performance for locating two targets, and 3) provide a comparison with the baselines and state-of-the-art DFL methods.

4.2.1 *Performance of the proposed approach for single-target localization*

The proposed approach can achieve robust performance even when the DFL system is disturbed by severe noise. Fig. 6 shows an example when the target is at the 36-th grid of the detection area. From Fig. 6 (a), the test signal is more severely polluted by noise than the original noiseless signal of Fig. 5 (c). Despite this adverse condition, through the block-sparse solution, the target can still be accurately located because the related element of block-sparse solutions is outstandingly nonzero (see Fig. 5 (b)). This can also be explained by the recovery result of Fig. 6 (c) in which the test signal is clearly reconstructed.

Fig. 7 shows an example to explain the reason why our proposal is more capable of DFL than the algorithms with ℓ_0 norm and ℓ_1 norm. Through the sparse solutions of ℓ_0 norm or ℓ_1 norm, the target location is incorrectly estimated because the element value of another group is larger than the ground-truth value (see Fig. 7 (a), (b), and (d)) because of the noise. In contrast, using the group-sparse solution of BSCPO, the target can be correctly located because the maximum element coincides with the ground truth (see Fig. 7 (c) and (d)).

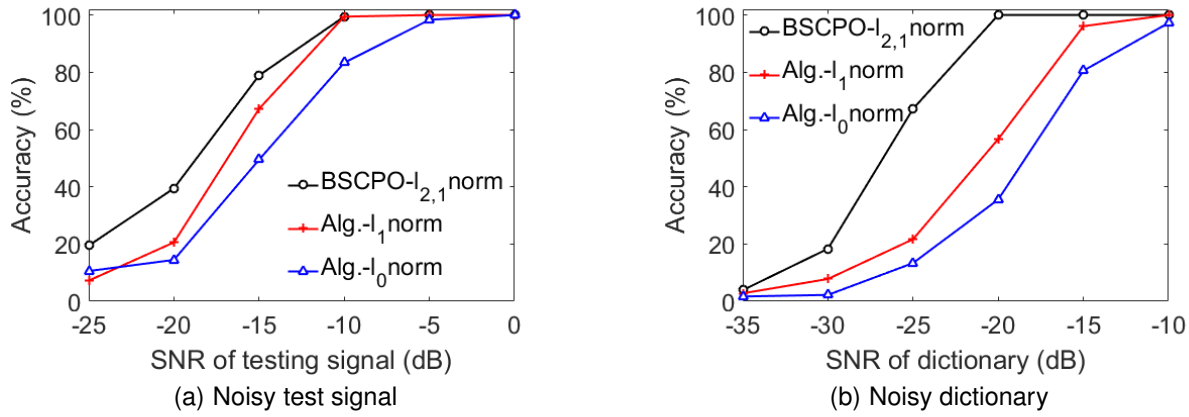

Fig. 8. Localization accuracy of algorithms with $\ell_{2,1}$ norm, ℓ_1 norm, and ℓ_0 norm on noisy test samples. Here, Alg. is short for algorithm.

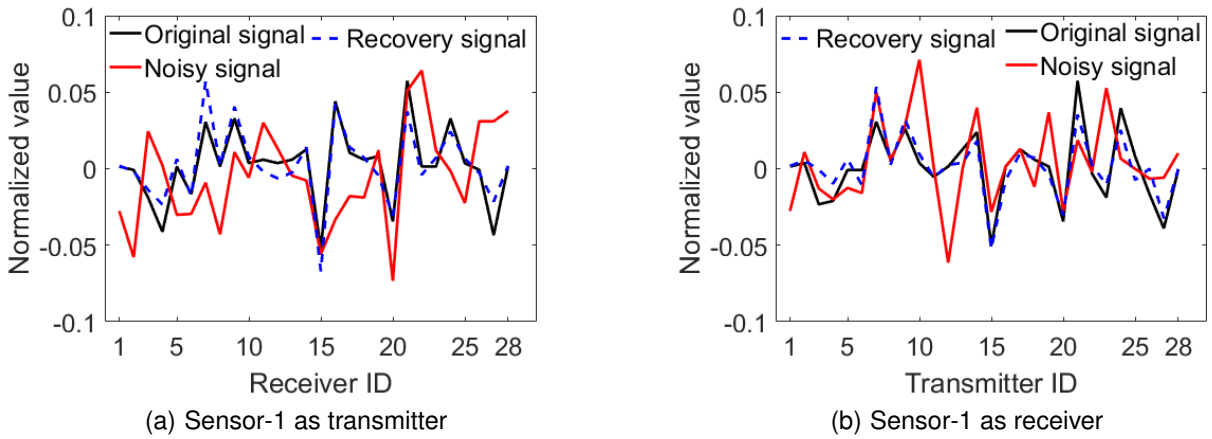

Fig. 9. Signal-recovery performance of the proposed BSCPO algorithm. Here, wireless sensor-1 is taken as an example. SNR = -5 dB.

TABLE 2
Comparison of the localization performance with other machine-learning algorithms

Compared terms	Noisy dictionary (-10 dB)	Noisy testing signal (-5 dB)
KNN [10]	2.8% (Failed)	2.8% (Failed)
SC-OMP [30]	2.8% (Failed)	2.8% (Failed)
SC-ISTA [10]	4.4% (Failed)	11.1%
SVM [32]	5.5% (Failed)	90.6%
CNN-1D [33]	57.8%	65.8%
Deep CAE [8]	85.0%	94.0%
The proposed	**100%**	**100%**

After considering all the test samples, Fig. 8 presents the localization accuracy of BSCPO and comparisons under various noisy conditions. From Fig. 8, the proposed BSCPO outperforms the algorithms with ℓ_0 norm and ℓ_1 norm in localization accuracy and robustness. BSCPO achieves high localization accuracy of 100% when the SNR of the noisy test signal and noisy dictionary exceeds -10 dB and -20 dB, respectively.

In particular, in Fig. 9, we take the wireless transmitter–receiver sensor-1 as an example for presenting the recovery or denoising performance of our proposed algorithm. The SNR of the noisy signal is -5 dB. From Fig. 9, there are some apparent amplitude offsets between the noisy signals and the original signals. Through the denoising process using our BSCPO algorithm, the signals are almost completely recovered and two recovery signals are very close to the corresponding noise-free original signals, respectively.

4.2.2 Comparison with baselines and state-of-the-art DFL methods

We now compare the proposed scheme with six other machine-learning methods except for the previously discussed algorithms with ℓ_0 norm and ℓ_1 norm. In particular, the deep CAE [8], SC-ISTA [10], and SC-OMP [30] are

TABLE 3
Ground-truth location distributions of two targets under six cases

Cases	Indexes of grids	Coordinate (Unit: feet)	
		Target 1 (x_1, y_1)	Target 2 (x_2, y_2)
1	Grid 25, Grid 26	(3, 15)	(6, 15)
2	Grid 25, Grid 27	(3, 15)	(9, 15)
3	Grid 25, Grid 28	(3, 15)	(12, 15)
4	Grid 25, Grid 29	(3, 15)	(15, 15)
5	Grid 25, Grid 30	(3, 15)	(18, 15)
6	Grid 23, Grid 25	(3, 15)	(15, 12)

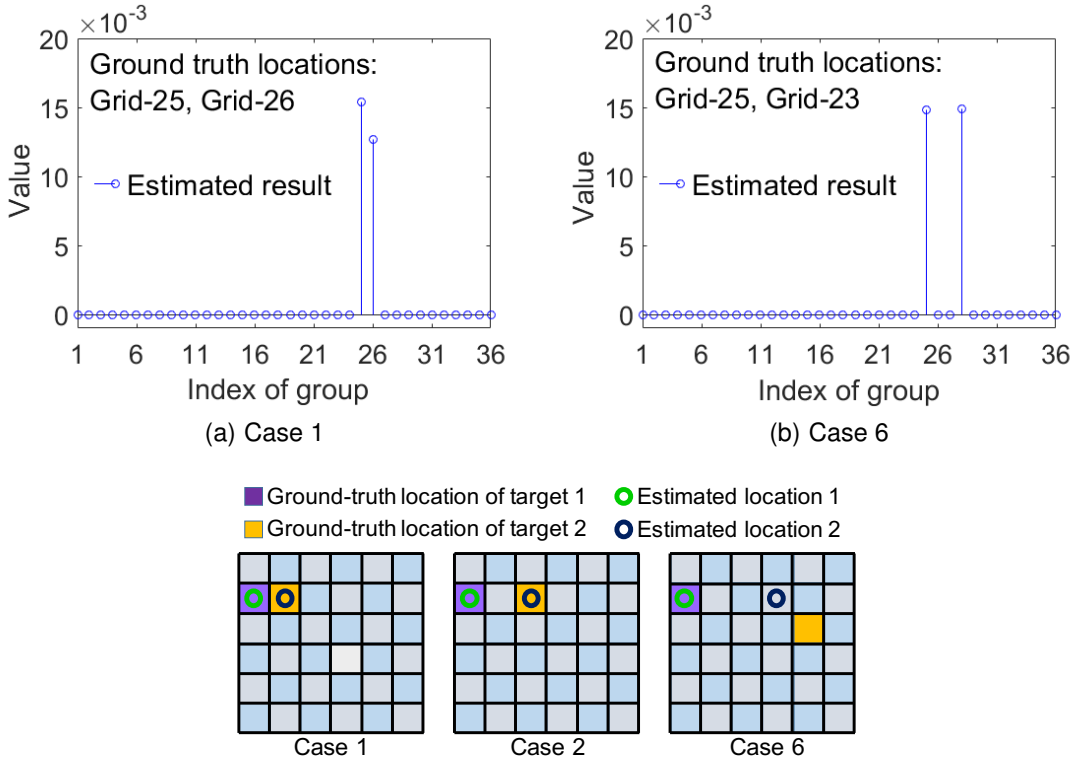

Fig. 10. Localization results of the proposed approach for multitarget experiments. (a) and (b) show location estimations with the block-sparse solution. The ground-truth location distributions of six cases are listed in Table 3.

existing state-of-the-art DFL algorithms that use the same dataset implemented in this paper. According to [9] and [10], the raw RSS signal without background elimination was used for SC-ISTA, SC-OMP, and KNN; therefore, the same conditions are adopted in the comparison experiments. Furthermore, we also perform experiments with the commonly used baseline classification methods, i.e., SVM [32] and the one-dimensional CNN (CNN-1D) [33]. For SVM, a one-vs-one strategy and radial basis function kernels are employed for the task of multiclass classification.

The comparison results are presented in Table 2. From Table 2, our BSCPO algorithm achieves the highest localization accuracy under severely noisy conditions of SNR = -10 (for dictionary) and SNR = -5 dB (for test signal). This indicates that BSCPO outperforms the other six machine-learning algorithms in terms of robustness and localization accuracy, which leads to a dependable DFL process.

4.2.3 Performance of the proposed approach for multitarget localization

We evaluate the localization performance of the proposed BSCPO with respect to locating multitargets where the case of locating two targets is taken as the example. The ground-truth locations of targets are distributed under six cases listed in Table 3.

The localization results on the raw test data are shown in Fig. 10. The targets can be accurately located from case 1 to case 5 (see Fig. 10 (a) and (c)). Here, the block-sparse solution of one correct-locating case, i.e., case 1, is taken as an example. For case 6, the target at grid 23 is incorrectly located at the diagonally adjacent grid 28 but is still approaching the correct location, while another target can

be located correctly at grid 25 (see Fig. 10 (b) and (c)). This indicates that the proposed BSCPO performs well in the task of multitarget localization.

***Discussion*:** In summary, the proposed block-sparse-coding scheme achieves a better localization performance and robustness for localizing a single target and multiple targets. Nonetheless, since the real data is directly transmitted among sensors, base station and servers, the privacy of legal objects may be leaked. Such a data privacy issue is very critical under a normal IoT architecture. However, the related studies as well as our current work does not consider the privacy preservation yet. Therefore, for the future work, we will further investigate the privacy issue faced by DFL system for various IoT applications. For privacy preservation, a promising method is federated machine learning. This is an emerging technology because it can update the model with transmitting the real data, which thereby protects the data privacy. In our future work, we will investigate how to update the variations of dictionary without leaking the real data. In addition, since the proposed block-sparse-coding method is robust to noisy signal and can be used for classification, it will be possibly extended for other applications, for example detecting some network attacks (e.g., the distributed denial-of-service attack).

5 Conclusion

In this paper, we present a block-sparse based ML approach to achieve an accurate and robust DFL process in IoT environment. In particular, $\ell_{2,1}$ norm is exploited to generate a group structure in the sparse solution. We then devise an optimization method with the proximal operator, which leads to the proposed ML algorithm, i.e. BSCPO. In addition, to prevent the network-related privacy from leaking to intruders, we add severe Gaussian noise in the original RSS signals to degrade the SNR. The dependability of the proposed scheme are then evaluated by comparing with multiple baselines and state-of-the-art machine-learning algorithms. Experimental results based on the real-world dataset show that BSCPO takes full use of the natural knowledge as group structure and is capable of target localization. To locate a single target, the proposed BSCPO achieves a high localization accuracy of 100% when the SNR of the noisy test signal and noisy dictionary exceeds -10 dB and -20 dB, respectively. For all six cases of multitarget localization, BSCPO can accurately locate the targets in a total of five cases and correctly recover the corresponding observation signals. Furthermore, BSCPO outperforms the other algorithms in terms of localization accuracy and robustness under heavily noisy conditions.

Appendix
Proof of Transformation with Respective to the Objective Function

The equation of original objective function (14) is as follows:

$$J(\mathbf{x}) = \frac{1}{2}\|\mathbf{y}-\mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_{2,1} \tag{14}$$

By incorporating the derivative of (14) to zero, we can obtain the minimizer $\mathbf{x}^*$ as follows,

$$\mathbf{x}^* = (\mathbf{D}^T\mathbf{D} + \lambda\frac{\partial(\|\mathbf{x}\|_{2,1})}{\partial\mathbf{x}})^{-1}\mathbf{D}^T\mathbf{y} \tag{15}$$

The modified objective function (16) is,

$$P(\mathbf{x}): \quad \underset{\mathbf{x}}{\operatorname{argmin}}\ \frac{\mu}{2}\|\mathbf{b}-\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_{2,1} \ + \ K \tag{16}$$

where K is regarded as a known constant that it does not depend on $\mathbf{x}$; μ is a scaling parameter that should be greater than the largest eigenvalue of $(\mathbf{D}^T\mathbf{D})$, for example, 1.01 times the largest eigenvalue; $\mathbf{b}$ is an intermediate variable with the same size of $\mathbf{x}$, given in the following derivation.

Here, we present the processes of modification and derivation from (14) to (16) in detail.

According to the analysis with respect to (15), $J(\mathbf{x})$ is not easily minimized because solving $(\mathbf{D}^T\mathbf{D})^{-1}$ may be computationally expensive or failed. This motivates us to construct a new function that avoids solving the $(\mathbf{D}^T\mathbf{D})^{-1}$ by the majorization-minimization (MM) approach [34], [35]. MM approach minimizes a difficult problem instead by a group of easier minimization problems (k=0,1,2...).

Based on the principle of MM, we should find a $H_k(\mathbf{x})$ so that at each iteration k, we can choose a $H_k(\mathbf{x})$ that coincides with $J(\mathbf{x})$ at $\mathbf{x}^{(k)}$ but otherwise upper-bounds $J(\mathbf{x})$, i.e., the following conditions

- (a) $H_k(\mathbf{x}) \geq J(\mathbf{x})$ for all $\mathbf{x}$
- (b) $H_m(\mathbf{x}^{(k)}) = J(\mathbf{x}^{(k)})$

Then, the procedure is designed as the following steps,

$$H_k(\mathbf{x}) = J(\mathbf{x}) + \text{Non-negative function of } \mathbf{x} \tag{17}$$

$$\begin{aligned} H_k(\mathbf{x}) = &\frac{1}{2}\|\mathbf{y}-\mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_{2,1} \\ &+ (\mathbf{x}-\mathbf{x}^{(k)})^T(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})(\mathbf{x}-\mathbf{x}^{(k)}) \end{aligned} \tag{18}$$

where μ must be selected to equal or greater than the maximal eigenvalue of $(\mathbf{D}^T\mathbf{D})$. Then, in the final term, $(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})$ will be a positive semi-definite matrix, which means that

$$\begin{aligned} &\mathbf{v}^T(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\mathbf{v} \geq 0, \ \forall\mathbf{v}, \text{ for} \\ &\mu \geq \text{maximal eigenvalue of } (\mathbf{D}^T\mathbf{D}) \end{aligned} \tag{19}$$

So we have a $H_k(\mathbf{x})$ as desired. Let us then check the derivative of the new function with respect to $\mathbf{x}$,

$$\frac{\partial H_k(\mathbf{x})}{\partial\mathbf{x}} = -\mathbf{D}^T\mathbf{y} - (\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\mathbf{x}^{(k)} + \mu\mathbf{x} + \lambda\frac{\partial(\|\mathbf{x}\|_{2,1})}{\partial\mathbf{x}} \tag{20}$$

$$\begin{aligned} &\frac{\partial H_k(\mathbf{x})}{\partial\mathbf{x}} = 0 \quad \Rightarrow \\ &\mathbf{x}^{**} = \mathbf{x}^{(k)} + \frac{1}{\mu}\mathbf{D}^T(\mathbf{y}-\mathbf{D}\mathbf{x}^{(k)}) + \frac{\lambda}{\mu}\frac{\partial(\|\mathbf{x}\|_{2,1})}{\partial\mathbf{x}} \end{aligned} \tag{21}$$

Compared (21) with (15), we can see that (21) avoids solving $(\mathbf{D}^T\mathbf{D})^{-1}$, which will significantly save the computational resource.

Finally, through the following derivation process, we can

obtain the modified objective function of (16).

$$\begin{aligned} H_k(\mathbf{x}) =& \frac{1}{2}\mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{D}\mathbf{x} + \frac{1}{2}\mathbf{x}^T\mathbf{D}^T\mathbf{D}\mathbf{x}^2 + \\ & \frac{1}{2}(\mathbf{x}-\mathbf{x}^{(k)})^T(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})(\mathbf{x}-\mathbf{x}^{(k)}) + \lambda\|\mathbf{x}\|_{2,1} \\ =& \frac{1}{2}\mathbf{y}^T\mathbf{y} + \frac{1}{2}(\mathbf{x}^{(k)})^T(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\mathbf{x}^{(k)} - \\ & \big(\mathbf{y}^T\mathbf{D} + (\mathbf{x}^{(k)})^T(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\big)\mathbf{x} + \frac{1}{2}\mu\mathbf{x}^T\mathbf{x} + \lambda\|\mathbf{x}\|_{2,1} \end{aligned} \tag{22}$$

Here, let

$$\begin{aligned} \mathbf{b} &= \frac{1}{\mu}\left[(\mathbf{D}^T\mathbf{y} + (\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\mathbf{x}^{(k)}\right] \\ &= \mathbf{x}^{(k)} + \frac{1}{\mu}\mathbf{D}^T(\mathbf{y} - \mathbf{D}\mathbf{x}^{(\mathbf{k})}) \end{aligned} \tag{23}$$

We can obtain the quadratic expression below

$$\begin{aligned} H_k(\mathbf{x}) &= \frac{\mu}{2}(-2\mathbf{b}^T\mathbf{x} + \mathbf{x}^T\mathbf{x}) \ + \ K_1 + \lambda\|\mathbf{x}\|_{2,1} \\ &= \frac{\mu}{2}\|\mathbf{b}-\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_{2,1} \ + \ K \end{aligned} \tag{24}$$

where K_1 consists of the first two terms of (22) which do not depend on $\mathbf{x}$, and K is a constant with respect to $\mathbf{x}$, given as follows

$$\begin{aligned} K =& \frac{1}{2}\mathbf{y}^T\mathbf{y} + \frac{1}{2}\mathbf{y}^T\mathbf{D}\mathbf{D}^T\mathbf{y} + \mathbf{y}^T\mathbf{D}(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\mathbf{x}^{(k)} - \\ & \frac{1}{2}(\mathbf{x}^{(k)})^T\mathbf{D}^T\mathbf{D}(\mu\mathbf{I}-\mathbf{D}^T\mathbf{D})\mathbf{x}^{(k)} \end{aligned} \tag{25}$$

In a consequence, The problem $P(\mathbf{x})$ in (16) equals to minimize $H_k(\mathbf{x})$ of (24).

REFERENCES

[1] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Scalable video coding and transport over broadband wireless networks," *Proceedings of the IEEE*, vol. 89, no. 1, pp. 6–20, 2001.

[2] A. Booranawong, N. Jindapetch, and H. Saito, "Adaptive filtering methods for rssi signals in a device-free human detection and tracking system," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2998–3009, 2019.

[3] Y. Zhao, W.-C. Wong, T. Feng, and H. K. Garg, "Calibration-free indoor positioning using crowdsourced data and multidimensional scaling," *IEEE Transactions on Wireless Communications*, 2019.

[4] D. Konings, F. Alam, F. Noble, and E. M. Lai, "Improved distance metrics for histogram-based device-free localization," *IEEE Sensors Journal*, vol. 19, no. 19, pp. 8940–8950, 2019.

[5] W. Meng, "Intrusion detection in the era of iot: Building trust via traffic filtering and sampling," *Computer*, vol. 51, no. 7, pp. 36–43, 2018.

[6] P. Merriaux, Y. Dupuis, R. Boutteau, P. Vasseur, and X. Savatier, "Robust robot localization in a complex oil and gas industrial environment," *Journal of Field Robotics*, vol. 35, no. 2, pp. 213–230, 2018.

[7] M. Akter, M. O. Rahman, M. N. Islam, M. M. Hassan, A. Alsanad, and A. K. Sangaiah, "Energy-efficient tracking and localization of objects in wireless sensor networks," *IEEE Access*, vol. 6, pp. 17 165–17 177, 2018.

[8] L. Zhao, H. Huang, X. Li, S. Ding, H. Zhao, and Z. Han, "An accurate and robust approach of device-free localization with convolutional autoencoder," *IEEE Internet of Things Journal*, 2019.

[9] D. Wang, X. Guo, and Y. Zou, "Accurate and robust device-free localization approach via sparse representation in presence of noise and outliers," in *IEEE International Conference on Digital Signal Processing (DSP)*. IEEE, 2016, pp. 199–203.

[10] H. Huang, H. Zhao, X. Li, S. Ding, L. Zhao, and Z. Li, "An accurate and efficient device-free localization approach based on sparse coding in subspace," *IEEE Access*, vol. 6, no. 1, pp. 61 782–61 799, 2018.

[11] P. Verma and S. K. Sood, "Fog assisted-iot enabled patient health monitoring in smart homes," *IEEE Internet of Things Journal*, 2018.

[12] I. Bisio, A. Delfino, F. Lavagetto, and A. Sciarrone, "Enabling iot for in-home rehabilitation: Accelerometer signals classification methods for activity and movement recognition," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 135–146, 2017.

[13] H. Huang, L. Zhao, H. Huang, and S. Guo, "Machine fault detection for intelligent self-driving networks," *IEEE Communications Magazine*, vol. 58, no. 1, pp. 40–46, 2020.

[14] E. J. Candes *et al.*, "The restricted isometry property and its implications for compressed sensing," *Comptes rendus mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[15] W. Ke, M. Chen, H. Zuo, Y. Wang, and T. Wang, "Device-free localization based on spatial sparsity with basis error self-calibration," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–5.

[16] J. Wang, Q. Gao, H. Wang, P. Cheng, and K. Xin, "Device-free localization with multidimensional wireless link information," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 356–366, 2015.

[17] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, and Q. Liang, "Sparse target counting and localization in sensor networks based on compressive sensing," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 2255–2263.

[18] L. Zhao, C. Su, Z. Dai, H. Huang, S. Ding, X. Huang, and Z. Han, "Indoor device-free passive localization with dcnn for location-based services," *The Journal of Supercomputing*, pp. 1–18, 2019.

[19] H. Yiğitler, R. Jäntti, O. Kaltiokallio, and N. Patwari, "Detector based radio tomographic imaging," *IEEE Transactions on Mobile Computing*, vol. 17, no. 1, pp. 58–71, 2017.

[20] A. Booranawong, K. Sengchuai, and N. Jindapetch, "Implementation and test of an rssi-based indoor target localization system: Human movement effects on the accuracy," *Measurement*, vol. 133, pp. 370–382, 2019.

[21] Z. Han, C. Su, S. Ding, H. Huang, and L. Zhao, "Device-free localization via sparse coding with log-regularizer," in *2019 IEEE 10th International Conference on Awareness Science and Technology (iCAST)*. IEEE, 2019, pp. 1–6.

[22] T. F. Sanam and H. Godrich, "A multi-view discriminant learning approach for indoor localization using amplitude and phase features of csi," *IEEE Access*, vol. 8, pp. 59 947–59 959, 2020.

[23] C. Feng, W. S. A. Au, S. Valaee, and Z. Tan, "Received-signal-strength-based indoor positioning using compressive sensing," *IEEE Transactions on Mobile Computing*, vol. 11, no. 12, pp. 1983–1993, 2012.

[24] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[25] C. A. Micchelli, J. M. Morales, and M. Pontil, "Regularizers for structured sparsity," *Advances in Computational Mathematics*, vol. 38, no. 3, pp. 455–489, 2013.

[26] X. Zhu, X. Li, S. Zhang, C. Ju, and X. Wu, "Robust joint graph sparse coding for unsupervised spectral feature selection," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 6, pp. 1263–1275, 2017.

[27] H. Wang, C.-S. Leung, H. C. So, J. Liang, R. Feng, and Z. Han, "Robust mimo radar target localization based on lagrange programming neural network," *arXiv preprint arXiv:1805.12300*, 2018.

[28] B. Tan, Y. Li, S. Ding, I. Paik, and A. Kanemura, "Dc programming for solving a sparse modeling problem of video key frame extraction," *Digital Signal Processing*, vol. 83, pp. 214–222, 2018.

[29] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 621–632, 2010.

[30] T. Liu, X. Luo, and Z. Liang, "Enhanced sparse representation-based device-free localization with radio tomography networks," *Journal of Sensor and Actuator Networks*, vol. 7, no. 1, p. 7, 2018.

[31] J. Wang, X. Zhang, Q. Gao, H. Yue, and H. Wang, "Device-free wireless localization and activity recognition: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, pp. 6258–6267, 2017.

[32] Z. Wen, J. Shi, Q. Li, B. He, and J. Chen, "ThunderSVM: A fast SVM library on GPUs and CPUs," *Journal of Machine Learning Research*, vol. 19, pp. 1–5, 2018.

[33] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ecg classification by 1-d convolutional neural networks," *IEEE*

Transactions on Biomedical Engineering, vol. 63, no. 3, pp. 664–675, 2016.

[34] M. A. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak, "Majorization–minimization algorithms for wavelet-based image restoration," *IEEE Transactions on Image processing*, vol. 16, no. 12, pp. 2980–2991, 2007.

[35] I. W. Selesnick, "Sparse signal restoration," *Connexions*, pp. 1–13, 2009.