Compressed-Domain ECG-based Biometric User Identification Using Task-Driven Dictionary Learning

TING-WEI SUN, DANISH ALI, and AN-YEU (ANDY) WU, Graduate Institute of Electrical Engineering, National Taiwan University

In recent years, user identification has become crucial for authorized machine access. Electrocardiography (ECG) is a new and rising biometrics signature. Rather than traditional biological traits, ECG cannot be easily imitated. In the long-term monitoring system, the wireless wearable ECG biomedical sensor nodes are resource-limited. Recently, compressive sensing (CS) technology is extensively applied to reduce the power of data transmission and acquisition. The prior CS-based reconstruction process aims at improving energy efficiency with different schemes, and they focus on the performance of reconstruction only. Therefore, we present a sparse coding-based classifier, trained by task-driven dictionary learning (TDDL), to realize low-complexity user identification in compressed-domain directly. TDDL is one of the dictionary learning and designed for classification tasks. It co-optimizes the dictionary and classifier weighting simultaneously, which gives better accuracy. In this article, we are proposing a TDDL-based compression learning algorithm for ECG biometric user identification as this directly identifies user identity (ID) without undergoing reconstruction process and conventional classifier. It can extract necessary information from the compressed-ECG signal directly to save the system power and computational complexity. The algorithm has 2%–10% accuracy improvements compared with state-of-the-art algorithms and maintains low complexity at the same time. Our proposed TDDL-CL will be the better choice in the long-term wearable ECG biometric devices.


Additional Key Words and Phrases: ECG biometric, compressive sensing, task-driven dictionary learning

ACM Reference format:
https://doi.org/10.1145/3461701

1 INTRODUCTION
Nowadays, the role of user identification has become more critical in authorized machine access. Therefore, we need a competent and reliable biometric system [1]. Electrocardiography (ECG), has identified itself as a novel biometric signature [2]. The ECG biometric user identification has several advantages, such as the ECG
signal is persistent and hard to be imitated. Additionally, in certain spots like the cleanroom or pandemic-control negative weight seclusion rooms, it is difficult to remove the veil or gloves to pass the security check. In the cases, it is progressively suitable to apply a remote-transmitted ECG signal to check if the individual has allowed authorization or not (Figure 1).

There are several studies on ECG-based biometrics with neural network (NN) algorithms [3, 4], and [5], as shown in Figure 2(a). Although the NN has advantages such as higher accuracy and robustness in general, there are two significant disadvantages of NN-based ECG biometrics; (i) the front end and transmission overhead will be worse, and (ii) the complexity of the NN network is extremely high. In the long-term monitoring system, the wireless wearable ECG biomedical sensor nodes are asset restricted [6]. These shortcomings cause the existing NN system to become almost unfit for direct application for long-term wearable ECG biometric devices.
Fortunately, the development of **compressive sensing** (CS) innovation reduces the ECG sensor’s power issue. CS is a technique that joins both sampling and compression to decrease the intensity of information obtaining and transmission \[7, 8\]. In comparison to the traditional technology, the CS-based ECG sensor can expand the battery lifetime by around 37\% \[9\]. Hence, the use of CS brings about an actual existence time expansion of the sensor node. In addition, CS could likewise be a work in encryption strategy \[10\]. CS can uncover implanted security and encryption highlights with unnecessary overhead \[11\]. In this way, CS-based ECG biometric user identification is progressively reasonable for practical considerations.

Currently, state-of-the-art CS-based ECG signal classification frameworks are **reconstruction learning** (RL) \[12, 13\], and **compression learning** (CL) \[14, 15\]. Before identifying the RL algorithm, it needs to reconstruct the ECG signal from the compressed ECG signal. Previous research called attention to that the processing time for CS reconstruction nearly the whole processing time. The RL algorithm is shown in Figure 2(b). The CL algorithm bypasses the reconstruction process and analyzes directly in the compressed domain, as shown in Figure 2(c).

After CS compression, although the information of identification is safeguarded, the representation will corrupt, prompting higher model complexity in the compressed domain \[16\].

There are many algorithms to solve linear inverse problems originating in image/signal processing. The CS-based reconstruction task has widely explored many sparse coding algorithms. These prior works aim at improving energy efficiency with different schemes. Nevertheless, they focus on the performance of reconstruction only. Therefore, we present a sparse coding-based classifier, trained by **task-driven dictionary learning** (TDDL) \[17\], to realize low-complexity CS-based user identification. **Dictionary learning** (DL) can be viewed as a transform layer to attain **sparse coefficients** (SC) having predictive information. The classifier based on SC can be of low complexity and high accuracy \[18\]. TDDL is one of the DL and is designed for classification tasks. The TDDL algorithm’s goal is to co-optimize the dictionary and classifier weighting simultaneously with the help of backpropagation and sparse coding, which gives better accuracy, as shown in Figure 3.

In this article, we propose a **TDDL-based compression learning** (TDDL-CL) algorithm. Our goal is to improve the accuracy and the generalization in the user identification scenario, with overall low complexity and power, by using TDDL and a **fast iterative shrinkage-thresholding algorithm** (FISTA) \[19\]. Figure 2(d) shows the inference block diagram of our algorithm. Although our TDDL-CL algorithm cannot achieve high accuracy compared with the NN-based algorithm, but precision is not required for the discriminative task as
long as the sparse coding algorithm catches its features. The TDDL-CL still has a certain level of accuracy and keeps the overall low computational complexity while retaining all the CS technique advantages, which will be further verified in a later section. Therefore, TDDL-CL will be the better choice in the long-term wearable ECG biometric devices. Our fundamental contributions are summed up as follows:

- **We proposed a TDDL-CL algorithm directly to identify user ID without reconstruction process and conventional classifier:** We proposed a TDDL-CL algorithm to identify user ID in compressed-domain directly. Traditional CS-based ECG biometric user identification requires reconstructing the compressed ECG signal and an extra classifier like a support vector machine (SVM). Our proposed algorithm can identify user ID in compressed-domain without reconstruction process and additional classifier.

- **The proposed TDDL-CL algorithm can achieve high accuracy and maintain low-complexity at the same time will be the better choice in the long-term wearable ECG biometric devices:** From the simulation and the analysis, compared with RL and CL, our algorithm has 2% and 10% higher accuracy and also reduced the memory overhead by 68.5% and 61.5%, respectively. In comparison to CA-CA [20], accuracy is improved by 6%.

The remainder of this article is sorted out as follows: Section 2 surveys earlier works, Section 3 subtleties our proposed algorithm. Section 4 demonstrates the simulation results. In Section 5, we break down the computational time and memory overhead. We finish up our work in Section 6.

## 2 RELATED WORKS

### 2.1 Compressive Sensing (CS)

CS [7, 8] is an epoch-making technique. Conventional front-end signal processing splits into two stages, data acquisition and compressing, and CS can combine these two stages into one. If the signal is sparse in a specific domain, the CS can guarantee reconstruction with few measurements without satisfying the Nyquist theorem. We can express the compressed measurement \( \hat{x} \) by CS as

\[
\hat{x} = \Phi x, \tag{1}
\]

where \( \hat{x} \in \mathbb{R}^M, \Phi \in \mathbb{R}^{M \times N} \) is the random sensing matrix, and \( x \in \mathbb{R}^N \) is the original signal. The measurement matrix needs to satisfy the constraint that the column in \( \Phi \) must be incoherent. The dimension of the measurement \( \hat{x} \) is much smaller than original signal \( x \), the system of equations is undetermined, having more unknowns than equations, system (1) either has no solutions or infinitely many solutions and so of course, this is not a properly stated problem in linear algebra [21, 22], this is what well-known CS process.

The purpose of the CS reconstruction algorithm is to recover the signal from the compressed measurement \( \hat{x} \). Most signals are not sparse in the time domain but sparse in other domain even the domain only have mathematical meaning. With proper projection matrix \( \Psi \), reconstruction can divide into two procedures. First, we solve the SC by (2)

\[
\min_{\alpha} \|s\|_0 = \sum_{i=1}^{N} |s_i| \quad \text{subject to} \quad \hat{x} = \Phi x = \Phi \cdot \Psi s = \Theta s, \tag{2}
\]

where \( s \in \mathbb{R}^N, \Psi \in \mathbb{R}^{N \times N} \) and \( s \) is the SC based on \( \Psi \), and \( \|s\|_0 \) is the \( l_0 \) norm of \( s \). The reconstructed signal \( x' \) is calculated by

\[
x' = \Theta s. \tag{3}
\]

The CS reconstruction is an underdetermined process [21, 22], the state-of-the-art algorithm, like basis pursuit (BP) [23], can find an optimal solution of Equation (2). Nevertheless, the computational complexity is unacceptably high.
One method for reconstruction is using the following $l_1$-norm optimization

$$
\alpha (x, D, \lambda) = \text{argmin}_{\alpha \in \mathbb{R}^d} \frac{1}{2} \|x + Da\|^2_2 + \lambda \|\alpha\|_1,
$$

via a greedy-type algorithm, such as **orthogonal matching pursuit (OMP)** [24], to accelerate the process of reconstruction.

2.2 Task Driven Dictionary Learning (TDDL)

DL is a part of signal processing that targets finding a casing in which some preparation information concedes a sparse representation. TDDL [17] is one of the DL and is designed for classification tasks. The goal of TDDL is to formulate the DL in a supervised way for the classification task. It optimizes the dictionary and classifier simultaneously compared to conventional DL, which provides better accuracy. Classification task performance is frequently related to data representation. TDDL is suitable for semi-supervised learning, can exploit unlabeled data in the sparse domain, and prompts cutting edge results for different signal processing tasks. TDDL is useful in the classification task, especially in large-scale data, it allows the use of millions of training samples. It can successfully exploit unlabeled data when only a few labeled samples are available. Compared with NN, it is much simpler to use since it does not require muddled heuristic systems to choose the parameters.

Considering the $m$ sample data $X \in \mathbb{R}^{N \times m}$ and their label $y \in \mathbb{R}^m$, $x \in \mathbb{R}^N$. TDDL intends to learn the dictionary $D_T \in \mathbb{R}^{N \times c}$, and low-complexity classification model $W_T \in \mathbb{R}^{c \times d}$, where $d$ is the number of the class, which can be utilized in edge computing. If the signal has sparsity, **predictive dictionary learning (PDL)** can serve as an energy-efficient classifier but has better classification accuracy than the conventional **machine learning (ML)** algorithm, such as SVM or NN. PDL consists of two layers, where the first layer serves as a feature extraction layer to transform target signals to SCs, and the second layer is a classifier based on the SCs from the previous layer. PDL optimizes the following empirical supervised loss, defined as Equation (4)

$$
(D_T, W_T) = \text{argmin}_{D \in \mathcal{D}_c, W \in \mathcal{W}} f_c (D_T, W_T, \lambda),
$$

and $f_c$ defined as Equation (5)

$$
f_c (D_T, W_T, \lambda) = \frac{1}{m} \sum_{i=1}^m l_c (y_i, W_T, \alpha_i) + \frac{\nu}{2} \|W_T\|_F^2,
$$

where $\mathcal{D}_c \triangleq \{D_T \in \mathbb{R}^{N \times c} \text{ s.t. } \|d_j\|_2 \leq 1, \forall j = 1, 2, \ldots, c\}$, $\mathcal{W} \triangleq \mathbb{R}^c$, $\lambda$ is the sparsity penalty used in classification and $\alpha_i = \alpha (x_i, D, \lambda)$ defined in Equation (4) is the sparse coding based on $D_T$ and $x$, $\nu$ is the regularization coefficient, $\|\cdot\|_F^2$ is the Frobenius norm, and $l_c$ is defined as

$$
l_c (y_i, W_T, \alpha^*) = \frac{1}{2} \|W_T^* \alpha^* - y_i\|_F^2,
$$

$l_c$ can be chosen as any supervised loss, and here adopts the linear loss with one-hot encoding and $\alpha^*$ is one of the solutions within $\alpha_i$. A naive way to optimize Equation (4) is to optimize $D_T$ and $W_T$ sequentially. Although this method enables an easy training process, the accuracy cannot be optimal. In the procedure of TDDL, the first step is to find the predictive sparse via sparse coding algorithms based on the dictionary from the previous iteration. This step can also be viewed as a feature extraction layer to explore the essence of the signals. Finally, we can update the classifier $W_T$ and the dictionary $D_T$ with the meta-information via gradient descent methods. After several iterations, we obtain the sparse coding-based classifier $W_T$. Therefore, it provides a more optimal solution compared to that by optimizing $D_T$ and $W_T$ sequentially. Although TDDL cannot achieve high accuracy compared with the NN-based algorithm. The TDDL algorithm still has a certain level of accuracy and keeps the overall low computational complexity while retaining all the CS technique advantages. It can be a very suitable choice in the long-term wearable ECG biometric devices.
2.3 Reconstructed Learning (RL) and Compressed Learning (CL)

The most intuitive algorithm for the CS-based ECG signal user identification is RL [12, 13]. RL first uses some portion of the ECG raw signal as training data to train ML models for user identification [2]. During inference, the back-end receiver reconstructs the compressed ECG signal and identifies the user ID by the pre-trained ML model. The CS block defines the sparse dictionary and accomplishes the reconstruction function. Recently, the NN algorithm has also been widely adopted for ECG signal classification [25, 26]. Nevertheless, the processing time for CS reconstruction is nearly the whole processing time. Furthermore, the rebuilding of the compressed ECG signal likewise devastates the privacy protection obtained from the CS algorithm. Also, without alignment, the differences between heartbeats will be significant. To improve accuracy and reduce the learning resources, the ECG signal should be aligned [27].

The CL [14, 15] algorithm bypasses the reconstruction process and directly performs the inference in the compressed domain, which is beneficial for both the CS and ML point of view. From the CS viewpoint, it takes out the full expense of recouping irrelevant data; as it were, CL resembles a filter and makes it conceivable just to recover the desired signals. From the ML viewpoint, CS can be regarded as a productive universal sparse dimensionality decrease between the data and measurement domain.

Random projections (RP) have been used in real-time applications to reduce the computation latency [28]. With RP, the separations between the points are approximately preserved as mapping down onto a lower-dimensional space. Therefore, in [29], they presented that if the classifier trained with the data directly comes from CS-domain, the performance is almost the same as the best possible classifier trained with the original data domain. Nevertheless, after CS compression, even though the information of identification is safeguarded, the representation will corrupt, prompting higher model complexity in the compressed domain [16]. Therefore, higher learning resources are needed to model the compressed ECG signal. Also, after CS sampling, the time-domain characteristics will no longer exist. CL cannot directly align the ECG signal in the compressed domain, which results in a drop in accuracy than the RL algorithm.

2.4 Compressed Alignment-Aided Compressive Analysis (CA-CA)

In [20], the authors proposed a compressed alignment-aided compressive analysis (CA-CA) algorithm by using the principal component analysis (PCA) for user identification with an ECG signal. In order to improve accuracy and reduce the learning resources, the algorithm needs to align the ECG signal. In general, ECG alignment algorithms are developed in the timedomain. Therefore, the authors also proposed a compressed alignment algorithm to remove the domain restriction and align the ECG signal in the compressed domain directly, which can further enhance the accuracy of identifications and reduce the total training time. With CA-CA, it can extract necessary information from compressed ECG signal without the reconstruction process and saving the system power and computational complexity.

During the training phase, the CA-CA algorithm first finds the reference point from the compressed ECG signal. They further generated a circular shift matrix based on the same reference point. After that, the compressed ECG signal was transferred into representation vectors \( s_{a,i} \) on the same PCA basis and circular shift matrix, with pseudo-inverse matrix method. In the end, they train the SVM models of each user by using the representation vectors \( s_{a,i} \). During inference, the algorithm still needs to calculate the circular shift matrix and correspond eigenspace compressed test ECG signal \( s_t \) by pseudo-inverse matrix method. Lastly, the \( s_t \) is passed to the pre-train SVM model to classify the user ID. In comparison to CL, CA-CA enhances accuracy and saves the total training time. Nevertheless, there are several disadvantages to the CA-CA algorithm. First of all, accuracy is still lower than the RL, and CA-CA uses SVM as a classifier. Therefore, when the number of user IDs increases, the computational complexity will increase, and the accuracy will decrease. Lastly, without DL and sparse coding, CA-CA cannot achieve actual data encryption. On the other hand, the TDDL-CL algorithm does not require a reconstruction process and an additional classifier. It is observed in Table 1, the proposed algorithm has the lowest complexity, which serves as an efficient classifier. Therefore, the proposed algorithm only
Table 1. Comparison of the Memory and Accuracy

<table>
<thead>
<tr>
<th>Framework</th>
<th>Classifier</th>
<th>Memory</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>RL</td>
<td>SVM</td>
<td>&gt;1M</td>
<td>92.4%</td>
</tr>
<tr>
<td>CL</td>
<td>SVM</td>
<td>&gt;1M</td>
<td>84.1%</td>
</tr>
<tr>
<td>CA-CA</td>
<td>SVM</td>
<td>&lt;0.5M</td>
<td>88.1%</td>
</tr>
<tr>
<td>Proposed TDDL-CL</td>
<td>None</td>
<td>&lt;0.05M</td>
<td>94.3%</td>
</tr>
</tbody>
</table>

has a low overhead, making it the most appropriate classifier for compressed inference. With high sparsity, the classifier can be designed with low complexity [30]. Therefore, our proposed TDDL-CL algorithm has improved based on accuracy, classifier usage, and computational complexity.

In Table 1, we compared the memory requirement and accuracy of different algorithms. It shows our proposed algorithm does not need extra classifiers and achieves the best accuracy.

2.5 Benefits Comparisons between TDDL and Existing Algorithm

TDDL shows up as a fitting classifier to deliver low complexity and high generalization [31]. To determine a limited labeled dataset challenge, the classifier ought to be of extraordinary speculation. Besides, the tight battery limitation of biometric devices and the low response time prerequisite of real-time ECG monitoring demand, the classifier ought to be of low unpredictability. For these constraints, TDDL can be presented as a powerful model. TDDL can make due in restricted labeled dataset since sparse component extraction layer proposes high speculation [32]. Likewise, TDDL has low complexity and few parameters via co-optimizing two-layer structure of an inadequate component extraction and a basic classifier.

3 PROPOSED TDDL-BASED COMPRESSION LEARNING (TDDL-CL) ALGORITHM

The detailed step of off-line training and on-line inference virtual code is summarized in Algorithms 1 and 2. The block diagram is shown in Figure 4.

3.1 Off-line Dictionary Learning Algorithm

The purpose of the off-line training algorithm is to use the raw ECG signal $X_{ECG}$ from different users to train $D_{ID}$ and $W_{ID}$ in the compressed domain, where $X_{ECG} \in \mathbb{R}^{N \times m}$, $D_{ID} \in \mathbb{R}^{M \times c}$, $c$ is the number of atoms in the dictionary, $W_{ID} \in \mathbb{R}^{c \times d}$, $d$ is the number of the user ID. In order to increase accuracy and training efficiency; here, we also use the alignment algorithms.

Initially, our proposed method compresses the input signal by the CS algorithm. Then, to improve the overall accuracy, in this work, we also applied the compressed-domain ECG alignment algorithm [20]. We found the coordinate of the reference point $p$ from the compressed ECG signal and generate a circular shift matrix $R_p$ based on the reference point $p$. After that, we apply the alignment algorithms to get the compressed-domain alignment ECG signal $\hat{x}_a$. Finally, we applied TDDL algorithm with $\hat{x}_a$ to train the dictionary $D_{ID}$ and weighting $W_{ID}$.

The choice of parameters is highly correlated with the final performance. So, we detailed how to choose some key parameters. For the value of $\lambda$, it has been mentioned in the previous research [33] that if the signal is close to zero mean and unit $l_2$-norm, $\lambda$ value around 0.15 will have good initial performance. In this work, after repeated experiments, we chose a value of 0.12. For the value of $\nu$, when there are a lot of training data, which like image or speech, this term becomes unnecessary, and the value can be set to a small value, when there are not many training points, which like the E.C.G. signal, this parameter is set up by cross-validation. In this work, the value is $10^{-4}$, also by the result of the repeated experiment. For the value of $\rho$, here, we choose a very classic formula $\rho = \rho_0/t$, which has been presented in [34]. We select values of $\rho_0$ by observing the lowest error on a small validation set. The other parameters in Algorithms 1 and 2 are self-learned.
3.2 On-line User Identification Algorithm

The reconstruction of compressed signals usually takes much energy, which is a critical issue for edge devices. Therefore, in user identification, if we can identify the ECG signal when they are highly compressed, we do not need to reconstruct the ECG signal. In other words, the primary purpose of the on-line user identification algorithm is to distinguish differences under the compressed domain. Given $\mathbf{D}_{ID}$, we obtain from the off-line algorithm, and test compressed-alignment ECG signal $\hat{\mathbf{x}}_t$. We can obtain

$$\alpha_{ID} = \arg\min_{\alpha \in \mathbb{R}^c} \frac{1}{2} \| \hat{\mathbf{x}}_t - \mathbf{D}_{ID} \alpha_{ID} \|^2_2 + \lambda_{ID} \| \alpha_{ID} \|_1.$$  

(8)

To solve the SC $\alpha_{ID}$ for compressed-domain biometric user identification efficiently, we adopt the FISTA [19] as our optimizer. Many other algorithms can be utilized for $\ell_1$-optimization. In our case, FISTA outperforms them in both accuracy and complexity, which will be further compared in Sections IV and V. FISTA is one of the classical gradient algorithms for solving linear inverse issues, with computational simplicity and a better rate of convergence. It works as follows, assume $\hat{\mathbf{x}}_t$ already aligned in the compressed domain. The gradient of the reconstruction loss is

$$\nabla_{\alpha} (\alpha_{ID}) = \nabla_{\alpha} \frac{1}{2} \| \hat{\mathbf{x}}_t - \mathbf{D}_{ID} \alpha_{ID} \|^2_2 = \mathbf{D}_{ID}^T \mathbf{D}_{ID} \alpha_{ID} - \mathbf{D}_{ID}^T \hat{\mathbf{x}}_t.$$  

(9)

It can be observed that $\mathbf{D}_{ID}^T \mathbf{D}_{ID}$ and $\mathbf{D}_{ID}^T \hat{\mathbf{x}}_t$ are fixed in each iteration of the sparse coding process, it can avoid computations on the same matrices and reduce the complexity. In the procedure of FISTA, it requires shrinking the value with a given threshold ($T$). The sparsity constraint is

$$\beta_{t+1} = S_{\lambda \eta} (\alpha_{ID}^t),$$  

(10)

where $\alpha_{ID}^t = \alpha_{ID} - \eta \nabla_{\alpha} f (\alpha_{ID})$ and $S_{\lambda \eta}$ updates each entry of $\alpha_{ID}^t$ by shrinkage operator.
\begin{algorithm}
\caption{Off-line Dictionary Learning}
\begin{algorithmic}
\Require N-dimensional raw ECG signal \( X = [x_1, x_2, \ldots, x_m] \), \( x \in \mathbb{R}^N \), the user ID label: \( y \in \mathbb{R}^m \), and sensing matrix \( \Phi \in \mathbb{R}^{M \times N} \), the number of Iterations: \( T \); regularization: \( \nu, \rho \).
\For {i from 1 to m}
\State Compressed the input signal \( \hat{x}_i = \Phi \cdot x \)
\State Find the reference point \( p_i = \max_{i} (\Phi^T \cdot \hat{x}_i) \)
\State Compute the circular shift matrix \( R_{p_i} \) with \( p_i \)
\( R_{p_i} = \begin{cases} \text{Circ}(I_{N \times N}, N/2 - p_i) & , p_i \leq N/2 \\
\text{Circ}(I_{N \times N}, N/2 - p_i + N), & p_i > N/2 \end{cases} \)
\State Align the compressed ECG signal \( \hat{x}_{al} = \Phi \cdot R_{p_i} \cdot x \)
\EndFor
\For {t = 1 to T}
\State Compute active set \( \alpha^* \) using \( l_1 \)-norm optimization
\State \( \alpha^* (x_t, D_T) = \arg \min_{\alpha = \mathbb{R}^c} \frac{1}{2} \| x_t - D_T \alpha \|^2_2 + \lambda \| \alpha \|_1 \)
\State \( \alpha^*_\lambda \rightarrow \{ j \in \{1, \ldots, d \} : \alpha^*[j] \neq 0 \} \)
\State Compute the \( \beta^* \): set \( \beta^*_\lambda = 0 \), and \( \beta^* = (D_T^T D_T)^{-1} \nabla_{\alpha^*_\lambda} \log Z(y_t, \Phi, \omega, \alpha^*_\lambda) \)
\State Compute the learning rate \( \rho_t = \rho / t \)
\State Update the parameter by the projected gradient:
\State \( W_T = \Pi_{W_T} \left[ W_T - \rho_t (\nabla_{W_T} \log Z(y_t, W_T, \alpha^*_\lambda) + \nu W_T) \right] \)
\State \( D_T = \Pi_{D_T} \left[ D_T - \rho_t (-D_T \beta^*_T \alpha^*_\lambda + (x_t - D_T \beta^*_T \alpha^*_\lambda)) \right] \)
\EndFor
\Ensure \( D_{ID} \leftarrow D_T \) and \( W_{ID} \leftarrow W_T \).
\end{algorithmic}
\end{algorithm}

\[ S_{\lambda \eta} (\alpha_{ID}^+) j = \left( |\alpha_{ID}^+ j| - \lambda \eta \right) + \text{sgn} \left( (\alpha_{ID}^+) j \right) , j = 1, 2, 3, 4, \ldots, d, \]

\[ \xi^+ = \begin{cases} \xi, \xi > 0 \\ 0, \xi \leq 0 \end{cases}. \]

\[ \alpha_{ID+1} = \beta_{t+1} + \frac{\lambda_{t-1}}{k_{t+1}} (\beta_{t+1} - \beta_t). \]

Finally, the SCs are obtained by the linear combination of the current shrunk term and the previous shrunk term, which can be viewed as a momentum acceleration method.

In the end, we use \( W_{ID} \) and \( \alpha_{ID} \) to identify the user ID \( Y \).

\[ Y = W_{ID}^T \alpha_{ID}. \]

\section{Experimental Results}
\subsection{Simulation Settings}
In Table 2, we organized the simulation settings in this work, including the hardware and the data. There are two accessible open databases for evaluating ECG user identification performance, Physionet QT[35], and ECG-ID database [36]. In the ECG-ID dataset, the data distribution is disproportionate. The smallest data only has 20 seconds, but the most prominent data is 440 seconds of the ECG-ID database. Generally speaking, in the ECG-ID database, the mean value and the standard deviations are 68 seconds and 63 seconds, respectively.
Table 2. Simulation Settings of TDDL-CL

<table>
<thead>
<tr>
<th>Simulation Environment</th>
<th>MATLAB 2019b with Intel i7-7700/8G RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>PhysioNet QT-Database</td>
</tr>
<tr>
<td>Sampling Frequency</td>
<td>250 Hz</td>
</tr>
<tr>
<td>Input Dimension (N)</td>
<td>250</td>
</tr>
<tr>
<td>Number of User ID</td>
<td>22/80</td>
</tr>
<tr>
<td>Training Data Size</td>
<td>[100:100:600]</td>
</tr>
<tr>
<td>Inference Data Size</td>
<td>250</td>
</tr>
<tr>
<td>Sensing Matrix Type (Φ)</td>
<td>Random Gaussian</td>
</tr>
<tr>
<td>CR (M/N)</td>
<td>[0.3:0.1:0.9]</td>
</tr>
</tbody>
</table>

Table 3. Simulation Settings of CA-CA, RL, and CL

<table>
<thead>
<tr>
<th>Simulation Environment</th>
<th>MATLAB 2019b with Intel i7-7700/8G RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database</td>
<td>PhysioNet QT-Database</td>
</tr>
<tr>
<td>Sampling Frequency</td>
<td>250 Hz</td>
</tr>
<tr>
<td>Input Dimension (N)</td>
<td>250</td>
</tr>
<tr>
<td>Number of User ID</td>
<td>22/80</td>
</tr>
<tr>
<td>Training Data Size</td>
<td>[100:100:600]</td>
</tr>
<tr>
<td>Inference Data Size</td>
<td>250</td>
</tr>
<tr>
<td>Size of the Dictionary</td>
<td>Mx39</td>
</tr>
<tr>
<td>Sensing Matrix Type (Φ)</td>
<td>Random Gaussian</td>
</tr>
<tr>
<td>CR (M/N)</td>
<td>[0.3:0.1:0.9]</td>
</tr>
<tr>
<td>Machine Learning Model</td>
<td>RBF Kernel SVM</td>
</tr>
<tr>
<td>Cost (C) Search Range</td>
<td>1, 5, 10</td>
</tr>
<tr>
<td>Gamma (γ) Search Range</td>
<td>(10(^{-4}) to 1)</td>
</tr>
<tr>
<td>Cross-Validation</td>
<td>50-fold</td>
</tr>
</tbody>
</table>

Disproportion between evaluated data has a serious impact on the final performance. Therefore, we only choose the QT database as an evaluation database. The QT database contains both healthy and cardiological disorders-based ECG signals with a balanced distribution. In order to confirm the versatility of our algorithm, we first verify healthy subjects. The database we selected contains 22 ECG recordings obtained from 22 healthy persons, followed by more patients (a total of 80 persons) with cardiological disorders for user identification. We did not only confirm that our algorithm has a consistent accuracy rate when the number of users increases, but also, our algorithm has the generality for patients with cardiological disorders. In order to reduce the difference between ECG measuring instruments, we normalized the amplitude of each ECG signal into \(-1 \sim 1\) (mV).

4.2 Performance Evaluation

We compared our algorithm with RL, CL, and CA-CA. For RL, before training the SVM model, we reconstructed the ECG signal from the compressed ECG signal and aligned the reconstructed ECG signal. For CL, we trained the SVM model on compressed ECG signals directly. For CA-CA, we trained the SVM model on representation vectors. The SVM setting is as follows: RBF Kernel, the Gamma search range is set to [10\(^{-4}\) to 1], and the cross-validation is 50-fold. The detailed algorithm settings of CA-CA, RL, and CL are listed in Table 3. The sensing matrix used in CL is also listed in Table 3, \(M\) is the dimension of compressed data. The reconstruction algorithm we used in RL is the OMP algorithm [37]. We must specifically mention here, that no matter what reconstruction
algorithm, we were chosen; we will ensure the quality of the reconstructed ECG signal. According to [38], we will ensure that the percentage root mean square (RMS) difference (PRD) of the reconstructed ECG signal is lower than 5. In other words, it can ensure good signal restoration quality for classifying. In our proposed algorithm, we train the dictionary and weight, as shown in Algorithm 1. The training data size here means samples for each person.

Figure 5(a) shows the user identification accuracy under the different training data size when compression ratio (CR) = 0.5, and the number of the test subject is 22 healthy persons. It can be seen that

- Compare with CL, TDDL-CL has 9% higher accuracy. Nevertheless, there is an equal and small improvement in accuracy compared with RL.
- Compare with CL, TDDL-CL can maintain around 90% accuracy while reducing training data by 67%.
- Compare with CA-CA, TDDL-CL has 2% higher accuracy.

Figure 5(b) shows the accuracy when adding cardiological disorders users, a total of 80 users. In this experiment, each individual’s training data still maintains a maximum of 600 samples. It can be observed that

Compare with CL and RL, TDDL-CL has 10% and 2% higher accuracy, respectively.

Compare with CA-CA, TDDL-CL has a 6% higher accuracy.

When the number of users increases, the accuracy decrement of sparse encoding is smaller than the SVM-based algorithm.

Here, all the parameters we have obtained are fine-tuned for the current database. There will likely be overfitting. Since our main application scenario is user identification, not a large number of generalized classifiers, such as image recognition or face recognition. Overfitting will not be an issue in user identification applications. Instead, we hope that user identification must have a good performance on the limited registered user data. Therefore, in some aspects, overfitting is also needed to help improve performance. ECG signals are not like faces or voices, and they are quite different from person to person. Broadly speaking, the consistency of the ECG signal is better than other biometric signals. Therefore, we only need to confirm that our algorithm can have good results in one of the ECG databases. We believe that in different scenarios where ECG signals are used for user identification, after adjusting parameters and training, good results can also be obtained.

To verify the generalization, we used another set of public ECG database to verify our algorithm again. The database we use here is the MIT-BIH Arrhythmia Database [39]. The MIT-BIH Arrhythmia Database contains 48 half-hour two-channel ECG recordings, a total of 48 people is included. We also normalized the amplitude.

---

**Fig. 5.** Accuracy vs number of training data with (a) test subject = 22. (b) test subject = 80.
of each ECG signal into –1 ∼1(mV). The experimental results are shown in the Figure 6. Because it has been confirmed that the performance of CL is not acceptable before, we did not add CL here for comparison. Because they are all the same type of ECG data. Therefore, the settings of this experiment are equivalent to the parameters in Table 2 and Section 3.1. From experiment results, it can be found that the accuracy of RL, CA-CA, and TDDL-CL is similar to the previous trend of using the ECG-ID database. The accuracy of TDDL-CL is still higher than the other two. It also further verified that our algorithm has good generalization in using ECG signals for user identification.

5 ANALYSIS OF COMPUTATIONAL TIME AND MEMORY OVERHEAD

In this section, the computational time has been analyzed for different algorithms, as shown in Figure 7. The number of training data is 600, and the CR is 0.9. The simulation settings are the same as in Table 2. It can be seen that the computational time of the proposed method is 73% lower than RL, which results in reducing computing resources. The computational time of the TDDL-CL is 30% lower than CL. On the other hand, the CA-CA only have pseudo-inverse matrix operation during inference. Therefore, the computational time of our TDDL-CL is 1.7 times higher than CA-CA. Furthermore, we verified whether similar results could be obtained under different CR.

We analyze the resource overhead of memory requirement in the on-line stage, as shown in Table 4. Where \( N \) is the dimension of input raw ECG data, \( M \) is the dimension of compressed data, \( d_R \) and \( d_F \) are the corresponding dictionary length of RL and FISTA, \( k_d \) is the number of user ID, and \( n_{SV} \) is the number of support vector in SVM.

In RL, when the CR is 0.5,

- Measurement matrix, \( M d_R = 125 \times 250 \equiv 31.25K \),
- Sparsity matrix, \( N d_R = 250 \times 250 \equiv 62.5K \).
- SVM model in the time domain \( n_{SV} N = 4920 \times 250 \equiv 1.23M \).

In CL, when the CR is 0.5,

- SVM model \( n_{SV} M = 8689 \times 125 \equiv 1.09M \), the \( n_{SV} \) increases because of learnability degradation where the ECG signal is unaligned.
Table 4. Memory Requirement of Different Algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Formula</th>
<th>Actual number</th>
</tr>
</thead>
<tbody>
<tr>
<td>RL</td>
<td>((M + N)d_R + n_{SV}N)</td>
<td>1.33 M</td>
</tr>
<tr>
<td>CL</td>
<td>(n_{SV}M)</td>
<td>1.09 M</td>
</tr>
<tr>
<td>CA-CA</td>
<td>((M + k_d)N + k_dM + n_{SV}k_d)</td>
<td>0.21 M</td>
</tr>
<tr>
<td>TDDL-CL</td>
<td>(d_Fk_d + d_fd_F)</td>
<td>0.012 M</td>
</tr>
</tbody>
</table>

In the CA-CA algorithm,

- Alignment resource, \((M + k_d)N = (125 + 39) \times 250 \approx 41\) K.
- Projection matrix, \(k_dM = 39 \times 125 \approx 4.875\) K.
- SVM model, \(n_{SV}k_d = 4095 \times 39 \approx 0.16\) M.

In the TDDL-CL algorithm,

- Classify operation, \(d_Fk_d = 100 \times 22 = 2.2\) K.
- FISTA, \(d_Fd_F = 100 \times 100 = 0.01\) M.

The total memory requirement of TDDL-CL (0.012 M) is only 0.9% of RL (1.33 M), 1.3% of CL (1.09 M), and 5.9% of CA-CA (0.21 M). This is a predictable result because TDDL-CL does not need an extra SVM classifier, which is the main bottleneck of the memory requirement. From the simulation and analysis, the proposed methodology has the following conclusions:

- Our proposed algorithm has 10%, 6%, and 2% accuracy improvements compared with CL, CA-CA, and RL, respectively.
- The computational time of TDDL-CL is higher than CA-CA but much smaller than RL and CL.
- The memory requirement of TDDL-CL is much lower than RL, CL, and CA-CA.
- Overall, the proposed method is far better in terms of accuracy and computational complexity as compared to RL and CL. In comparison with CA-CA, although the computing time is relatively high nevertheless, the accuracy of CA-CA under a large number of users is lower than the applicable range \(\leq 90\%\).

6 CONCLUSION

We introduce a sparse coding-based classifier, trained by TDDL, to realize low-complexity user identification in compressed-domain directly. In the long-term monitoring system, the wireless wearable biomedical sensor devices are known to be resource-limited with the CS algorithm resulting in a lifetime extension of the sensor devices. We proposed a TDDL-CL algorithm to identify user ID without a reconstruction process and an extra classifier. It can extract necessary information from the compressed-ECG signal directly to save the system power and computational complexity. The proposed method, compared with RL and CL, gives better performance in both the accuracy and the computational resources. Although in comparison to CA-CA, the computing resources are relatively high, our proposed work holds better accuracy for a large number of users.

REFERENCES


Received June 2020; revised March 2021; accepted April 2021