RESEARCH ARTICLE

Chunguang WANG, Jinjiang LIU, Jixiang SUN

Compression algorithm for electrocardiograms based on sparse decomposition

Abstract Sparse decomposition is a new theory in signal processing, with the advantage in that the base (dictionary) used in this theory is over-complete, and can reflect the nature of a signal. Thus, the sparse decomposition of signal can obtain sparse representation, which is very important in data compression. The algorithm of compression based on sparse decomposition is investigated. By training on and learning electrocardiogram (ECG) data in the MIT-BIH Arrhythmia Database, we constructed an over-complete dictionary of ECGs. Since the atoms in this dictionary are in accord with the character of ECGs, it is possible that an extensive ECG datum is reconstructed by a few nonzero coefficients and atoms. The proposed compression algorithm can adjust compression ratio according to practical request, and the distortion is low (when the compression ratio is 20:1, the standard error is 5.11%). The experiments prove the feasibility of the proposed compression algorithm.

Keywords sparse decomposition, orthogonal matching pursuit (OMP), K-SVD, electrocardiogram (ECG)

1 Introduction

With the gradually widespread application of all types of electrocardiographs in clinical medicine, increasingly more electrocardiogram (ECG) data need to be stored and transmitted. Under the premise that a reconstructed ECG can meet the need of clinical diagnosis, ECG data compression technology can reduce data to facilitate ECG data storage and transmission.

Chunguang WANG (✉), Jixiang SUN
College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China
E-mail: novelspring@163.com

Jinjiang LIU
Department of Computer Science, Nanyang Normal University, Nanyang 473061, China

According to the different airspace in ECG data compression, the compression of an ECG can be classified into time-domain compression and transform-domain compression. The former appears earlier and uses the time-domain character of ECG directly. Its compression ratio is lower, but the compression speed is higher and the distortion of the reconstructed ECG is low. Thus, its application is widespread in earlier periods. The latter changes an ECG into a transform domain such as frequency domain [1] or wavelet domain [2]. This compression, which is suitable for removing redundancy, encodes and compresses the coefficients of transformation. Since its compression ratio is higher and can be controlled as needed, it is currently the mainstream compression algorithm. In recent years, there have been many compression algorithms based on wavelets at home and abroad. In Ref. [3], one-dimensional ECG data are first segmented and aligned into a two-dimensional data array. This data array is transformed by a two-dimensional wavelet transform, and the wavelet coefficients are then coded and compressed by a modified coding algorithm. In Ref. [4] a novel ECG compression and de-noising algorithm based on the wavelet transform is introduced. These compression algorithms have obtained better results. However, because there is no consequent connection between elements of the wavelet base and ECG, and these elements cannot reflect the structural characteristics of ECG, the algorithm is likely to cause greater distortion.

The ECG compression algorithm based on sparse decomposition is a kind of transform-domain compression algorithm. By training on and learning the ECG data, we can construct an overcomplete dictionary in accordance with the characteristics of the ECG. Based on this dictionary, the ECG is decomposed into a sparse vector and atoms corresponding to nonzero values of the vector. Using this sparse vector and atoms, the original ECG can be reconstructed and the distortion is very low. The data amount of spare decomposition coefficients is significantly smaller than the original ECG; therefore, this compression algorithm can realize a higher compression ratio. Because the dictionary is constructed by training on and learning

the ECG data, the atoms in the dictionary comply more with the characteristics of ECGs than the elements of other transform-domain bases. Therefore, with the same compression ratio, the ECG compression algorithm has lower distortion than other transform-domain compression algorithms.

2　Sparse decomposition, orthogonal matching pursuit (OMP) algorithm and K-SVD

2.1　Sparse decomposition

In 1993, Mallat and his partner proposed a new theory of using an overcomplete and redundant dictionary to sparsely decompose signal [5]. The new theory differs from the conventional decomposition theories in that it uses an over-complete and redundant dictionary (base) while the conventional decomposition theories use a common complete and orthogonal base. The base elements have many restrictions. Thus, when signals are decomposed based on the complete and orthogonal dictionary, many structural characters of signals often cannot be fully reflected by the base elements, and the representation of signals is not sparse enough. In contrast, the amount of atoms in an overcomplete dictionary is significantly more than that of elements in a complete dictionary. By designing a suitable algorithm, the atoms can be changed according to the character of signals and thus can approach the structure of the signals. Therefore, the over-complete dictionary reflects the character of signals more faithfully and the solution vector produced in signal decomposition is more sparse, which is conducive to signal compression and feature extraction. Based on the over-complete dictionary and in accord with the characters of signals, we can obtain the signal approximation, which is the best linear combination by M atoms of the dictionary. This theory is called sparse decomposition, and the approximation is called a sparse approach of signal. The linear representation of the signal is called a sparse representation of signal.

Suppose $\boldsymbol{D} \in \mathbb{R}^{n \times K}$, $\boldsymbol{y} \in \mathbb{R}^n$, $\boldsymbol{x} \in \mathbb{R}^K$ represent the over-complete dictionary ($K > n$), the original signal and the solution vector of sparse decomposition of the original signal, respectively. The formula of sparse decomposition is

$$\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 \ \text{s.t.} \ \boldsymbol{y} = \boldsymbol{D}\boldsymbol{x}, \tag{1}$$

or

$$\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 \ \text{s.t.} \ \|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{x}\|_2 \leqslant \varepsilon. \tag{2}$$

Equation (1) is a sparse decomposition that needs accurate reconstruction, while Eq. (2) is a sparse decomposition that needs only sparse approximate representation.

2.2　OMP algorithm

OMP algorithm is an important greedy optimal atom searching algorithm. When the dictionary is N-dimensional and N is finite, the OMP can converge in N steps, i.e., the OMP overcomes the disadvantage of slow convergence of the MP algorithm.

We suppose that $\boldsymbol{f} \in \mathbb{R}^n$ represents the original signal, $\boldsymbol{D} = \{\boldsymbol{x}_i\} \in \mathbb{R}^{n \times K}$ represents an overcomplete dictionary, where the norms of all atoms are 1, and $R^k \boldsymbol{f}$ is the kth iterative remaining signal. Before initialization, let $\boldsymbol{f}_0 = 0$, $R^0 \boldsymbol{f} = \boldsymbol{f}$, $\boldsymbol{x}_0 = 0$, $a_0^0 = 0$, $k = 0$. After the kth step decomposition, the signal can be represented as

$$\boldsymbol{f} = \sum_{n=1}^{k} a_n^k \boldsymbol{x}_n + R^k \boldsymbol{f} = \boldsymbol{f}_k + R^k \boldsymbol{f},$$

$$\langle \boldsymbol{x}_n, R^k \boldsymbol{f} \rangle = 0, \ n = 1,2,...,k. \tag{3}$$

Here, a_n^k represents the coefficient produced in the kth step decomposition, and the signal decomposition in the $(k+1)$th step is

$$\boldsymbol{f} = \sum_{n=1}^{k+1} a_n^{k+1} \boldsymbol{x}_n + R^{k+1} \boldsymbol{f},$$

$$\langle \boldsymbol{x}_n, R^{k+1} \boldsymbol{f} \rangle = 0, \ n = 1,2,...,k+1, \tag{4}$$

$$\boldsymbol{x}_{k+1} = \sum_{n=1}^{k} b_n^k \boldsymbol{x}_n + \boldsymbol{\gamma}_k,$$

$$\langle \boldsymbol{\gamma}_k, \boldsymbol{x}_n \rangle = 0, \ n = 1,2,...,k. \tag{5}$$

Here, $\sum_{n=1}^{k} b_n^k \boldsymbol{x}_n = \boldsymbol{P}_{V_k} \boldsymbol{x}_{k+1}$ represents the projection of $\boldsymbol{x}_{k+1}$ on $\{\boldsymbol{x}_1, \boldsymbol{x}_2,..., \boldsymbol{x}_k\}$, and $\boldsymbol{\gamma}_k = \boldsymbol{P}_{V_k^{\perp}} \boldsymbol{x}_{k+1}$ represents the element of $\boldsymbol{x}_{k+1}$, which is the vertical component of $\{\boldsymbol{x}_1, \boldsymbol{x}_2,..., \boldsymbol{x}_k\}$.

$$a_n^{k+1} = a_n^k - \alpha_k b_n^k, \ n = 1,2,...,k,$$

$$a_{k+1}^{k+1} = \alpha_k. \tag{6}$$

Here,

$$\alpha_k = \frac{\langle R^k \boldsymbol{f}, \boldsymbol{x}_{k+1} \rangle}{\langle \boldsymbol{\gamma}_k, \boldsymbol{x}_{k+1} \rangle} = \frac{\langle R^k \boldsymbol{f}, \boldsymbol{x}_{k+1} \rangle}{\|\boldsymbol{\gamma}_k\|^2}$$

$$= \frac{\langle R^k \boldsymbol{f}, \boldsymbol{x}_{k+1} \rangle}{\|\boldsymbol{x}_{k+1}\|^2 - \sum_{n=1}^{k} b_n^k \langle \boldsymbol{x}_n, \boldsymbol{x}_{k+1} \rangle}.$$

The remaining signal $R^{k+1} \boldsymbol{f}$ satisfies the condition of

$$R^{k+1} \boldsymbol{f} = R^k \boldsymbol{f} - \alpha_k \boldsymbol{\gamma}_k$$

and

$$\|R^{k+1}\boldsymbol{f}\|^2 = \|R^k\boldsymbol{f}\|^2 - \frac{\langle R^k\boldsymbol{f}, \boldsymbol{x}_{k+1}\rangle^2}{\|\boldsymbol{\gamma}_k\|^2}.$$

The detailed algorithm and proofs can be found in Ref. [6].

2.3 K-SVD

The K-SVD [7] algorithm has a close connection with K-means cluster algorithm. When only one atom is demanded to approach each signal, K-SVD is degraded into K-means. Suppose that $\boldsymbol{D}\in\mathbb{R}^{n\times K}$, $\boldsymbol{y}\in\mathbb{R}^n$, and $\boldsymbol{x}\in\mathbb{R}^K$ represent the dictionary, training signal and coefficient vector in sparse representation of training signal, respectively, and $\boldsymbol{Y}=\{\boldsymbol{y}_i\}_{i=1}^N$ represents the sets of N training signals and $\boldsymbol{X}=\{\boldsymbol{x}_i\}_{i=1}^N$ represents the sets of solution vector of $\boldsymbol{Y}$. Then the objective function of K-SVD is

$$\min_{\boldsymbol{D},\boldsymbol{X}}\left\{\|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2\right\}, \text{ s.t. } \forall i, \|\boldsymbol{x}_i\|_0\leqslant l. \qquad (7)$$

Here, l is the upper limit of diversity, which is the number of non-zero elements in each solution vector.

Updating the dictionary is done column by column in the process of learning the K-SVD. Suppose the kth column $\boldsymbol{d}_k$ in the dictionary is to be updated, then

$$\|\boldsymbol{Y}-\boldsymbol{D}\boldsymbol{X}\|_F^2 = \left\|\boldsymbol{Y}-\sum_{j=1}^K \boldsymbol{d}_j\boldsymbol{x}_T^j\right\|_F^2$$
$$= \left\|\left(\boldsymbol{Y}-\sum_{j\neq k}\boldsymbol{d}_j\boldsymbol{x}_T^j\right)-\boldsymbol{d}_k\boldsymbol{x}_T^k\right\|_F^2$$
$$= \|\boldsymbol{E}_k-\boldsymbol{d}_k\boldsymbol{x}_T^k\|_F^2. \qquad (8)$$

Here, $\boldsymbol{x}_T^k$ represents the kth row of $\boldsymbol{X}$ (which is different from the kth column of $\boldsymbol{X}$, i.e., $\boldsymbol{x}_k$). Define the set $\omega_k = \{i\,|\,1\leqslant i\leqslant N, \boldsymbol{x}_T^k(i)\neq 0\}$ as the index set of signal $\{\boldsymbol{y}_i\}$ that uses $\boldsymbol{d}_k$ in the decomposition, i.e., the index set of $\boldsymbol{x}_T^k(i)\neq 0$. Define $\boldsymbol{\Omega}_k$ as a matrix of size $N\times|\omega_k|$, with ones on the $(\omega_k(i),i)$th entries, and zeros elsewhere. Define $\boldsymbol{x}_R^k=\boldsymbol{x}_T^k\boldsymbol{\Omega}_k$, $\boldsymbol{Y}_R^k=\boldsymbol{Y}\boldsymbol{\Omega}_k$, $\boldsymbol{E}_R^k=\boldsymbol{E}_k\boldsymbol{\Omega}_k$, which are the shrink results of the row vector $\boldsymbol{x}_T^k$, $\boldsymbol{Y}$, $\boldsymbol{E}_k$ by discarding the zero entries. The length of $\boldsymbol{x}_R^k$ is $|\omega_k|$, hence $\boldsymbol{Y}_R^k$ and $\boldsymbol{E}_R^k$ are $n\times|\omega_k|$ matrices. $\boldsymbol{E}_R^k$ can be done directly via SVD, and its SVD decomposition is $\boldsymbol{E}_R^k=\boldsymbol{U}\Delta\boldsymbol{V}^{\mathrm{T}}$. Define $\tilde{\boldsymbol{d}}_k$ as the first column of $\boldsymbol{U}$, then $\tilde{\boldsymbol{d}}_k$ is the update of $\boldsymbol{d}_k$. At the same time, update the coefficient vector $\boldsymbol{x}_R^k$ with the product of the first column of $\boldsymbol{V}$ and $\Delta(1,1)$. After updating the dictionary, sparse decomposition is performed on the new dictionary $\tilde{\boldsymbol{D}}$, and if the stop condition is satisfied, K-SVD will stop. The stop condition can either be the iteration times or the error ratio between the original signal and the reconstructed signal.

The K-SVD algorithm is very flexible and can be combined with some familiar optimal atom searching algorithms of sparse decomposition such as matching pursuit, orthogonal matching pursuit, basis pursuit [8] and FOCUSS [9].

3 Compression algorithm for ECGs based on sparse decomposition

The computational complexity of sparse decomposition changes at an exponential rate with the length of atoms in the over-complete dictionary. Since this is an NP problem, sparse decomposition thus cannot be used in the real time problem. In data compression, the most important factors are compression ratio and distortion degree, and the demand of real-time computation is not too high. Thus, it is the familiar method for compression algorithm that uses a great deal of computation to obtain a high compression ratio and low distortion.

First, we decide the length n and the number K of atoms in the over-complete dictionary, and the ECG used in training is split into sample vector $\boldsymbol{y}$ whose length is n. These vectors become training matrix $\boldsymbol{Y}$. The initialization of the dictionary can be realized by taking K n-dimensional vectors of $\boldsymbol{Y}$ as its atoms or by producing K atoms randomly. The former is used in our algorithm. The maximal diversity l of sparse decomposition is fixed, which represents the maximal number of atoms representing the original signals. The stop condition of iteration is then decided, which can be the times of iteration L or error ratio ε or the combination of L and ε. After the initialization, the iteration computation begins until the stop condition is satisfied. First, the set of solution vectors of $\boldsymbol{Y}$ is decomposed using OMP algorithm; second, the K-SVD algorithm is used to update each atom in the dictionary and the corresponding row of $\boldsymbol{X}$; the stop condition is then judged. If the stop condition is satisfied, the iteration completes, otherwise, the iteration continues. The algorithm flowchart is shown in Fig. 1.
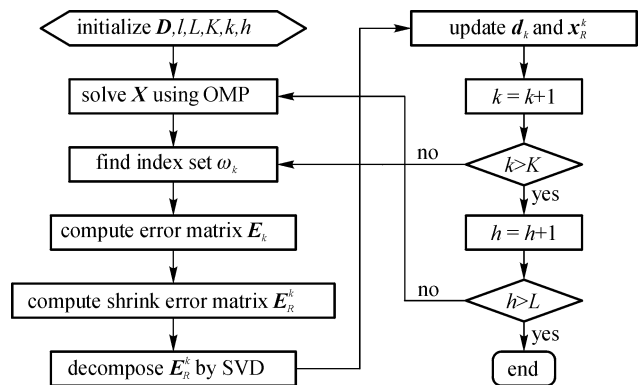

Fig. 1 Flowchart of the algorithm

Based on the overcomplete dictionary obtained by training, the ECG can be decomposed into a sparse solution vector. When storing the ECG, only a very sparse solution vector and the overcomplete dictionary are needed. The effects of the dictionary on the compression ratio can be ignored when the amount of compression data is large enough. The ECG can be reconstructed with high quality using the solution vector and the overcomplete dictionary.

4 Experimental results and analysis

4.1 Compression result for same dimension and different diversity

An ECG datum is selected randomly as experimental datum from the MIT-BIH Arrhythmia Database (in practice the 100th record is selected), and its sample ratio is 360 Hz. First, some vectors of suitable length are cut from the ECG datum to construct the training matrix Y. The first K columns of the training matrix are then considered as the initial over-complete dictionary. The number of non-zero values in the solution vector is no more than l, and the number of training iterations is no more than 50. Next, the over-complete dictionary D is obtained by training, at the same time, the solution vector matrix of the training matrix is obtained. Finally, by using the OMP algorithm, the non-training part of the datum is decomposed into a sparse vector based on the above dictionary. The sparse decomposition of the whole ECG datum is realized. If the effect of the over-complete dictionary is ignored, the compression ratio is equal to $n:2l$. Defining the reconstructed training matrix as $\hat{Y}$, the formula of relative root mean square error (RRMSE) in the compression algorithm can be

$$\text{RRMSE} = \sqrt{\frac{\sum_{i=1}^{K}\sum_{j=1}^{N}(Y(i,j) - \hat{Y}(i,j))^2}{\sum_{i=1}^{K}\sum_{j=1}^{N}Y(i,j)^2}} \times 100\%.$$

When $n = 120$, $K = 240$, $N = 3000$ and the value of l is different, the experimental results are as follows:

Table 1 Compression ratio (CR) and RRMSE of the algorithm for different l

l	3	4	5	6
CR	20:1	15:1	12:1	10:1
RRMSE/%	5.11	4.18	3.60	3.17

When $l = 6$, the waveforms of the first 10 atoms in the over-complete dictionary are shown in Fig. 2. It can be seen that the waveform and wave width of these atoms reflect the character of ECG intensively, to the exclusion of the atoms whose frequency range are higher than that of the ECG in the over-complete dictionary.
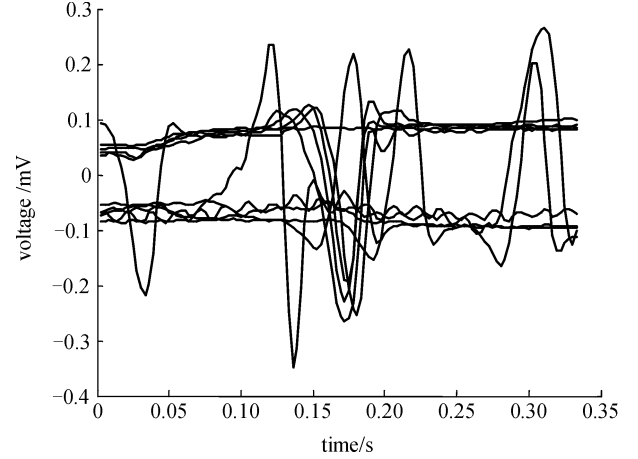

Fig. 2 Waveforms of the first 10 atoms

Another ECG not included in the training data is selected as experimental data and sparse decomposition is performed under the condition of $l = 6$. The experimental results are shown in Fig. 3. Comparing the original ECG with the reconstructed ECG shows that the latter is smoother than the former, which means that the high frequency noise of the original ECG is filtered out and all of the useful signals are retained, including bitty incisure, denoted by arrows in Fig. 3.

4.2 Compression result for different data

A few ECG data are selected as experimental data from the MIT-BIH Arrhythmia Database. These data include some kinds of arrhythmia and all types of ECG exceptions caused by drugs and breaths. When $n = 60$, $K = 120$, $N = 6000$, $l = 6$, the RRMSEs of ECG data are shown in Table 2. From Table 2, it can be concluded that when CRs are 5:1, RRMSEs are low, i.e., distortion is low under the above condition. These results mean that our compression algorithm can be applied to abnormal ECGs.

4.3 Comparison with other compression algorithms of ECGs

For the same ECG data in the MIT-BIH Arrhythmia Database, by using DCT transform [1] to compress the data when the average value of CR reaches 15.76, the average RRMSE is 6.76; by using wavelet transform [2] to compress the data when the average value of CR reaches 10.6, the average RRMSE is 5.3. Compared with the experimental results in Tables 1 and 2, the RRMSE and distortion of our compression is lower under the condition of similar CRs.
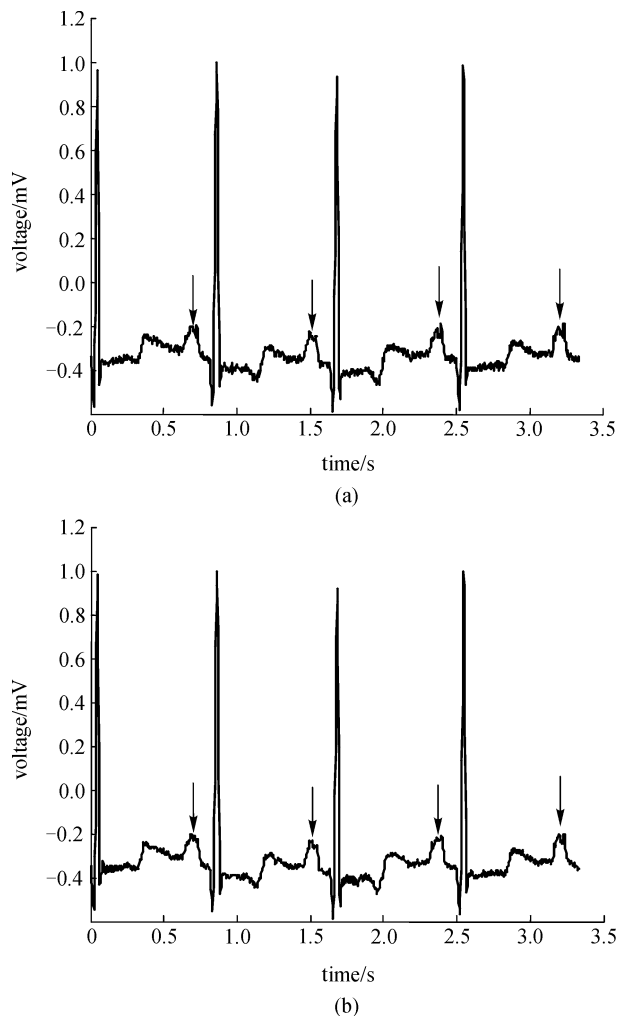
are infinite, the length of the atoms in an overcomplete dictionary can be any natural number in theory, and thus the CR of an ECG compression algorithm based on sparse decomposition can be infinite in theory. Obviously, infinite CR cannot be achieved because of computational complexity. However, this algorithm brings forward a theoretically high CR in the compression of ECG. In practice, the length of atoms in an overcomplete dictionary is decided by application requirements.

Sparse decomposition is a new theory in signal processing. Although its development and application are not perfect, its strong potential can be foreseen from current extensive and successful applications. In this article, the application of sparse decomposition in ECG data compression is studied. By learning and training on the ECG database, the dictionary corresponding with the characteristics of ECGs is constructed. With this dictionary, ECG signals are sparsely decomposed and compressed. The experiments demonstrate that the compression results of this new theory are better compared to earlier theories.

Fig. 3 ECG and its reconstruction. (a) Original ECG; (b) reconstructed ECG

Table 2 RRMSEs of different ECGs (CRs are all 5:1)

number	101	102	117	202	230
RRMSE/%	3.20	3.79	1.83	3.06	3.08

The main computation in this compression algorithm concentrates on learning and construction of the overcomplete dictionary. The OMP algorithm is then relatively easier in sparse decomposition based on this dictionary. Overall, the algorithm proposed in this paper is complicated because of the overcomplete dictionary and the greedy searching algorithm (OMP).

5 Algorithm discussion and conclusions

When the ECG data amount is large enough and the effect of the overcomplete dictionary can be ignored, the CR of our ECG compression algorithm is $n:2l$. If the ECG data

References

1. Cheng W, Fang B, Shen Y. A new ECG compression method based on 2-step VQ of DCT coefficients. Chinese Journal of Biomedical Engineering, 2005, 24(6): 690–695 (in Chinese)
2. Kou P, Fang B, Shen Y. A reconstruction quality controlled compression algorithm for ECG signal. Beijing Biomedical Engineering, 2004, 23(2): 109–111 (in Chinese)
3. Wang X Y, Meng J. Hybrid 2-D ECG compression method based on wavelet transform. Acta Biophysica Sinica, 2006, 22(3): 217–224 (in Chinese)
4. Alesanco A, Olmos S, Istepanian R, Garcia J. A novel real-time multilead ECG compression and de-noising method based on the wavelet transform. In: Proceedings of Computers in Cardiology, 2003, 593–596
5. Mallat S G, Zhang Z F. Matching pursuits with time-frequency dictionaries. IEEE Transactions on Signal Processing, 1993, 41(12): 3397–3415
6. Pati Y C, Rezaiifar R, Krishnaprasad P S. Orthogonal matching pursuits: recursive function approximation with applications to wavelet decomposition. In: Proceedings of the 27th Asilomar Conference in Signals, Systems and Computers. California: IEEE Computer, 1993: 1–5
7. Aharon M, Elad M, Bruckstein A M. K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing, 2006, 54(11): 4311–4322
8. Chen S S, Donoho D L, Saunders M A. Atomic decomposition by basis pursuit. SIAM Review, 2001, 43(1): 129–159
9. Kreutz-Delgado K, Murray J F, Rao B D, Engan K, Lee T W, Sejnowski T J. Dictionary learning algorithms for sparse representation. Neural Computation, 2003, 15(2): 349–396