BioMedical Engineering OnLine

RESEARCH   **Open Access**

Automatic online detection of atrial fibrillation based on symbolic dynamics and Shannon entropy

Xiaolin Zhou[1,2]*, Hongxia Ding[1,2], Benjamin Ung[3], Emma Pickwell-MacPherson[3] and Yuanting Zhang[1,2,3]*

*Correspondence:
xl.zhou@siat.ac.cn;
ytzhang@ee.cuhk.edu.hk
[1] Institute of Biomedical and Health Engineering, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Xili Nanshan, Shenzhen 518055, China
[2] The Key Laboratory for Health Informatics of the Chinese Academy of Sciences at Shenzhen Institutes of Advanced Technology, Shenzhen 518055, China
Full list of author information is available at the end of the article

Abstract

Background: Atrial fibrillation (AF) is the most common and debilitating abnormalities of the arrhythmias worldwide, with a major impact on morbidity and mortality. The detection of AF becomes crucial in preventing both acute and chronic cardiac rhythm disorders.

Objective: Our objective is to devise a method for real-time, automated detection of AF episodes in electrocardiograms (ECGs). This method utilizes RR intervals, and it involves several basic operations of nonlinear/linear integer filters, symbolic dynamics and the calculation of Shannon entropy. Using novel recursive algorithms, online analytical processing of this method can be achieved.

Results: Four publicly-accessible sets of clinical data (Long-Term AF, MIT-BIH AF, MIT-BIH Arrhythmia, and MIT-BIH Normal Sinus Rhythm Databases) were selected for investigation. The first database is used as a training set; in accordance with the receiver operating characteristic (ROC) curve, the best performance using this method was achieved at the discrimination threshold of 0.353: the sensitivity (*Se*), specificity (*Sp*), positive predictive value (*PPV*) and overall accuracy (*ACC*) were 96.72%, 95.07%, 96.61% and 96.05%, respectively. The other three databases are used as testing sets. Using the obtained threshold value (i.e., 0.353), for the second set, the obtained parameters were 96.89%, 98.25%, 97.62% and 97.67%, respectively; for the third database, these parameters were 97.33%, 90.78%, 55.29% and 91.46%, respectively; finally, for the fourth set, the *Sp* was 98.28%. The existing methods were also employed for comparison.

Conclusions: Overall, in contrast to the other available techniques, the test results indicate that the newly developed approach outperforms traditional methods using these databases under assessed various experimental situations, and suggest our technique could be of practical use for clinicians in the future.

Keywords: ECG, RR interval, Atrial fibrillation, Nonlinear filter, Integer filter, Symbolic dynamics, Shannon entropy

Background

Atrial fibrillation (AF) is recognized as the most common clinically encountered arrhythmia in adults [1], which affects approximately 0.4% of the general population. The prevalence of this tachyarrhythmia increases with age, with less than 1% affected in persons under the age of 60 years and in excess of 6% for those over the age of 80 years [2,3]. Atrial fibrillation is associated with a high risk of stroke, heart disease (e.g., congestive

cardiac failure), and cardiovascular mortality [1,4]. There is also a close relationship between AF and obesity [5], obstructive sleep apnea [6], and long-term alcoholism [7], which reciprocally bear cumulative risks for promoting the development of AF [1]. The early identification of AF appears to be crucial for patients with cardiovascular disease, especially for stroke patients to whom the secondary stroke prevention is of primary importance.

Issues relating to clinical significance of rhythm classification and the impetus for improving the accuracy of atrial tachyarrhythmia estimation have motivated the development of innovative computerized AF detectors. Since the early 1980s, a series of sophisticated methods have been investigated to cope with the challenges of AF detection [8-25]. Most of which are based upon two main character traits of this type of arrhythmia shown in a surface electrocardiogram (ECG): (*i*) RR (R-wave peak to R-wave peak) interval irregularity (i.e., chaotic behavior of heart rate variability), and (*ii*) P-wave absence (PWA) or F-wave substitution (i.e., very low amplitude waveforms of odd morphologies) resulting from the abnormal rapid atrial activity (AA). Although P waves or cardiac AA can be an alternative clue in the detection of AF, the absence or presence of P waves are not readily identifiable as various types of high-intensity noise often coexist in ECGs, which may lead to a low degree of predictive accuracy. In addition, the relationship between AA in the surface ECG and the diverse mechanisms of AF has not yet been well delineated [3]. Due to the challenges in detecting AA in ECG measurements, detection techniques based on inferences from RR intervals are preferred to produce relatively robust outcomes [21-23,25].

In this study, a reliable method for the fully automated detection of AF episodes from surface ECGs is proposed. This method comprises of a three-pass procedure. The initial pass, where a RR interval sequence is pre-processed with nonlinear and integer filters, which aims to generate low/high scale reference sequences. The second pass, which aims to obtain a symbolic sequence, where the information of the RR interval sequence is subsequently compressed by the symbolic dynamics with sequences obtained from the initial pass. Finally, Shannon entropy is used in the third pass, to calculate the entropy of the symbolic sequence and thereby discriminate whether or not AF is present in the current cardiac beat. Further methodological insight of present key points on the online analytical processing of measurements through the recursive realization with respect to beat-by-beat classification is discussed in the following sections. Ultimately, we quantitatively investigate the performance of our newly developed technique to that of currently state-of-the-art techniques with four widely used clinical databases under various experimental situations.
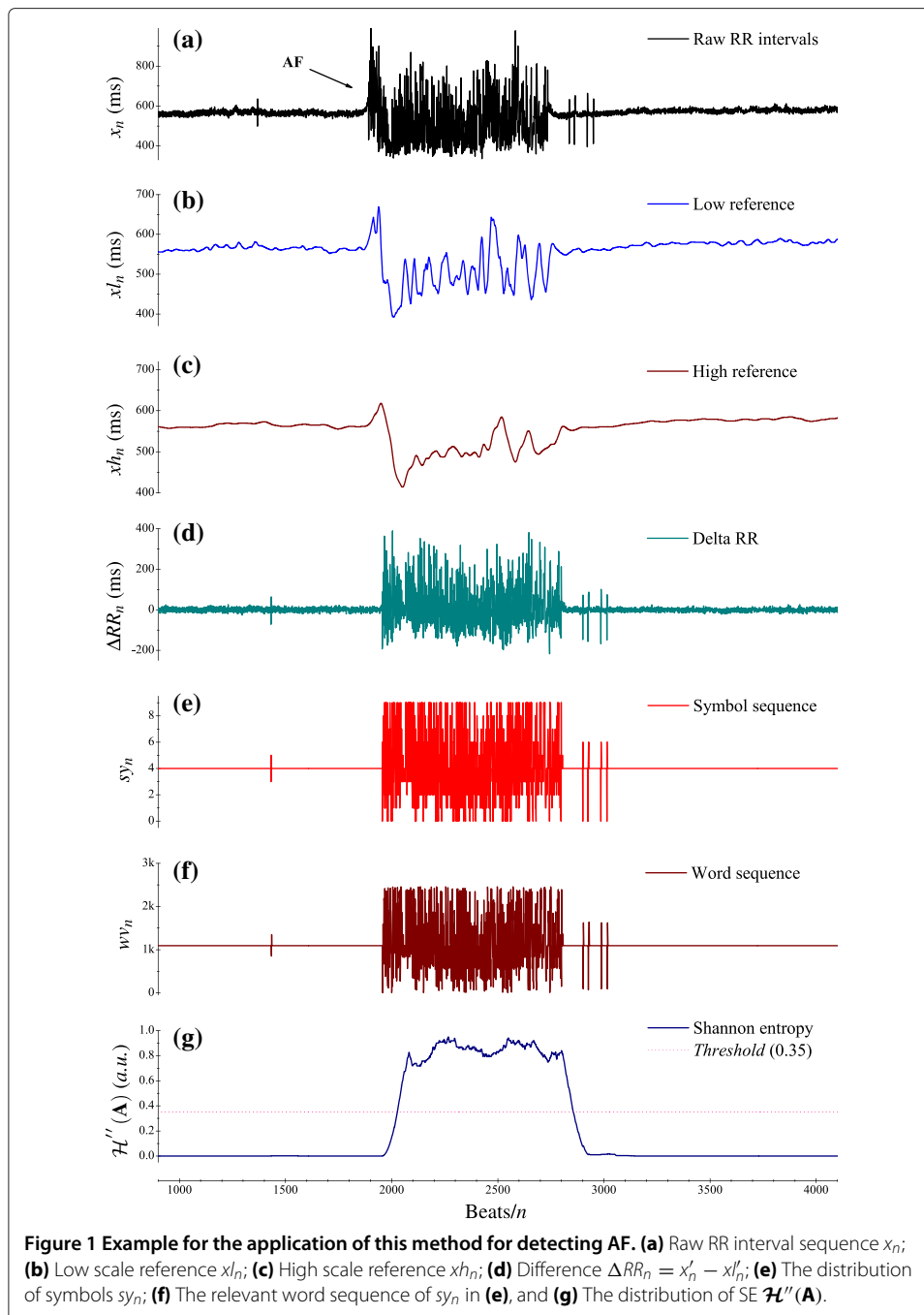
Methodology

Pre-processing of RR_n series

A. Median filter

A median filter is implemented by windowing the acquired data, ranking the samples in the window, and outputting the median of the sorted samples. Considering a RR interval (RR_n) sequence x_n, as shown in Figure 1(a), the output y_n of this nonlinear filter is given by,

$$y_n = \text{median}\{x_{n-w}, \cdots, x_n, \cdots, x_{n+w}\} \tag{1}$$

where the window is of a fixed width $2w + 1$. From the perspective of signal processing, the time delay of the median filter is w. A window size of 17 is used herein, with a delay

Figure 1 Example for the application of this method for detecting AF. (a) Raw RR interval sequence x_n; **(b)** Low scale reference xl_n; **(c)** High scale reference xh_n; **(d)** Difference $\Delta RR_n = x'_n - xl''_n$; **(e)** The distribution of symbols sy_n; **(f)** The relevant word sequence of sy_n in **(e)**, and **(g)** The distribution of SE $\mathcal{H}''(\mathbf{A})$.

of 8 samples. The introduction of a median filter brings about two advantages: (*i*) the suppression of unwanted outliers, which are mostly caused by erroneously detected (or missed) R-wave peaks; (*ii*) to preserve sharp edges (i.e., onsets and terminations of AF episodes) without extensively blurring the context.

B. Integer filter for low scale reference

Subsequently, we filter the output y_n of median filter with a low-pass filter of the form

$$H_l(z) = \frac{1 - z^{-16}}{1 - z^{-1}} \tag{2}$$

where, the gain is $Gain1 = 16 = 2^4$, and the intrinsic delay of $H_l(z)$ is 7.5 samples. This low-pass filter is applied to smooth y_n resulting from the previous median filtering. Another benefit of the low-pass filter is the removal of fluctuations possibly caused by Respiratory Sinus Arrhythmia (RSA) phenomena around the current sample from acquisition. Let xl_n be the output of this filter, as illustrated in Figure 1(b).

C. Integer filter for high scale reference

Another low-pass filter $H_h(z)$ is then applied to the resultant xl_n of the previous low-pass filter $H_l(z)$,

$$H_h(z) = \frac{1 - z^{-32} - z^{-64} + z^{-96}}{1 - 2z^{-1} + z^{-2}} \tag{3}$$

where, the gain is $Gain2 = 2048 = 2^{11}$, and the relevant delay of $H_h(z)$ is 47 samples. This low-pass filter is introduced to generate a reference RR sequence of a larger scale, which needs to be exploited in the definition of symbolic series as explained in the following subsection. The resulting output denoted by xh_n is shown in Figure 1(c).

As we have seen, the time delays of x_n and xl_n are -62.5 and -47 samples with respect to xh_n, respectively. To ensure synchronization of the filtered data, let x'_n and xl'_n denote the corresponding time-delay corrected sequences of x_n and xl_n, respectively. Then, $\Delta RR_n = x'_n - xl'_n$ can be defined as the difference in time delay, seen in Figure 1(d).

Symbolic dynamics of ΔRR_n

The purpose of employing symbolic dynamics is to describe the dynamic behavior of ΔRR_n with respect to xh_n. Symbolic dynamics encodes the information as a variation of RR_n to a series with fewer symbols, with each symbol representing an instantaneous state. The implemented thresholds can be defined as: $thre1 = xh_n \times 2^{-4}$ (with $thre1 = xh_n >> 4$), $thre2 = xh_n \times 2^{-3}$ (with $thre2 = xh_n >> 3$), $thre3 = thre1 + thre2$, $thre4 = xh_n \times 2^{-2}$ (with $thre4 = xh_n >> 2$) and $thre5 = thre4 + thre1$. The mapping function of the symbol transform can therefore be defined as,

$$sy_n = \begin{cases} 0 & \text{if } \Delta RR_n < -thre4 \\ 1 & \text{else if } \Delta RR_n < -thre3 \\ 2 & \text{else if } \Delta RR_n < -thre2 \\ 3 & \text{else if } \Delta RR_n < -thre1 \\ 4 & \text{else if } \Delta RR_n < thre1 \\ 5 & \text{else if } \Delta RR_n < thre2 \\ 6 & \text{else if } \Delta RR_n < thre3 \\ 7 & \text{else if } \Delta RR_n < thre4 \\ 8 & \text{else if } \Delta RR_n < thre5 \\ 9 & \text{other cases} \end{cases} \tag{4}$$

The raw RR sequence x_n is then quantified into symbol sequence sy_n with specific symbols from the predefined "alphabet" in Eq. (4) (i.e., 0 to 9). Recalling Figure 1(a)-(d) and scanning the distribution of calculated symbols in Figure 1(e), we confirm that most of normal beats are defined as zero symbols, and possible abnormal beats (arrhythmias, e.g., AF) are defined as non-zero symbols by the transform Eq. (4).

To facilitate the analysis of sy_n, the widely used 3-symbol template (i.e., a word consists of 3 successive symbols) is applied to examine entropic properties. The word value can then be calculated by a novel operator as defined below,

$$wv_n = (sy_{n-2} \times 2^8) + (sy_{n-1} \times 2^4) + sy_n \tag{5}$$

where, $sy_{n-2} \times 2^8$ and $sy_{n-1} \times 2^4$ are implemented with $sy_{n-2} << 8$ and $sy_{n-1} << 4$, and $0 \leqslant wv_n \leqslant 2457$. Figure 2 briefly elucidates the transformation of the symbol sequence with the template and the corresponding word, while Figure 1(f) depicts the word sequence of sy_n shown in Figure 1(e).
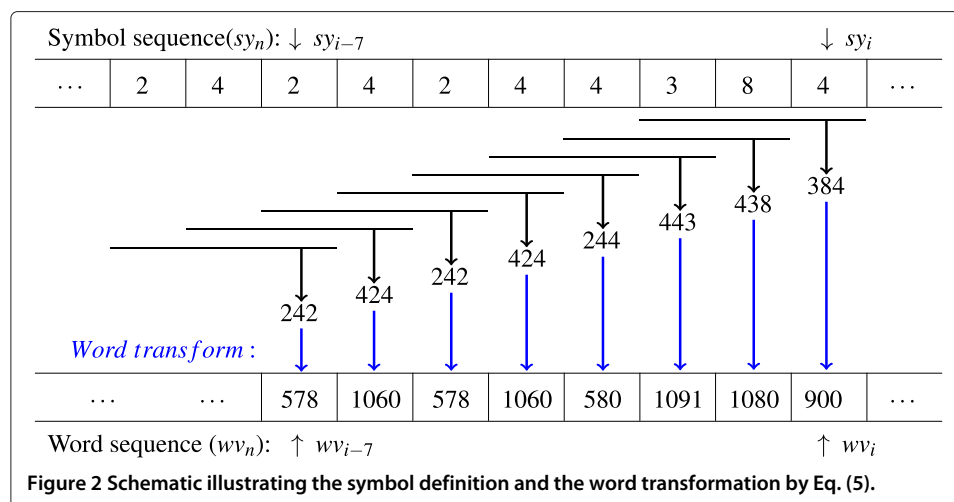
Shannon entropy

Shannon entropy (SE) is a statistical tool that quantifies a time series in terms of the information size. For the sake of completeness, we define the discrete probability space of a dynamic system as $\mathbf{A} = (A|P)$. The total number of elements in $\mathbf{A}$ is N. The characteristic elements can then be defined as $A = \{a_1, \cdots, a_k\}$, as well as the relevant probability set $P = \{p_1, \cdots, p_k\}(1 \leqslant k \leqslant N)$. Each element a_i has probability $p_i = N_i/N$ $(0 < p_i \leqslant 1, \sum_{i=1}^{k} p_i = 1)$, where N_i is the total number of the element a_i in $\mathbf{A}$. Thus, the SE of $\mathbf{A}$ is defined as [26],

$$\mathcal{H}(\mathbf{A}) = -\sum_{i=1}^{k} p_i \log_2 p_i \tag{6}$$

By Jensen's inequality, we can prove $\mathcal{H}(\mathbf{A}) \leqslant \log_2 k \leqslant \log_2 N$ with equality if $p_i \equiv 1/N$ *and* $k \equiv N$ for all i. Then, a uniform distribution of $\mathcal{H}(\mathbf{A})$ can be expressed as,

$$\mathcal{H}'(\mathbf{A}) = -\frac{1}{\log_2 N} \sum_{i=1}^{k} p_i \log_2 p_i \tag{7}$$

Figure 2 Schematic illustrating the symbol definition and the word transformation by Eq. (5).

where, if $N \equiv 1$, make $\log_2 N = 1$. Eq. (7) is also referred to as the normalized entropy, since the entropy is divided by the maximum entropy $\log_2 N$. A coarser version of $\mathcal{H}'(\mathbf{A})$ can be defined as,

$$\mathcal{H}''(\mathbf{A}) = -\frac{k}{N \log_2 N} \sum_{i=1}^{k} p_i \log_2 p_i \qquad (8)$$

Currently, the dynamic $\mathbf{A}$ consists of all 127 consecutive word elements from wv_{n-126} to wv_n (the bin size in this case is $N = 127$). By determining the characteristic set A and the relevant probability set P with these elements, we can thus calculate the SE $\mathcal{H}''(\mathbf{A})$. The presence of AF is then detectable, with the rhythm labeled AF if $\mathcal{H}''(\mathbf{A})$ exceeds a discrimination threshold, and otherwise non-AF, which can be seen in Figure 1(g). We utilize the training database to determine the optimal discrimination threshold by investigating various threshold settings which lie within the range [0.0, 1.0]; the best performing threshold of 0.353 is thus derived and employed for the performance assessment using different testing databases.

Key issues of online processing
From Eq. (1)–(5) and (8), outwardly, this AF detection technique poses computational challenges. However, these challenges can be overcome by implementing clever recursive algorithms with beat-by-beat, real-time processing.

A. Pseudo-recursive median filtering
The median filter in Eq. (1) can be implemented with a so-called pseudo-recursive method: for input x_i, we define $S = \{s_r \uparrow: 1 \leqslant r \leqslant 2w + 1\}$ as a sorted array of successive elements from x_{i-2w-1} to x_{i-1}, where the output y_i is obtained by following steps ❶-❺ below,

❶ A Binary search technique is used to seek out the position m of the sample x_{i-2w-1} which will depart from the window (i.e., $s_m = x_{i-2w-1}$. Simultaneously, x_i will get into the window);

❷ The Binary search technique is applied again to search for the position t at which the input x_i needs to be set (i.e., $s_t < x_i \leqslant s_{t+1}$);

❸ From positions m to t, the current s_r is replaced with the adjacent $s_{r\pm 1}$ (the '$\pm$' indicates where the element is taken from the right or left, with the '$+$' and '$-$' symbols representing the element to the right and left, respectively);

❹ Replace the element s_t with x_i;

❺ Median s_{w+1} of the updated S becomes output y_i.

For the following input x_{i+1}, we repeat steps ❶ to ❺ and obtain the new output s_{w+1} (i.e., y_{i+1}), as shown in Figure 3, where the sorting utilizes the Binary search technique twice. Comparing our technique with the traditional median filter, the computational complexity can be decreased from approximately $O(n^2)$ to $O(n)$.

B. Recursive implementation of integer filters
The recursive implementation (also referred to as the "difference equation") of the filter $H_l(z)$ can be expressed as,

$$xl_n = xl_{n-1} + y_n - y_{n-16} \qquad (9)$$

Figure 3 Schema of the pseudo-recursive median filtering (the rightward case).
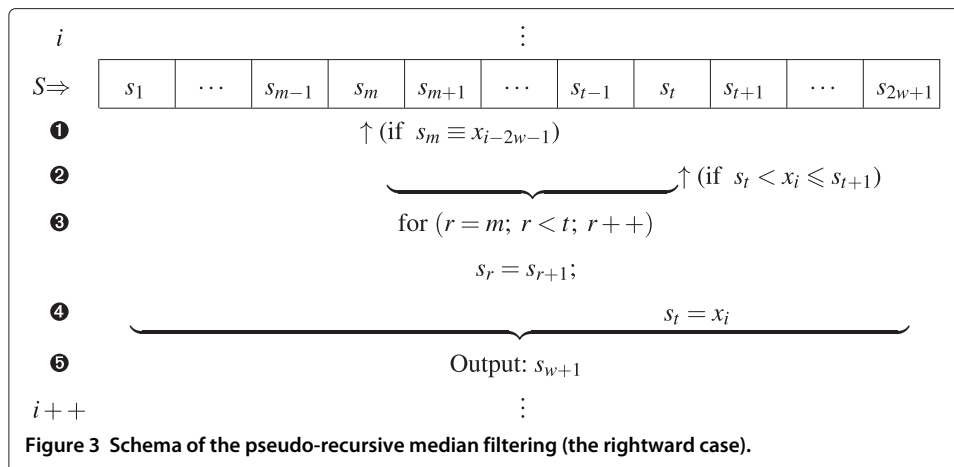
The above equation, Eq. (9) includes 1 integer addition, 3 integer subtractions as well as 1 integer right-shift operation, when $xl_n >> 4$ (as $Gain1 = 2^4$) to offset the gain of $H_l(z)$. The filter $H_h(z)$ can then be computed recursively using

$$xh_n = (xh_{n-1} \times 2) - xh_{n-2}$$
$$+ xl_n - xl_{n-32} - xl_{n-64} + xl_{n-96}$$

(10)

where, $xh_{n-1} \times 2$ is implemented with $xh_{n-1} << 1$. The above equation, Eq. (10) consists of 2 integer additions, 8 integer subtractions, 1 integer left-shift operation and 1 integer right-shift operation, when $xh_n >> 11$ (as $Gain2 = 2^{11}$) to offset the gain of $H_h(z)$.

C. Mapping the definition of $-\frac{1}{\log_2 N} p_i \log_2 p_i$

Investigating the dynamic **A**, we immediately see that each characteristic symbol of each bin N may have the probability $p_i = i/N$ ($1 \leqslant i \leqslant N$, i.e., $1/N \leqslant p_i \leqslant 1$). Along these lines, a probability array *PiMap* can be pre-calculated,

$$PiMap[127] = -\frac{Cons}{\log_2 N} \left\{ p_1 \log_2 p_1, \cdots, p_{63} \log_2 p_{63}, \right.$$
$$\left. p_{64} \log_2 p_{64}, \cdots, p_{127} \log_2 p_{127} \right\}$$
$$\stackrel{\lfloor \cdot \rfloor}{=} \{7874, \cdots, 71790, 71291, \cdots, 0\}$$

(11)

where, $Cons = 1000000$ is a constant such that decimal floating points can be converted into integers and $N = 127$, and $\stackrel{\lfloor \cdot \rfloor}{=}$ indicates to take the integer part of each $-\frac{Cons}{\log_2 N} p_i \log_2 p_i$.

Notably, for each cardiac cycle screened, this predefined *PiMap* permits the sole operation by picking the straightforward integer (i.e., *PiMap[i]*) from the set *PiMap* in accordance with the index i rather than calculating $-\frac{1}{\log_2 N} p_i \log_2 p_i$ using arithmetic and logarithmic operations. The use of this predefined calculation significantly decreases calculation times.

### D. Recursive implementation of $\mathcal{H}''(\mathbf{A})$

We define a buffer array nu_{wv_i} ($wv_i \leqslant 2457$) to store the number of the ith characteristic element wv_i in space $\mathbf{A}$. For the input wv_n, it will get into $\mathbf{A}$ (i.e., wv_n will be the rightmost element), and simultaneously the leftmost element wv_{n-127} will depart from $\mathbf{A}$, see Figure 2 for clarity. It is obvious that a variation of SE $\mathcal{H}'(\mathbf{A})$ is purely determined by nu_{wv_n} and $nu_{wv_{n-127}}$ in dynamic $\mathbf{A}$. Therefore, $\mathcal{H}''(\mathbf{A})$ is calculated recursively by the algorithm below,

---

① if $wv_n \equiv wv_{n-127}$ then $\boxed{sh'_n = sh'_{n-1}}$;

② else

$\quad$ { $te\_in = nu_{wv_n}$;

$\quad\quad te\_out = nu_{wv_{n-127}}$;

$\quad\quad nu_{wv_{n-127}} - -$;

$\quad\quad nu_{wv_n} + +$;

$\quad\quad$ if $nu_{wv_{n-127}} < 0$ then $nu_{wv_{n-127}} = 0$;

$$\boxed{\begin{aligned} sh'_n = sh'_{n-1}+ \\ (PiMap[\,nu_{wv_n}] - PiMap[\,te\_in\,]\,)+ \\ (PiMap[\,nu_{wv_{n-127}}] - PiMap[\,te\_out]\,) \end{aligned}} \quad (*$$

$\quad\quad$ if $te\_in \equiv 0$

$\quad\quad$ { if $te\_out > 1$ *or* $te\_out \equiv 0$ then $k + +$; }

$\quad\quad$ else

$\quad\quad$ { if $nu_{wv_{n-127}} \equiv 0$ *and* $te\_out \equiv 1$ then $k - -$; }

$\quad$ }

③ $\boxed{sh''_n = \dfrac{k}{127000000} sh'_n}$;

---

where sh'_n and sh''_n represent $\mathcal{H}'(\mathbf{A})$ and $\mathcal{H}''(\mathbf{A})$, respectively; (* indicates that $PiMap[\,i\,] = 0$ is fixed for the case $i \equiv 0$; and $127000000 = N * Cons = 127 \times 1000000$. For the next input wv_{n+1}, steps ①-③ are again executed to obtain sh''_{n+1}. From an online processing perspective, the time delays of sh''_n are 64 and 126.5 samples with respect to xh_n and x_n, respectively.

An architecture of the overall logic of the recursive realization can be seen in Figure 4. By using recursive algorithms, this AF detector consists of several basic operations, such as integer addition/subtraction, integer comparison and integer shifting. In effect, the calculation of sh''_n and distinguishing the current beat x_n, only needs to include 1 multiplication and 1 division lying within $\frac{k}{127000000}$·, together with 1 floating-point comparison between sh''_n and a threshold. Consequently, a useful computational efficiency can be achieved.

Figure 4 Flowchart of the recursive realization of this detector for beat-by-beat assessing AF.
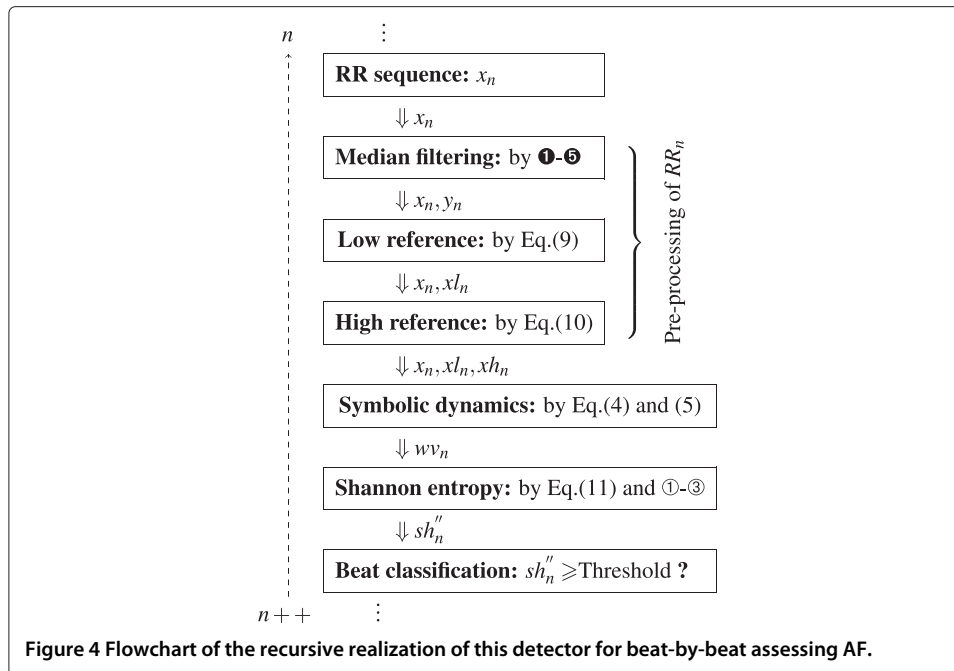
Materials and evaluation

Clinical ECG data sets

Performance of this new AF detection method is evaluated using four popular sets of clinical ECGs (the Long-Term AF Database [LTAFDB], the MIT-BIH AF Database [AFDB], the MIT-BIH Arrhythmia Database [MITDB], and the MIT-BIH Normal Sinus Rhythm Database [NSRDB]) [27]. The LTAFDB database is used as the initial training set, while the other three databases are used as the testing sets. The contents of these databases are summarized in Table 1. All reference annotations of the four databases are examined in this study.

Table 1 Four publicly-accessible sets of clinical data are selected for evaluation

Databases	f_s (Hz)	Total beats (AF beats)	Brief description
LTAFDB	128	8996056 (5326145)	It consists of 84 long-term (typically 24 to 25 hours) ECG recordings of subjects with paroxysmal or sustained AF.
AFDB	250	1221574 (519687)	It contains 25 long-term (10 hours) ECG recordings of subjects with AF (mostly paroxysmal). Of which raw ECG data of two records ("00735" and "03665") are not available, and two records ("04936" and "05091") include many incorrect reference annotations
MITDB	360	109590 (11496)	It is a collection of 48 half-hour two-lead recordings which were arrhythmia obtained from 47 subjects and contains affluent information, such as AF and AFL
NSRDB	128	1729523 (0)	It includes 18 long-term records of subjects. Each recording is about 24 hours in duration. These records had no significant arrhythmias detected in this database

Performance metrics

The performance of our newly developed algorithm and existing methods are investigated in terms of sensitivity (*Se*), specificity (*Sp*), positive predictive value (*PPV*), and overall accuracy (*ACC*),

$$
\begin{aligned}
Se &= \frac{TP}{TP + FN}, \ PPV = \frac{TP}{TP + FP}, \\
Sp &= \frac{TN}{TN + FP}, \ ACC = \frac{TP + TN}{TP + TN + FP + FN}
\end{aligned}
\tag{12}
$$

where, for a specific data set, *TP* (true positive) is the number of beats in AF which are correctly detected as AF, *TN* (true negative) is the number of beats in non-AF which are correctly detected as non-AF, *FP* (false positive) is the number of beats in non-AF which are incorrectly detected as AF, and *FN* (false negative) is the number of beats in AF which are incorrectly detected as non-AF. The proportion of beats in true AF which are correctly identified as AF is represented by *Se*, while *Sp* represents the proportion of beats in true non-AF which are correctly identified as non-AF, *PPV* represents the proportion of algorithm results that are true positive, and *ACC* represents the overall accuracy of our method. We consider *Se* and *Sp* as the main metrics, while *PPV* and *ACC* are complementary.

Results and discussion

The values of SE $\mathcal{H}''(\mathbf{A})$ for AF (519687 beats) and non-AF (701887 beats) annotations in the AFDB database (a total of 1221574 beats for all of the 25 records) can be seen in Figure 5. It is apparent that $\mathcal{H}''(\mathbf{A})$ discriminates AF well.

The receiver operating characteristic (ROC) curves are widely used in the medical field to determine the optimal discrimination threshold for clinical tests. In this work, the LTAFDB database is used as the training set to obtain the optimal threshold for our algorithm. The threshold is tested from 0.0 to 1.0 in increments of 0.001 for the training set,

Figure 5 Histogram distribution of the $\mathcal{H}''(\mathbf{A})$ for annotated AF and non-AF beats of the AFDB database.
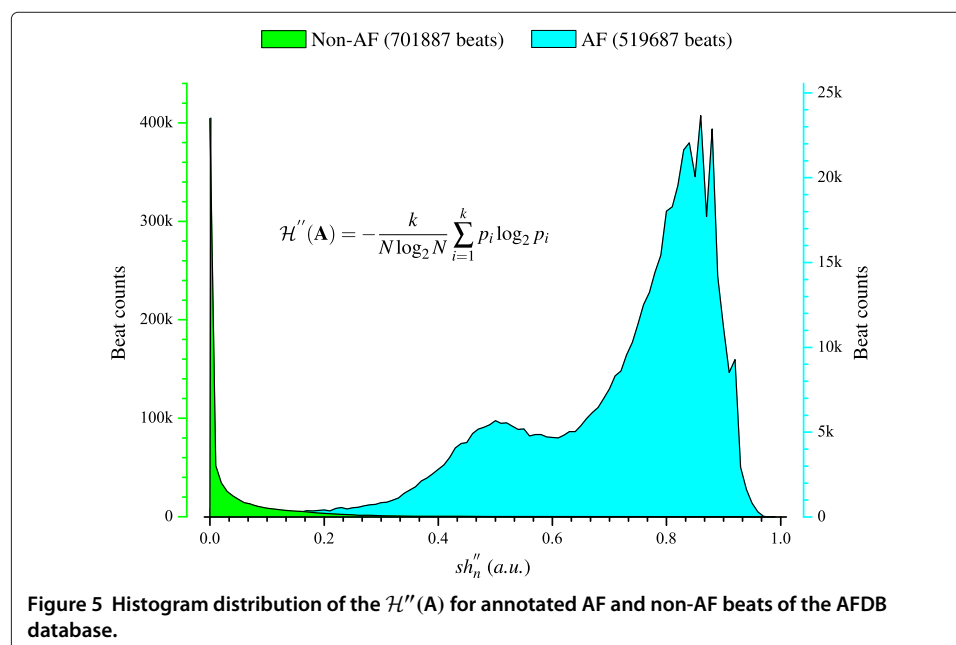
and the values of *Se*, *Sp*, $1 - Sp$, *PPV* and *ACC* are calculated for each threshold setting. Thus we obtain the ROC curve, as shown in Figure 6. In the ROC space of Figure 6, *a* is the point of the perfect classification, at which the *Se* and *Sp* are both equal to 100% and *b* is the point of the best performance of our method on the ROC curve, at which it has the shortest distance to *a*. We can thus determine the parameters at position *b*, where the discrimination threshold is 0.353, and the values for *Se*, *Sp*, *PPV* and *ACC* are 96.72%, 95.07%, 96.61% and 96.05%, respectively. We therefore take the best performing threshold value of 0.353 for quantitative assessment when our method is applied to other three testing databases.

For our newly presented method, the statistical results from the testing sets AFDB, MITDB and NSRDB databases are summarized in Table 2. Specifically, for the AFDB set, the calculated *Se*, *Sp*, *PPV* and *ACC* parameters are 96.89%, 98.25%, 97.62% and 97.67%, respectively. Examining the AFDB[†] set ([†] indicates records "00735" and "03665" omitted), the parameters are 96.82%, 98.06%, 97.61% and 97.50%, respectively. For the AFDB[‡] set ([‡] indicates records "04936" and "05091" omitted), the parameters are 97.83%, 98.19%, 97.56% and 98.04%, respectively. For the MITDB data set, the parameters are 97.33%, 90.78%, 55.29% and 91.46%. It is important to recognize that for MITDB data set, the *PPV* value (55.29%) is low which indicates that many of the positive results are detected as false positives using this testing procedure. Calculating the combined values from these databases, the parameters are 96.89%, 98.27%, 92.30% and 98.03%, respectively for the AFDB+NSRDB set, and 97.53%, 98.26%, 90.09% and 98.16% for the AFDB[‡]+NSRDB set. For the NSRDB set, the only calculated parameter *Sp* is 98.28%, as there is no manual AF annotation in the NSRDB database.

The existing algorithms for the AF detection are also investigated using the same databases (i.e., the same records and the same reference annotations), and using the same evaluation metrics. Table 3 shows a collection of latest published results from prior
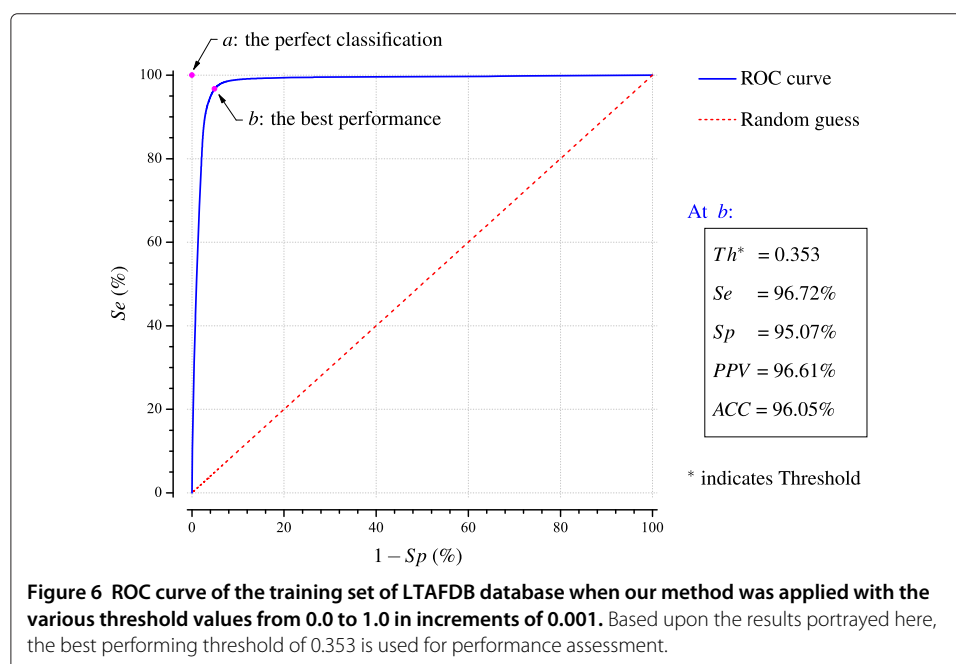

Figure 6 ROC curve of the training set of LTAFDB database when our method was applied with the various threshold values from 0.0 to 1.0 in increments of 0.001. Based upon the results portrayed here, the best performing threshold of 0.353 is used for performance assessment.

Table 2 Statistical results of this method for three testing databases (at the threshold of 0.353)

Method	Features	Year	Database	Key techniques	Results			
					Se(%)	*Sp*(%)	*PPV*(%)	*ACC*(%)
This method	RRI	2013	AFDB	Nonlinear filter + integer filters + symbolic dynamics + SE	96.89	98.25	97.62	97.67
			AFDB[†]		96.82	98.06	97.61	97.50
			AFDB[‡]		97.83	98.19	97.56	98.04
			MITDB		97.33	90.78	55.29	91.46
			NSRDB		NA	98.28	NA	NA
			AFDB+NSRDB		96.89	98.27	92.30	98.03
			AFDB[‡]+NSRDB		97.53	98.26	90.09	98.16

[†]Records "00735" and "03665" omitted.
[‡]Records "04936" and "05091" omitted.
'NA' indicates not applicable because there is no beat with AF reference annotation in this database.

literature. The list is not intended to be exhaustive, and more complete investigations are available in [19,28].

We first introduce the methods based on the variability of RR intervals (RRI) [10,11,13,21-23,25].

Kikillus, *et al* [10] conducted a Markov modeling (MM) technique to identify AF. The calculated test results of *Se* and *Sp* were 94.1% (+2.79%, values in parentheses are the differences between our results and the reported results, hereinafter the same) and 93.4% (+4.87%) for the AFDB+NSRDB database.

The method introduced by Dash, *et al* [11], relies on the combination of the root mean square of successive differences (RMSSD), the turning points ratio (TPR) and SE. The presence of AF using this method was considered if given conditions based on thresholds were satisfied. For the AFDB[‡] database, the calculated *Se* and *Sp* values were 94.4% (+3.43%) and 95.1% (+3.09%), respectively; and 90.2% (+7.13%) and 91.2% (-0.42%) for the MITDB set, respectively. When compared to our method with respect to the MITDB set, the *Sp* is slightly better than our method, however, there is an unacceptably lower rate of AF identification *Se*.

Tatento, *et al* [13] presented a novel technique using the Kolmogorov-Smirnov test. By choosing the AFDB data set for evaluation, the calculated *Se*, *Sp* and *PPV* values were 94.4% (+2.49%), 97.2% (+1.05%) and 96.0% (+1.62%), respectively. Other researchers' re-investigated corresponding values were 91.20% (+5.69%), 96.08% (+2.17%) and 90.32% (+7.30%) [19], respectively.

Lian, *et al* [21] developed an AF detector with its basis centered on the Map of RR intervals versus change of RR intervals (RdR). For the AFDB and MITDB sets, the *Se* and *Sp* values were 95.8% (+1.09%) and 96.4% (+1.85%), 98.9% (-1.57%) and 78.8% (+11.98%), respectively. The calculated *Sp* for the NSRDB database was 90.0% (+8.28%) . By comparison, when tested on the MITDB set, the *Se* is slight higher than that of our new method; there is, however, a markedly lower rate of non-AF detection *Sp*.

An attractive approach to AF detection was initiated by Huang, *et al* [23]. It utilized a histogram of ΔRR_n and standard deviation (SD) analysis. The calculated *Se* and *Sp* were 96.1% (+0.79%) and 98.1% (+0.15%), when the AFDB set was assessed. The calculated *Sp* was 97.9% (+0.38%) for the NSRDB database. It provided the closest performance to that of this newly proposed method, as can be seen in Tables 2 and 3.

Table 3 Overview of published results of the existing methods using the same databases

Method	Features	Year	Database	Key techniques	Results			
					Se(%)	Sp(%)	PPV(%)	ACC(%)
Lee, *et al* [25][*]	RRI	2013	AFDB[‡]+NSRDB	Sample entropy	97.26	95.91	–	96.14
Huang, *et al* [23]	RRI	2011	AFDB	Histogram+SD analysis+...	96.1	98.1	–	–
			NSRDB		NA	97.9	NA	NA
Lake, *et al* [22]	RRI	2011	AFDB	COSEn	91	94	–	–
Lian, *et al* [21][*]	RRI	2011	AFDB	Map of RdR	95.8	96.4	–	–
			MITDB		98.9	78.8	–	–
			NSRDB		NA	90.0	NA	NA
Parvaresh, *et al* [20][*]	AR	2011	AFDB[†]	LDA classifier	96.14	93.20	90.09	–
Babaeizadeh, *et al* [16]	RRI/AA	2011[*]	AFDB[†]	Markov	87.27[*]	95.47[*]	92.75[*]	–
	(FSA)	2009			92	–	97	–
Couceiro, *et al* [15]	RRI/AA	2011[*]	AFDB[†]	Neural network classifier	96.58[*]	82.66[*]	78.76[*]	–
	(PWA/FSA)	2008			93.8	96.09	–	–
Schmidt, *et al* [14]	RRI/AA	2011[*]	AFDB[†]	Markov+Templete matching+...	89.20[*]	94.58[*]	91.62[*]	–
	(PWA/FSA)	2008						
Tatento, *et al* [13][*]	RRI	2011[*]	AFDB	Kolmogorov-Smirnov test	91.20[*]	96.08[*]	90.32[*]	–
		2001			94.4	97.2	96.0	–
Slocum, *et al* [12]	AA	2011[*]	AFDB[†]	Power percentage	62.80[*]	77.46[*]	64.90[*]	–
	(PWA/FSA)	1992						
Dash, *et al* [11]	RRI	2009	AFDB[‡]	RMSSD+TPR+SE	94.4	95.1	–	–
			MITDB		90.2	91.2	–	–
Kikillus, *et al* [10][*]	RRI	2007	AFDB+NSRDB	Histogram+DIFF.+pNN200	94.1	93.4	–	–

[*]The authors proposed several methods, in which, the method with the best performance is presented here.
[†]Records "00735" and "03665" omitted.
[‡]Records "04936" and "05091" omitted.
★Reinvestigated in [19].
'–' indicates without report. 'NA' indicates not applicable because there is no beat with AF reference annotation in this database. See text or relevant literature for abbreviation.

Lee, *et al* [25] investigated three statistical techniques to determine the presence of AF, and the best performance achieved when Sample entropy was employed. Using the AFDB[‡]+NSRDB data set, the calculated *Se, Sp* and *ACC* were 97.26% (+0.27%), 95.91% (+2.35%) and 96.14% (+2.02%), respectively.

Parvaresh, *et al* [20] evaluated three classifiers for AF screening by using autoregressive modeling (AR). Within this method, AR coefficients of 15-second segments of ECGs were taken as features. When tested with the AFDB[†] set, the best performance occurred at the so-called LDA classifier: the calculated *Se, Sp* and *PPV* were 96.14% (+0.68%), 93.20% (+4.86%) and 90.09% (+7.52%), respectively.

Slocum, *et al* [12] published a method based on the reference of AA. The frequency spectrum analysis (FSA) of the remainder generated by canceling the ventricular activity from the surface ECG was applied for differentiating rhythms. Due to the lack of a constant phase relationship between the atrial and ventricular activities, the performance of this type of technique is not high. The AF detection method based only on AA showed inferior performance as can be clearly seen from the Table 3: evaluated on the AFDB[†] set, the calculated *Se, Sp* and *PPV* values were 62.80% (+34.02%), 77.46% (+20.60%) and 64.90% (+32.71%), respectively [19].

Other methods that take advantage of multiple character traits (i.e., RRI/AA and FSA/PWA) have also been developed [14-16], and were re-investigated in [19]. Recent data consistently indicates these techniques have relatively lower performance, as can be seen in Table 3. The accuracy of multi-feature (or only AA feature) based techniques have been limited by practical challenges encountered in the reliable determination of AA (and/or PWA). Currently, a common rule of thumb is that, as a whole, the sole RRI based techniques are likely to yield better results than those rely on making inferences from multi-feature (or only AA feature) of the surface ECG, since the R-wave peak is the most prominent characteristic trait of an ECG recording and the least susceptible to various kinds of noise.

It can be frequently difficult to determine a perfect discrimination threshold for AF episode classification, and it is therefore worth performing further analysis to determine whether the varying discrimination threshold settings significantly influence the performance of our method. For the testing databases, the performance of our method is investigated at various threshold settings. Discrimination threshold values from 0.20 to 0.50 in increments of 0.001 are tested for each of the data sets. Plots of the corresponding results can be seen in Figure 7, where we clearly see that our new method is, preferably performed with the threshold ranging from 0.30 to 0.36. It is sufficient to select a random threshold in this range to investigate the performance of this method. Therefore, the calculated best performing threshold value (i.e., 0.353), derived from the ROC curve of the training set (i.e., LTAFDB database), is appropriate for performance evaluation.

In summary, the results of this study demonstrate that the combination of nonlinear/ linear integer filters, symbolic dynamics and SE yields a robust detector. This new detector exhibited a higher detection rate than previous methods. This could possibly lead to incorporation into computerized ECG interpretation systems to improve the reliability of arrhythmia classification.

A special issue on computational complexity

The computational time of our method is also investigated. Our technique is implemented using the C++ programming language. Table 4 displays the computation time
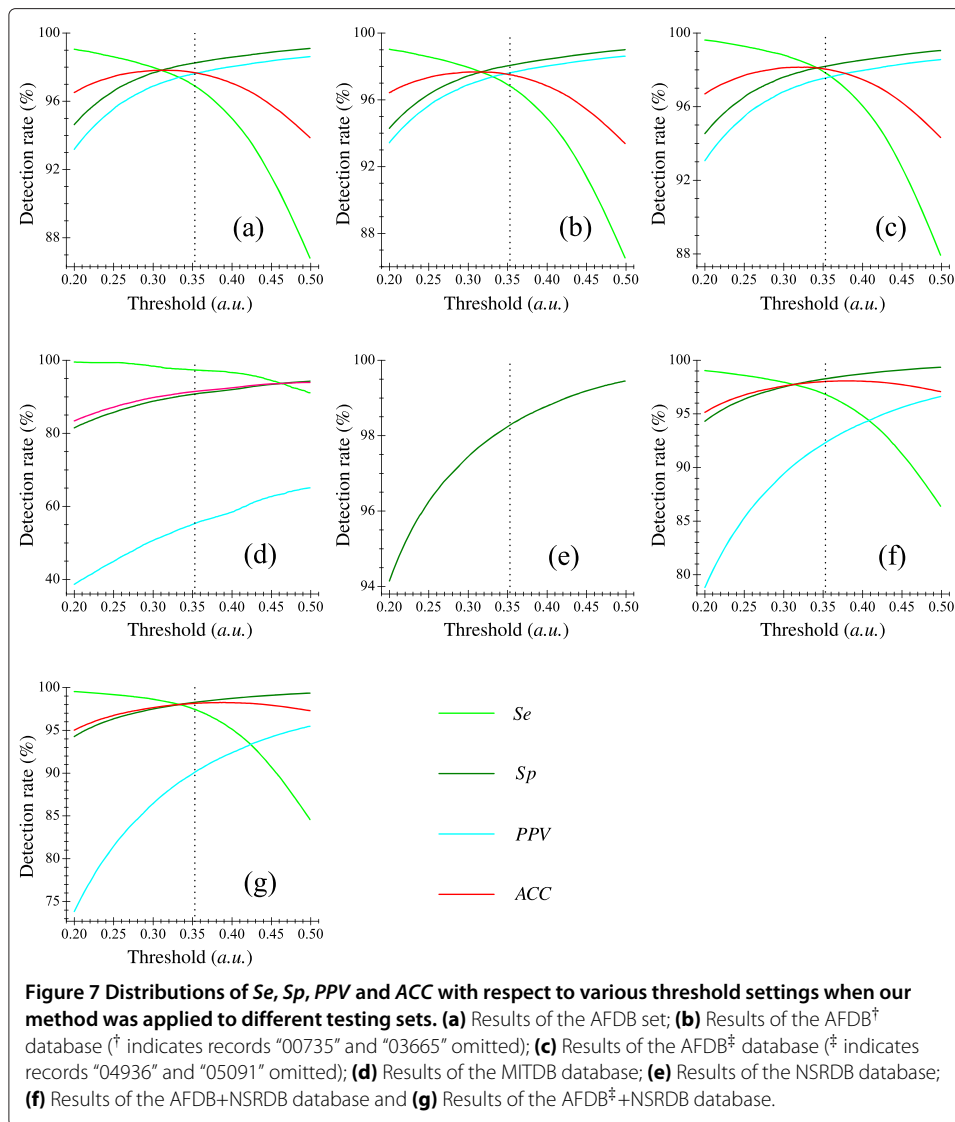
Figure 7 Distributions of *Se*, *Sp*, *PPV* and *ACC* with respect to various threshold settings when our method was applied to different testing sets. (a) Results of the AFDB set; **(b)** Results of the AFDB[†] database ([†] indicates records "00735" and "03665" omitted); **(c)** Results of the AFDB[‡] database ([‡] indicates records "04936" and "05091" omitted); **(d)** Results of the MITDB database; **(e)** Results of the NSRDB database; **(f)** Results of the AFDB+NSRDB database and **(g)** Results of the AFDB[‡]+NSRDB database.

Table 4 The computation time of the processing of this method

Databases	Signal duration (sec)	Computation time (sec)[§]
LTAFDB	6970560 (1936.27 hours)	11.09
AFDB	917052.96 (254.74 hours)	1.445
AFDB[†]	843688.72 (234.36 hours)	1.353
AFDB[‡]	843688.72 (234.36 hours)	1.406
MITDB	86666.67 (24.07 hours)	0.116
NSRDB	1574976 (437.49 hours)	1.825
AFDB+NSRDB	2492028.96 (692.23 hours)	3.258

[†]Records "00735" and "03665" omitted.
[‡]Records "04936" and "05091" omitted.
[§]Desktop test environment: (a) hardware: Intel Pentium(R) Dual-Core E5800(3.20GHz)/DDR3 RAM (2GBytes,800MHz)/ HDD(7200rpm); (b) software: WINDOWS XP Professional/mingw32-g++/C++. The computation times are the average values of 100 trials, and they include the time consumption for importing raw data from the HDD into the RAM.

taken by our method while testing with the available databases. Detailed information of the Desktop test environment can be seen in the footnote of Table 4. The computation time of our method is significantly less than the total duration of all records in each database, which indicates that the time consumption is negligible: typically about 0.116 seconds per 24 hours of data processed. Larburu, *et al* [19,28] investigated a variety of existing methods processed on a computer server and they concluded that the method proposed by Cerutti, *et al* [29] had the lowest computation time of approximately 0.36 seconds per 1 hour of data processed (using the AFDB[†] set, the relevant *Se*, *Sp* and *PPV* were 96.10% (+0.72%), 81.55% (+16.51%) and 75.76% (+21.85%) [19], see [19,28,29] for details). This implies that our method is especially suitable in real-time, long-term ECG monitoring. In addition, Big data is coming of age; our newly developed method shows promise to be of practical use.

Benefits and limitations

In this study, we use a discrimination threshold of 0.353 for AF classification. Of note, from Figure 7, increasing in threshold value improves *Sp* but decreases *Se*. By contrast, the decreasing in threshold values improves *Se* but decreases *Sp*. A compromising solution is thus necessary, and this makes it easy for one to apply specific threshold settings to the concrete application. In spite of this, comparing the latest detection methods when testing with each database, we confirm that a discrimination threshold of 0.353 is adequate to permit better performance of this new method under various situations.

It is commonly asserted and accepted that there is a great deal of time-consuming routines involved in the assessment of AF due to the statistical analysis of irregular/chaotic arrhythmia characteristics. Dramatic benefits can be achieved with the implementation of this AF detector through properly designed recursive algorithms as well as a novel pre-defined set $-\frac{1}{\log_2 N} p_i \log_2 p_i$ for the calculation of $\mathcal{H}''(\mathbf{A})$, which may markedly reduce computational complexity.

The bin size N was set to 127 in this study because a small quantity of words inside a small bin ($\ll N$), in general, might indeed reduce the accuracy of estimating the word wv_n probability distribution [30]. However, for sporadic AF episodes of relatively short duration (e.g., ten seconds), it might incur false negative detection, and this may be a potential limitation. In this regard, it is an inherent technical difficulty that needs to be overcome in the future, though AF episodes of very short duration are rare in practice. Nevertheless, it is essential to remember this limitation.

Once again, as stated in the previous section, a small *PPV* calculated from the MITDB database implies that this newly proposed approach needs to be further refined towards a universally applicable method.

Conclusions

As currently available techiniques are only modestly effective in AF episode screening, we developed a fully automated detection method which aims to fulfill two essential needs: (*i*) earlier real-time identification of AF, and (*ii*) higher reliability of detection. Therefore, with a method available elsewhere for real-time R-wave detection [31], this newly proposed method could be used in intensive care units. The online realization is easy to implement and is computationally attractive as it consists of only several basic operations such as integer addition/subtraction, integer left/right -shifting, integer comparison,

and multiplication and division lying only within $\frac{k}{127000000}$·, as well as 1 floating-point comparison between $\mathcal{H}''(\mathbf{A})$ and the threshold for the rhythm classification. Several state-of-the-art methods have been briefly reviewed, along with their methodologies and detection accuracy. Our new method is evaluated and compared with these existing methods using the LTAFDB, AFDB, NSRDB, and MITDB databases under various situations. We have also presented explicit tables for quantitative assessment of the performance and computation times. Collectively, our results suggest that this AF detector outperforms the existing methods with respect to the performance metrics *Se*, *Sp*, *PPV* and *ACC*. It is also worth emphasizing that a few reference annotations of these data sets are themselves imprecise, just as in the AFDB set. Therefore, extensive sets of exact reference annotations are still needed for investigation.

Appendix

Please visit the "https://onedrive.live.com/?gologin=1&mkt=zh-CN#cid=498A9A3CCEE 3B366&id=498A9A3CCEE3B366%21132" for the compiled C++ dynamic link library files or contact the author for them.

Author details
[1]Institute of Biomedical and Health Engineering, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Xili Nanshan, Shenzhen 518055, China. [2]The Key Laboratory for Health Informatics of the Chinese Academy of Sciences at Shenzhen Institutes of Advanced Technology, Shenzhen 518055, China. [3]Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong.

References
1. Mathew S, Patel J, Joseph S: **Atrial fibrillation: mechanistic insights and treatment options.** *Eur J Intern Med* 2009, **20**(7):672–681.
2. Fuster V, Rydén L, Asinger R, Cannom D, Crijns H, Frye R, Halperin J, Neal Kay G, Klein W, Lévy S, Mcnamara R, Prystowsky E, Samuel Wann L, George Wyse D, Gibbons R, Antman E, Alpert J, Faxon D, Gregoratos G, Hiratzka L, Jacobs A, Russell R, Smith S, Alonso-Garcia A, Blomström-Lundqvist C, De Backer G, Flather M, Hradec J, Oto A, Parkhomenko A, et al.: **ACC/AHA/ESC guidelines for the management of patients with atrial fibrillation: executive summary: a report of the American College of Cardiology/American Heart Association Task Force on Practice Guidelines and the European Society of Cardiology Committee for Practice Guidelines and Policy Conferences (Committee to Develop Guidelines for the Management of Patients With Atrial fibrillation) developed in collaboration with the North American Society of Pacing and Electrophysiology.** *J Am Coll Cardiol* 2001, **38**(4):1231–1265.
3. Petrutiu S, Ng J, Nijm G, Al-Angari H, Swiryn S, Sahakian A: **Atrial fibrillation and waveform characterization.** *IEEE Eng Med Biol Mag* 2006, **25**(6):24–30.
4. Hylek E, Go A, Chang Y, Jensvold N, Henault L, Selby J, Singer D: **Effect of intensity of oral anticoagulation on stroke severity and mortality in atrial fibrillation.** *N Engl J Med* 2003, **349**(11):1019–1026.
5. Miyasaka Y, Barnes M, Gersh B, Cha S, Bailey K, Abhayaratna W, Seward J, Tsang T: **Secular trends in incidence of atrial fibrillation in Olmsted County, Minnesota, 1980 to 2000, and implications on the projections for future prevalence.** *Circulation* 2006, **114**(2):119–125.
6. Gami A, Pressman G, Caples S, Kanagala R, Gard J, Davison D, Malouf J, Ammash N, Friedman P, Somers V: **Association of atrial fibrillation and obstructive sleep apnea.** *Circulation* 2004, **110**(4):364–367.

7. Mukamal K, Tolstrup J, Friberg J, Jensen G, Grønbæk M: **Alcohol consumption and risk of atrial fibrillation in men and women the Copenhagen City Heart Study.** *Circulation* 2005, **112**(12):1736–1742.
8. Moody G, Mark R: **A new method for detecting atrial fibrillation using RR intervals.** In *Computers in Cardiology 1983, Volume 10*. Aachen: IEEE Computer Society Press; 1983:227–230.
9. Artis S, Mark R, Moody G: **Detection of atrial fibrillation using artificial neural networks.** In *Computers in Cardiology 1991*. Venice: IEEE Computer Society Press; 1991:173–176.
10. Kikillus N, Hammer G, Lentz N, Stockwald F, Bolz A: **Three different algorithms for identifying patients suffering from atrial fibrillation during atrial fibrillation free phases of the ECG.** In *Computers in Cardiology 2007*. Durham: IEEE; 2007:801–804.
11. Dash S, Chon K, Lu S, Raeder E: **Automatic real time detection of atrial fibrillation.** *Ann Biomed Eng* 2009, **37**(9):1701–1709.
12. Slocum J, Sahakian A, Swiryn S: **Diagnosis of atrial fibrillation from surface electrocardiograms based on computer-detected atrial activity.** *J Electrocardiol* 1992, **25:**1–8.
13. Tateno K, Glass L: **Automatic detection of atrial fibrillation using the coefficient of variation and density histograms of RR and $\triangle$RR intervals.** *Med Biol Eng Comput* 2001, **39**(6):664–671.
14. Schmidt R, Harris M, Novac D, Perkhun M: **Atrial fibrillation detection.** [WO Patent 2,008,007,236. Jan 18, 2008].
15. Couceiro R, Carvalho P, Henriques J, Antunes M, Harris M, Habetha J: **Detection of atrial fibrillation using model-based ECG analysis.** In *19th International Conference on Pattern Recognition, 2008*. Tampa: IEEE; 2008:1–5.
16. Babaeizadeh S, Gregg R, Helfenbein E, Lindauer J, Zhou S: **Improvements in atrial fibrillation detection for real-time monitoring.** *J Electrocardiol* 2009, **42**(6):522–526.
17. Park J, Lee S, Jeon M: **Atrial fibrillation detection by heart rate variability in Poincare plot.** *Biomed Eng Online* 2009, **8**(38):1–12.
18. Yaghouby F, Ayatollahi A, Bahramali R, Yaghouby M, Alavi A: **Towards automatic detection of atrial fibrillation: A hybrid computational approach.** *Comput Biol Med* 2010, **40**(11):919–930.
19. Larburu N, Lopetegi T, Romero I: **Comparative study of algorithms for Atrial Fibrillation detection.** In *Computing in Cardiology 2011*. Hangzhou: IEEE; 2011:265–268.
20. Parvaresh S, Ayatollahi A: **Automatic atrial fibrillation detection using autoregressive modeling.** In *2011 International Conference on Biomedical Engineering and Technology*. Kuala Lumpur: APCBEES; 2011:4–5.
21. Lian J, Wang L, Muessig D: **A simple method to detect atrial fibrillation using RR intervals.** *Am J Cardiol* 2011, **107**(10):1494–1497.
22. Lake D, Moorman J: **Accurate estimation of entropy in very short physiological time series: the problem of atrial fibrillation detection in implanted ventricular devices.** *Am J Physiol Heart Circ Physiol* 2011, **300:**H319–H325.
23. Huang C, Ye S, Chen H, Li D, He F, Tu Y: **A novel method for detection of the transition between atrial fibrillation and sinus rhythm.** *IEEE Trans Biomed Eng* 2011, **58**(4):1113–1119.
24. Yaghouby F, Ayatollahi A, Bahramali R, Yaghouby M: **Robust genetic programming-based detection of atrial fibrillation using RR intervals.** *Expert Syst* 2012, **29**(2):183–199.
25. Lee J, Reyes B, McManus D, Mathias O, Chon K: **Atrial fibrillation detection using an Iphone 4S.** *IEEE Trans Biomed Eng* 2013, **60:**203–206.
26. Shannon C: **A mathematical theory of communication.** *Bell Syst Tech J* 1948, **27**(3):379–423.
27. **The Physionet ECG database.** [Online]. Available: http://physionet.org/physiobank/database/
28. Larburu N: **Comparative study of algorithms for atrial fibrillation detection.** *Master's thesis.* Public University of Navarra, Pamplona, Spain, 2011, [Online]. Available: http://academica-e.unavarra.es/handle/2454/4136
29. Cerutti S, Mainardi L, Porta A, Bianchi A: **Analysis of the dynamics of RR interval series for the detection of atrial fibrillation episodes.** In *Computers in Cardiology 1997*. Lund: IEEE Computer Society Press; 1997:77–80.
30. Voss A, Kurths J, Kleiner H, Witt A, Wessel N, Saparin P, Osterziel K, Schurath R, Dietz R: **The application of methods of non-linear dynamics for the improved and predictive recognition of patients threatened by sudden cardiac death.** *Cardiovasc Res* 1996, **31**(3):419–433.
31. Pan J, Tompkins W: **A real-time QRS detection algorithm.** *IEEE Trans Biomed Eng* 1985, **32**(3):230–236.