International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:6, 2009

A Combinatorial Model for ECG Interpretation

Costas S. Iliopoulos*, Spiros Michalakopoulos**

Abstract—A new, combinatorial model for analyzing and interpreting an electrocardiogram (ECG) is presented. An application of the model is $\mathcal{QRS}$ peak detection. This is demonstrated with an online algorithm, which is shown to be space as well as time efficient. Experimental results on the *MIT-BIH Arrhythmia* database show that this novel approach is promising. Further uses for this approach are discussed, such as taking advantage of its small memory requirements and interpreting large amounts of pre-recorded ECG data.

Keywords—Combinatorics, ECG analysis, MIT-BIH Arrhythmia Database, QRS Detection, String Algorithms

I. Introduction

A. ECG Data

An electrocardiogram (ECG) is obtained by placing electrodes on the skin and measuring the direction of electrical current discharged by the heart. The current is plotted into waveforms and displayed as in Figure 1.

A *lead* provides a view of the heart's electrical activity between one positive and one negative pole [1]–[4]. Most standard ECG recordings are obtained using a 12-lead device, in clinical settings, and a 2-lead device, in Holter (ambulatory) monitors. The sample in Figure 1 is from a 2-lead reading, as are all 48 records in the MIT-BIH (Massachusetts Institute of Technology - Beth Israel Hospital) Arrhythmia database [5] that was used for development and testing purposes. The figure shows readings from *Modified Limb lead II* and *Precordial lead* V_1.

The different leads provide alternate views of the heart and often a combination of many leads are required to perform a diagnoses; other times, one or two are enough. In each case, the leads that give the most information are preferred. For example, virtually all the information needed to assess *atrial enlargement* [4], can be found in leads *II* and V_1.

B. ECGs and Algorithms

There are many algorithms [6]–[8], and software that interpret electrocardiograms. A comparative study [9] identified a number of algorithmic approaches used, such as neural networks [10], digital filters/wavelet transform [11], [12] and syntactic algorithms [13], [14]. Some of these algorithms have been implemented and tested with remarkable results, most notably [11].

The novel approach taken in this paper has similarities to the syntactic methods. Using a simple alphabet the presented algorithm reads the signals and from the differences between consecutive electrical potentials builds sequences of characters, as shown in Figure 2.

String algorithms have proven to be efficient and useful for solving many molecular biology related problems [15], [16]. Pattern matching techniques used could have similar applications in biomedicine, for example for efficient algorithms on *electroencephalographic* (EEG) data, as for ECGs.

In this paper, the model's practicality is illustrated on an implementation of automated $\mathcal{QRS}$ detection. Tests were performed on all the records in the MIT-BIH Arrhythmia database and the results reaffirmed the belief that the model is worth investigating further. Possible applications of the method include more complex ECG analysis tasks, such as automatic heartbeat classification for which accurate algorithms are still required [17].

The rest of the paper is organized as follows. Next, in Section II, some definitions are presented as they relate to the problem and the algorithm, both of which are described in Section III. The implementation and experimental results are discussed in Section IV. Finally, Section V describes other applications of the model, extensions to the algorithm and the conclusions.

II. Definitions

A *signal* s is a k-tuple $(t, p_{MLII}, p_{V1}, ...)$, where t is the time in seconds and p_E is the electrical potential at lead E in millivolts. When the signal read is from a single lead, it is represented as a pair, a 2-tuple, (t, p).

A *sequence of signals* $s_1, s_2, \ldots, s_n$ is a sequence of pairs, (t_1, p_1), (t_2, p_2), ..., (t_n, p_n). The readings are taken at regular time *intervals*, the *sample rate* σ. Note that $\sigma = t_{i+1} - t_i$ = $t_i - t_{i-1}$, for all $i \in [2, n-1]$. For example, in the MIT-BIH Arrhythmia database, 360 samples are taken per second, i.e. the sample rate $\sigma = 1000/360$ ms.

Let $\hat{\Sigma}$ and Σ the following alphabets:

$$\hat{\Sigma} = \{--, -, 0, +, ++\}$$

$$\Sigma = \{C^{--}, C^{-}, C^{0}, C^{+}, C^{++}\}$$

and $\mathcal{A}$ the set of non-trivial pairs:

$$\mathcal{A} = \{(+,0), (-,0), (++,+), (--,-)\}$$

Note that trivial pairs are $(0,0), (+,+), ...$

The *rate of change* is the difference between the potentials of two consecutive signals. Thus, at position i, the rate of change is $p_i - p_{i-1}$. A gradual rate *increase* is depicted as C^{+}, a sharp rate increase as C^{++} and gradual and sharp *decreases* equivalently as C^{-} and C^{--}. A negligible or zero rate change is C^{0}.

The authors are with the Algorithm Design Group, Department of Computer Science, King's College London, Strand, London WC2R 2LS, England. Web: http://www.dcs.kcl.ac.uk/adg, email: {csi,spiros}@dcs.kcl.ac.uk
* Digital Ecosystems & Business Intelligence Institute, Curtin University, Perth, Australia. ** Supported by the Engineering and Physical Sciences Research Council (EPSRC) under the Doctoral Training Award (DTA).

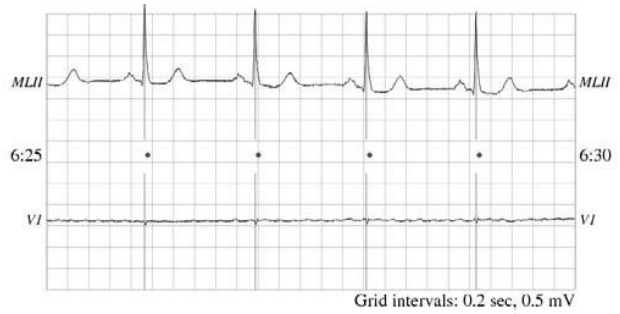

Fig. 1: Short Sample of ECG for Record 101 from MIT-BIH Arrhythmia Database. The readings are from leads *MLII* and V_1.

Two parameters, a *low threshold* μ_E and a *high threshold* μ'_E are used to filter the rates of change for readings from lead E. For single lead readings, the thresholds are denoted as μ and μ'. For example, in the MIT-BIH Arrhythmia database, it has heuristically been determined that reasonable values for many records are $\mu_{MLII} = 0.01$ and $\mu'_{MLII} = 0.05$.

The *direction* of the rate of change is a character d in Σ. To determine d_i, Equation 1 is used:

$$d_i = \begin{cases} C^0, & \text{if } |p_{i+1} - p_i| \leq \mu \\ C^-, & \text{if } -\mu' < p_{i+1} - p_i < -\mu \\ C^+, & \text{if } \mu < p_{i+1} - p_i < \mu' \\ C^{--}, & \text{if } p_{i+1} - p_i \leq -\mu' \\ C^{++}, & \text{if } p_{i+1} - p_i \geq \mu' \end{cases} \quad (1)$$

Two consecutive rate changes have *direction equivalence*, when there's a smooth transition from one rate change to the other, without a significant change of direction.

A consecutive sequence of direction equivalent rate changes is defined as $\mathcal{C}[x,k]$ of C^x, where $x \in \hat{\Sigma}$, $k \in \mathbb{N}^+$ and for some $C^x \in \Sigma$. A *direction equivalent series* $\mathcal{C}[x,k]$, is referred to simply as a *series* when it is implicit from the context.

Formally, C^x *is equivalent to* C^y,

$$C^x \sim C^y \quad (2)$$

if $(x,y) \in \mathcal{A}$. Furthermore, the two sequences are equivalent

$$C^{x_1}C^{x_2}...C^{x_k} \approx C^{y_1}C^{y_2}...C^{y_k} \quad (3)$$

if $C^{x_i} \sim C^{y_i}$, $\forall i$.

For example, $C^+ \sim C^0$, but $C^- \nsim C^{++}$ (*not equivalent to*) and the sequence $C^0C^+C^+C^0 \approx \mathcal{C}[+,4]$, but $C^-C^{--}C^{--}C^0C^- \not\approx \mathcal{C}[-,5]$.

Also note, $\mathcal{C}[+,k] \approx C^{x_1}C^{x_2}...C^{x_k}$, where $x_i \in \{+,0\}$, for $i \in [1..k]$, $\mathcal{C}[++,k] \approx C^{x_1}C^{x_2}...C^{x_k}$ where $x_i \in \{++,+\}$ for $i \in [1..k]$ and similarly for $\mathcal{C}[-,k]$ and $\mathcal{C}[--,k]$.

The time t that the peak of the $\mathcal{R}$ wave is encountered (and by extension the $\mathcal{QRS}$ complex) is stored in vector $v_{\mathcal{R}}$. Thus, the time of the occurrence of the i'th $\mathcal{R}$ wave is denoted as $v_{\mathcal{R}_i}$. The time difference between two $\mathcal{R}$ waves, the $\mathcal{RR}$ *interval* can be easily calculated, $v_{\mathcal{R}_i} - v_{\mathcal{R}_{i-1}}$.

III. The Algorithm

The first requirement for an ECG analysis algorithm, as for an ambulatory monitoring system is to be able to analyze the heart rate and rhythm disturbances [18]. Normally the first step towards this is to detect the $\mathcal{QRS}$ complexes.

A. $\mathcal{QRS}$ Detection Problem

Problem 1 ($\mathcal{QRS}$ Detection): Given a subject's electrocardiogram as a sequence of signals s_1, s_2 ..., $s_n = (t_1, p_1)$, (t_2, p_2), ..., (t_n, p_n), where t_i, $p_i \in \mathbb{R}$, detect the $\mathcal{QRS}$ complexes and the $\mathcal{RR}$ intervals.

The output of an algorithmic solution to this problem should be a list of times t_i, in which either an $\mathcal{R}$ peak is encountered, as in the MIT-BIH Arrhythmia database, or the beginning of a $\mathcal{Q}$ peak starts, as in the AHA (American Heart Association) database [19]. Both can be said to correctly "detect the $\mathcal{QRS}$ complexes". The $\mathcal{RR}$ intervals can be inferred from this list by a simple subtraction, $t_j - t_{j-1}$, of two consecutive elements.

B. Parameters and Conditions

Let pattern $\pi = \mathcal{C}[x_1,k_1]\mathcal{C}[x_2,k_2]...\mathcal{C}[x_\ell,k_\ell]$ be a direction equivalent series. Problem 1 is solved by searching online for pattern π in text $d_1d_2\ldots d_{n-1}$, where d_i is determined using Equation 1. To identify the $\mathcal{R}$ peaks, ℓ is set to 2.

An average persons heart rate is between 60 and 100 beats per min but endurance athletes for example can have a heart rate well below 60 beats per min when in a state of rest [4], [18]. Hence, for each person, what is a 'normal' beat will depend on their physiology, the general condition of their health and the current state of their heart. Thus, the specific characteristics of the patient need be taken into account.

Parameters rr_min and rr_max are defined to be the minimum and maximum values between consecutive beats ($\mathcal{RR}$ Interval) that are considered normal. Similarly, pot_E_min and pot_E_max (pot_min and pot_max for the single lead) are the minimum and maximum potentials for lead E at the detection of an $\mathcal{R}$ wave.

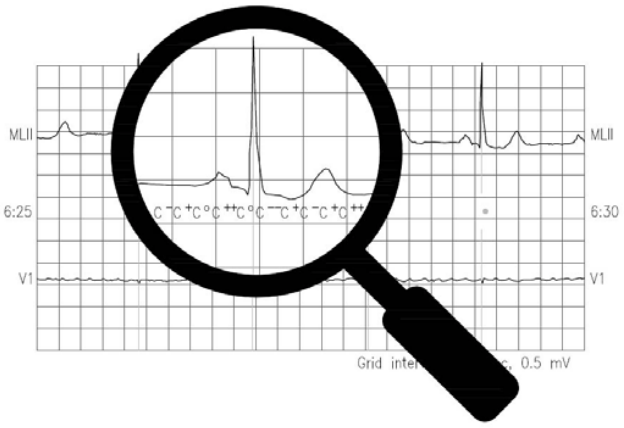

Fig. 2: The differences between consecutive ECG signals define letters from alphabet $\Sigma = \{C^0, C^+, C^-, C^{++}, C^{--}\}$

C. Algorithm Outline and Pseudo-code

An outline of the algorithm:

STEP 1

During the learning period determine values μ, μ', rr_max, rr_min, pot_min, pot_max, k_1 and k_2 for the subject.

STEP 2

Determine d_i by difference in potential from last signal, as in Equation 1.

STEP 3

Either add d_i to current series $\mathcal{C}[x,k]$ or start a new series if certain conditions met. If added to current series and identifies a normal beat, then store in $v_{\mathcal{R}}$.

STEP 4

Repeat steps 2 and 3 until end of electrocardiogram.

In step 3 it is stated that a new series is started if certain conditions are met. The next paragraphs describe what these conditions are and how they are met.

The algorithm holds 5 counters, one for each character in Σ, i.e. $+_{count}$, $-_{count}$, 0_{count}, $--_{count}$ and $++_{count}$. The appropriate counter is incremented when the current series is $\mathcal{C}[x,k]$ and d_i is not equivalent to x, i.e. if $d_i = C^y \nsim C^x$, the y counter (y_{count}) is incremented. All the counters are zeroed when a new series is initiated.

To start a new series, d_i's counter must be above a certain number ν. This means that the current series $\mathcal{C}[x,k]$ is only terminated after ν characters $y_1, y_2, ..., y_\nu$, such that $y_1 = y_2 = ... = y_\nu$ and $y_i \nsim x$ are read. This implies that whichever non-equivalent character y's counter, y_{count}, reaches ν first, becomes the new x.

Finally, again in step 3, if $d_i \sim x$ a check must be made whether a normal beat has been identified. In effect, what is checked is whether the time of this beat has occurred within rr_min and rr_max of the last identified beat. Pseudo-code for the algorithm is depicted in Algorithm 1.

Algorithm 1 Process ECG

```
1: function PROCESSECG
2:     C[x,k] ← [0,0]                          ▷ current series
3:     C'[x,k] ← [0,0]                         ▷ previous series
4:     while not end of ECG do
5:         calculate d_i as in Equation 1
6:         if d_i ~ C^x then
7:             increment current series
8:             if normal beat then
9:                 store in v_R
10:            else
11:                store in v_offbeat
12:        else
13:            increment appropriate count: y_count ← y_count + 1
14:            if y_count ≥ ν then              ▷ new series
15:                C'[x,k] ← C[x,k]
16:                C[x,k] ← [y, y_count]
17:                zeroise all counters
```

D. Discussion on the Space Complexity

In the learning phase, the ECG is read and can then be discarded at the end of this initial period. The only storage requirements are for the parameters determined during this stage, i.e. rr_min, rr_max, pot_min, pot_max, μ, μ', k_1 and k_2.

Holter monitors normally record ECG data for a period of 24 or 48 hours. Recent advances in mobile device technology have made it possible for cellular phones and other hand-held devices to be considered for recording ECG data [20], [21]. The recorded data can be compressed [22], [23], and either held on the device or packaged and sent. In the case of cellular phones, the ECG data could be automatically transmitted at regular times, or when certain exceptional program conditions are met. This data is then analyzed and acted upon in a clinical setting.

The above algorithm does not have any additional space requirements to existing algorithms, and all that needs storing are the constants mentioned. Furthermore, a lookup table of patterns that identify irregularities could be stored on the device for detection of conditions that require immediate attention.

IV. Experimental Results

The algorithm was implemented, and tests were carried out on the MIT-BIH Arrhythmia database. The implementation was done on a Windows machine with an Intel Celeron M processor of 1.60GHz and 896MB RAM and the tests were run through Cygwin, a Linux-like environment for Windows. The implementation was done in C++ using the STL and the program design is based on the State design pattern.

In evaluating the performance of the program the only concern is the accuracy of $\mathcal{QRS}$ detection and not the speed of execution. The algorithm is an online algorithm and thus must run in linear time as the data is being recorded via the electrodes. It should be clear from Algorithm 1 that the algorithm's speed complexity is indeed $O(n)$.

Standard statistical terms and formulas [24], were used for the evaluation. A correctly identified beat is a true positive (TP) event, a incorrectly identified beat is a false positive (FP) event and a missed beat is a false negative (FN) event. The two measures of the performance of the algorithm are $\mathcal{QRS}$ sensitivity ($\mathcal{QRS}\ Se$) and positive predictivity ($\mathcal{QRS}\ +P$):

$$\mathcal{QRS}\ Se = \frac{TP}{TP+FN} \qquad \mathcal{QRS}\ +P = \frac{TP}{TP+FP}$$

Thus, $\mathcal{QRS}\ Se$ measures the fraction of events that were identified by the algorithm, and $\mathcal{QRS}\ +P$ measures the fraction of detections that are events.

The program was tested on all records (inc. paced beat) in the database. It produces two text files which collectively contain all the *beat events* identified. One file is the *offbeats* and the other is the *normal* beats. The offbeats file has a list of times that denote occurrences of non-normal beats, such as premature ventricular contraction (PVC), indications of arrhythmia or other cardiac irregularities, noise, or an FP detection on the programs part. The algorithm is unable to distinguish between these just yet.

TABLE I: Record-by-record experimental results on the MIT-BIH Arrhythmia database.

rec	$\mathcal{QRS}$ Se	$\mathcal{QRS}$ $+P$	rec	$\mathcal{QRS}$ Se	$\mathcal{QRS}$ $+P$
100	98.74	99.58	201	67.39	100.0
101	99.74	99.41	202	84.55	99.94
102	99.73	98.11	203	48.77	75.72
103	97.05	90.46	205	96.59	99.67
104	97.74	96.85	207	34.30	89.36
105	95.17	98.37	208	71.48	92.51
106	73.58	86.19	209	70.11	71.64
107	94.34	81.66	210	67.20	99.60
108	50.61	74.09	212	95.01	90.12
109	18.10	96.69	213	39.74	82.73
111	91.55	95.20	214	83.18	98.24
112	99.57	79.71	215	63.04	71.05
113	99.47	87.96	217	92.25	96.59
114	97.94	87.28	219	92.89	75.48
115	98.90	63.94	220	91.32	67.14
116	94.15	48.74	221	85.00	99.13
117	99.69	66.91	222	50.24	97.97
118	83.72	76.42	223	84.67	79.00
119	78.51	47.71	228	69.58	87.07
121	03.72	89.23	230	99.30	66.96
122	82.57	65.97	231	76.84	97.61
123	99.37	65.95	232	97.71	98.37
124	95.03	67.90	233	74.27	94.11
200	61.25	85.29	234	99.13	99.82

Aggregate Statistics					
	Gross Statistics			Average Statistics	
recs	$\mathcal{QRS}$ Se	$\mathcal{QRS}$ $+P$	recs	$\mathcal{QRS}$ Se	$\mathcal{QRS}$ $+P$
all	79.08%	82.11%	all	80.1	84.99

The file with the normal beats is compared to the human annotated file (*rec.atr* for Record *rec*) in the database, using PhysioNet [19] tools. First it is converted into an annotation file using *rr2ann* and then the beat-by-beat comparator *bxb* [25] is run, which compares the two files. The latter gives the $\mathcal{QRS}$ Se and $\mathcal{QRS}$ $+P$ measurements.

Table I shows the performance of the algorithm on all 48 records in the test database. The results are for fixed parameters: $\mu = 0.01$, $\mu' = 0.05$, $k_1 = k_2 = 2$, $rr\_min = 210$ and $pot\_max = -500$. The remaining parameters were determined by the program during the learning phase, i.e. the first 5 minutes of each record.

The *average* and *gross* statistics are shown for the whole database. The average statistics give overall percentages, where each record is given equal weight, whereas for the gross statistics each event (beat) is given equal weight.

Table II shows the results on records grouped by performance percentages. This shows that the algorithm performs very well (above 95%) on 37.5% of records, and poorly (less than 80%) on roughly half the records. The next section discusses, as well as conclusions, improvements that can be made to the algorithm and program.

V. Conclusions and Further Works

A novel approach to ECG interpretation has been investigated in this paper. A combinatorial model was presented and an algorithm to identify the $\mathcal{QRS}$ peaks was discussed. The algorithm was implemented and tested on a standard database. The experimental results were promising and in some cases comparable to more traditional algorithmic approaches, a fact that suggests that further research and development of the method is desirable.

A further use of the model is for prerecorded ECG data. Holter monitors are often used to gather ECG data for an extended period of time, for further analysis in the clinic. Using the presented combinatorial model, this data could be indexed using for example a suffix tree or suffix array [15], [26]. For an ECG of length n, this indexing can be done in $O(n)$ time, given that the alphabet Σ is of constant size ($|\Sigma| = 5$). Patterns of the form $\pi = \mathcal{C}[x_1, k_1]\mathcal{C}[x_2, k_2]...\mathcal{C}[x_\ell, k_\ell]$ can be searched in time relative to the length of the pattern. For example, the pattern used to detect $\mathcal{QRS}$ peaks in the program is $\mathcal{C}[x, 2]\mathcal{C}[x, 2]$, on which the search time is minimal.

Another application of the method on prerecorded data is to build a dictionary of patterns which identify irregularities. An Aho-Corasick automaton [27] could be built for the patterns and a trace against the ECG will identify all their occurrences in linear on the length of the ECG time.

Improvements can be made to the program for the on-line version. More sophisticated statistical analysis algorithms should be employed on the data obtained in the learning period, to give 'better' parameters as input to the test period of the program. These statistical methods could depend on the type of patterns being sought in the ECG and may thus give different parameters as input to the test period, in real world clinical situations.

Classification of the types of beats is potentially a further use of the model worth investigating, as is the identification of tachycardia and bradycardia. More sophisticated patterns and algorithms will need to be developed to use the model to detect arrhythmia and other cardiac anomalies. More difficult to identify waves and intervals will need to be identified in the ECG. Another extension and improvement to the algorithm

TABLE II: Grouped performance of the algorithm on the test database.

$\mathcal{QRS}$ Se			
	Records	#	Percentage
$> 99\%$	101,102,112,113,117,123,230,234	8	16.67%
95%-99%	100,103,104,105,114,115,124, 205,212,232	10	20.83%
90%-95%	107,111,116,217,219,220	6	12.50%
80%-90%	118,122,202,214,221,223	6	12.50%
$< 80\%$	106,108,109,119,121,200,201,203,207, 208,209,210,213,215,222,228,231,233	18	37.50%

$\mathcal{QRS}$ $+P$			
	Records	#	Percentage
$> 99\%$	100,101,201,202,205,210,221,234	8	16.67%
95%-99%	102,104,105,109,111,214,217, 222,231,232	10	20.83%
90%-95%	103,208,212,233	4	8.33%
80%-90%	106,107,113,114,121,200,207,213,228	9	18.75%
$< 80\%$	108,112,115,116,117,118,119,122,123, 124,203,209,215,219,220,223,230	17	35.42%

and program would be to use readings from multiple leads.

REFERENCES

[1] Mary B. Conover. *Understanding Electrocardiography*. Mosby, November 2002.
[2] Springhouse (Editor). *ECG Interpretation Made Incredibly Easy!* Lippincott Williams & Wilkins, 4^{th} edition, 2007.
[3] Emanuel Stein. *Rapid Analysis of Arrhythmias*. Lippincott Williams & Wilkins, 3^{rd} edition, 2000.
[4] Malcolm S. Thaler. *The Only EKG Book You'll Ever Need*. Lippincott Williams & Wilkins, 5^{th} edition, October 2006.
[5] Massachusets Institute of Technology. Mit-bih ecg database. Available: http://ecg.mit.edu/.
[6] G.B. Moody and R.G. Mark. Development and evaluation of a 2-lead ecg analysis program. *Computers in Cardiology*, pages 39–44, 1982.
[7] Yu Hen Hu, S. Palreddy, and W. J. Tompkins. A patient-adaptable ecg beat classifier using a mixture of experts approach. *Biomedical Engineering, IEEE Transactions on*, 44(9):891–900, 1997.
[8] V. X. Afonso, O. Wieben, W. J. Tompkins, T. Q. Nguyen, and Shen Luo. Filter bank-based ecg beat classification. In *Proceedings of the 19th Annual International Conference of the IEEE*, volume 1, pages 97–100. Engineering in Medicine and Biology Society, October 1997.
[9] B. U. Kohler, C. Hennig, and R. Orglmeister. The principles of software qrs detection. *Engineering in Medicine and Biology Magazine*, 21(1):42–57, 2002.
[10] Y. H. Hu, W. J. Tompkins, J. L. Urrusti, and V. X. Afonso. Applications of artificial neural networks for ecg signal detection and classification. *Journal of Electrocardiology*, 26, 1994.
[11] V. X. Afonso, W. J. Tompkins, T. Q. Nguyen, and Shen Luo. Ecg beat detection using filter banks. *Biomedical Engineering, IEEE Transactions on*, 46(2):192–202, Feb. 1999.
[12] Hisashi Inoue and Akio Miyazaki. A noise reduction method for ecg signals using the dyadic wavelet transform (special section of papers selected from itc-cscc'97). *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 81(6):1001–1007, 1998.
[13] J. K. Udupa and I. S. N. Murthy. Syntactic approach to ecg rhythm analysis. *IEEE Trans. Biomed. Eng.*, BME-27(7):370– 375, July 1980.
[14] G. Papakonstantinou, E. Skordalakis, and F. Gritzali. An attribute grammar for qrs detection. *Pattern Recogn.*, 19(4):297–303, 1986.
[15] Dan Gusfield. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.
[16] Pavel A. Pevzner. *Computational Molecular Biology: An Algorithmic Approach*. The MIT Press, August 2000.
[17] Philip De Chazal, Maria O'Dwyer, and Richard B. Reilly. Automatic classification of heartbeats using ecg morphology and heartbeat interval features. *IEEE Transactions on Biomedical Engineering*, 51:1196–1206, 2004.
[18] ANSI/AAMI. *EC38: Ambulatory Electrocardiographs*. Association for the Advancement of Medical Instrumentation, 1998.
[19] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. Ch. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000 (June 13). Circulation Electronic Pages: http://circ.ahajournals.org/cgi/content/full/101/23/e215.
[20] F. Sufi, Q. Fang, and I. Cosic. Ecg r-r peak detection on mobile phones. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 3697–3700, 2007.
[21] X. Chen, C. T. Ho, E. T. Lim, and T. Z. Kyaw. Cellular phone based online ecg processing for ambulatory and continuous detection. *Computers in Cardiology*, pages 653–656, Sept. 2007.
[22] J. A. Crowe, N. M. Gibson, M. S. Woolfson, and M. G. Somekh. Wavelet transform as a potential tool for ecg analysis and compression. *Biomed. Eng.*, 14(3), 1992.
[23] Michael L. Hilton. Wavelet and wavelet packet compression of electrocardiograms. *IEEE Trans. Biomed. Eng*, 44:394–402, 1997.
[24] ANSI/AAMI. *EC57: Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment*. Association for the Advancement of Medical Instrumentation, 1998.
[25] George B. Moody. *WFDB Applications Guide*. Harvard-MIT Division of Health Sciences and Technology, 10^{th} edition, November 2008.
[26] Maxime Crochemore, Christophe Hancart, and Thierry Lecroq. *Algorithms on Strings*. Cambridge University Press, 2007.
[27] Alfred V. Aho and Margaret J. Corasick. Efficient string matching: an aid to bibliographic search. *Commun. ACM*, 18(6):333–340, 1975.