Med Bio Eng Comput (2007) 45:229–239
DOI 10.1007/s11517-006-0118-1

ORIGINAL ARTICLE

Unsupervised classification of ventricular extrasystoles using bounded clustering algorithms and morphology matching

David Cuesta-Frau · Marcelo O. Biagetti · Ricardo A. Quinteiro · Pau Micó-Tormos · Mateo Aboy

Received: 19 June 2006 / Accepted: 1 October 2006 / Published online: 9 November 2006

Abstract Ventricular extrasystoles (VE) are ectopic heartbeats involving irregularities in the heart rhythm. VEs arise in response to impulses generated in some part of the heart different from the sinoatrial node. These are caused by the premature discharge of a ventricular ectopic focus. VEs after myocardial infarction are associated with increased mortality. Screening of VEs is typically a manual and time consuming task that involves analysis of the heartbeat morphology, QRS duration, and variations of the RR intervals using long-term electrocardiograms. We describe a novel algorithm to perform automatic classification of VEs and report the results of our validation study. The proposed algorithm makes use of bounded clustering algorithms, morphology matching, and RR interval length to perform automatic VE classification without prior knowledge of the number of classes and heartbeat features. Additionally, the proposed algorithm does not need a training set.

D. Cuesta-Frau (✉) · P. Micó-Tormos
Technological Institute of Informatics,
Polytechnic University of Valencia,
Campus Alcoi Plaza Ferrándiz y Carbonell, 2,
03801 Alcoi, Spain
e-mail: dcuesta@disca.upv.es

M. O. Biagetti · R. A. Quinteiro
Cardiac Electrophysiology Laboratory,
Department of Physiology, Favaloro University,
Buenos Aires, Argentina

M. Aboy
Department of Electronics Engineering Technology,
Oregon Institute of Technology, Oregon, USA

Keywords Clustering algorithms · Dynamic time warping · Holter recordings · Ventricular extrasystoles · Ventricular premature complexes

1 Introduction

Long-term ECG monitoring is widely used to recognize potentially lethal ventricular arrhythmias. However, manual analysis and inspection of long-term ECG (Holter records) is often an involved and time consuming process. Consequently, automatic ECG processing and analysis algorithms to detect, classify, and cluster heartbeats are important. In this paper we describe an algorithm that can be used to cluster heartbeats and detect ectopic complexs of ventricular origin.

Ectopic heartbeats are an arrhythmia involving variations in an otherwise normal heartbeat. In many cases, they occur without obvious cause and are benign whereas, in other cases, they might be due to more severe abnormalities including coronary artery disease, high blood pressure, diabetes, ischemia, electrolyte imbalance, stess or drug consumption [4].

In the present study we propose an unsupervised method to automatically classify ventricular extrasystoles (VE) in Holter recordings. This method relies on bounded clustering algorithms, morphology matching, and RR interval length. Our method does not require any prior knowledge of the number of classes or the parameters of the heartbeats. A training set is not necessary either.

1.1 Clinical significance

Ventricular extrasystoles or ventricular premature complexes (VPC) are the most common form of ven-

tricular arrhythmias [4]. They originate in the ventricles and can be of potential high risk, depending primarily on their frequency and prematurity. VEs may lead to the appearance of ventricular tachycardia and fibrillation [7]. They are more common in men than in women, and prevalence increases with age. The assessment and treatment of VEs is usually challenging and it involves time consuming tests to evaluate blood, thyroid, or electrolyte levels, an echocardiogram, treadmill testing, cardiac catheterization, electrophysiology testing, or an electrocardiogram [9].

Ventricular extrasystoles can be treated by correcting the underlying abnormality regarding electrolyte imbalances, avoiding stimulants or medications that may cause them, treating the possible heart disorders, using proper medication or specific devices, and, in some cases, applying surgery [9]. Automatic analysis of Holter recordings has the potential to reduce considerably the time required to diagnose and treat VEs.

1.2 Electrocardiogram evaluation

VPCs are infrequent and rarely detected using a short duration single 12-lead ECGs. Long-term monitoring, namely, Holter recordings [10], are better suited to detect VPCs. In an electrocardiogram, VE can be identified according to the following features [4, 7]:

- QRS duration exceeds that of dominant QRS complexes because abnormal ventricular activation takes place via intramyocardial functional pathways.
- Different morphology (Fig. 1). Bizarre QRS complexes are present. No preceding premature P waves occur. Rarely, a sinus wave is conducted. T wave usually is in the opposite direction from the R wave. If beats originate from a single focus, all the VPC have the same morphology, although different from the normal morphology.
- Usually, ventricular extrasystoles are premature, RR intervals are shorter than average RR.

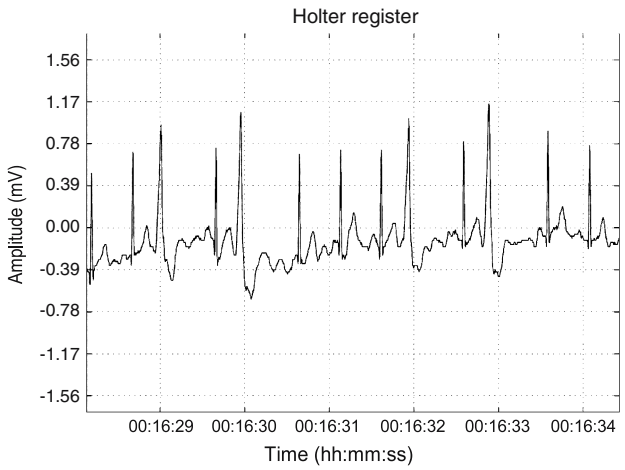

Fig. 1 Example of a real ECG signal illustrating different heartbeat morphologies caused by VE

Automatic algorithms to process and analyze long ECG registers are significant, since manual screening of 24 and 48-h Holter recordings is time consuming and difficult. In this paper we describe a novel unsupervised algorithm to solve this problem. The proposed algorithm makes use of a combination of partitional and hierarchical clustering algorithms with a morphology matching technique based on dynamic time warping (DTW) [2]. Additionally, the algorithm does not require previous annotations or training.

1.3 Overview of hearbeat classification algorithms

In [1], the authors described a method to automatically classify heartbeats using morphology, QRS duration, and RR intervals. This algorithm requires fiducial point detection and a training set for the linear discriminants classifier models.

Dokur and Olmez [5] employed a hybrid neural network for ECG beat classification. They also use a training set and Fourier and Wavelet coefficients determined by dynamic programing as the feature extraction methods.

Engin [6] utilized a fuzzy-hybrid neural network for heartbeat classification using three different feature sets: autoregressive model coefficients, third order cumulant, and the variance of the discrete Wavelet transform using a Daubechies wavelet function. They used a learning set of 800 beats and a testing set of 400 beats with a fixed number of classes of 4, and the number of clusters is 18.

The method presented in [8] classifies heartbeats using a preclassified category prototypes using a genetic algorithm to optimally compute the parameters of the method.

Lagerholm et al. [13] developed a method to cluster ECG complexes into 25 groups using a functional approximation and self-organizing maps.

Finally, Shyu et al. [16], described a method for VPC detection using the Wavelet transform and a fuzzy neural network.

All these algorithms require the user to provide training set. Our proposed algorithm addresses the problem of heartbeat classification and does not require a training set or user-specified parameters. The main characteristics of the algorithm include:

- Unsupervised: It does not require the user to annotate heartbeats in advance, a training set, provide the number of unknown clusters, and the number of unknown morphologies. No a priori knowledge of the number of groups or the number of heartbeats in each class is required either.
- Simple and intuitive feature extraction and feature selection methods: hearbeat morphology and non-uniform sampling.
- Based on VPC basic features: shorter RR intervals, different morphology. A new feature is introduced: Polarity (P). QRS duration is not included to avoid the need to detect these fiducial points.
- Worst case application: single lead and global classifier.
- Temporal cost plays a key role.
- The method is tested using most of the registers from the MIT arrhythmia database [14].

2 Methodology

The general stages of the method are depicted in Fig. 2, and the complete algorithm is in Algorithm 1. Signals are acquired with Holter recorders. Registers are then preprocessed in order to reduce the possible signal interferences. Next a QRS detector is applied and registers are segmented using the R wave location found. For each heartbeat obtained, features are extracted and selected. Finally, a combined clustering method is conducted to find the groups in the heartbeat set. Following, these stages are described in detail.

Algorithm 1 . Unsupervised detection of ventricular extrasystoles

function ClassifyVPC($y[n]$) {Input is the recorded ECG y[n]}
begin
$y_p[n]$:=Preprocess($y[n]$); {Noise reduction}
H :=Segment($y_p[n]$); {QRS detection and beat extraction}
$\widehat{H}$:=TraceSegmentation($H, 120$); {Get heartbeats of length 120 samples}
$\widehat{H}$:=OffsetRemoval($\widehat{H}$); {$\widehat{h_i} = \widehat{h_i} - \text{mean}(\widehat{h_i})$}
C^1 :=GetInitialPartition($\widehat{H}$); {k-means clustering}
C^f :=MergeClusters(C^1); {hierarchical clustering}
ClassifyVPC:= C^f; {Output is a set of clusters}
end

2.1 Preprocessing

Input records, termed $y[n]$, may contain noise, baseline wandering, power line interference and artifacts, and therefore the performance of the method might be affected. In order to diminish the influence of such elements, signals are filtered using a low-pass filter with cut-off frequency of 35 Hz to reduce the noise and power line interference, and then high-pass filtered with a cut-off frequency of 1 Hz to reduce baseline wandering. Although the signal is also partially filtered, this preprocessing does not affect the separability of the two groups, VPC and normal beats. The output of this stage is an enhanced signal $y_p[n]$.

2.2 Segmentation

Segmentation is carried out based on R wave locations. These locations may be obtained using a QRS detector algorithm of good performance, as many reported in the technical literature [12]. For example, we consider [11] a good choice for its accuracy and implementability. From these R wave locations, heartbeats in $y_p[n]$ are segmented, starting at 25% of the previous RR interval length before the R location, and ending at 75% of the current RR interval length after it. These heartbeats h_i make up a set:

$$H = \{h_1, h_2, \ldots, h_N\},$$

where N is the total number of beats, and $H[i] = h_i$. For each heartbeat:

$$h_i[n] = \{x_{i1}, x_{i2}, \ldots, x_{iL_i}\}$$

where $h_i[j] = x_{ij}$ are the heartbeat samples, and L_i is the heartbeat length.

2.3 Normalization

Once the register heartbeats have been segmented, a simple normalization process takes place before fea-

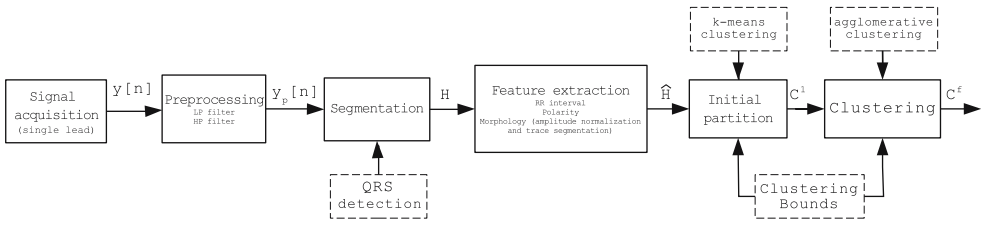

Fig. 2 Block diagram of the method proposed. First discrete time signal $y[n]$ (single lead) is acquired using a Holter recorder. This signal is then filtered and segmented, and the resulting hearbeats are grouped by means of a clustering stage

tures are extracted. This normalization process is an offset subtraction to set the mean value of each heartbeat to 0.

2.4 Feature extraction

Every heartbeat $h_i[n] = \{x_{i1}, x_{i2}, \ldots, x_{iL_i}\}$ carries morphology information in its samples. However, as stated in Sect. 1.2, VPCs can be detected not only by morphology changes, but also by prematurity (shorter RR intervals), changes in T wave direction, and changes in QRS duration. A method to detect VPCs should take into account all these features, but in order not to have to detect the fiducial points to measure QRS duration, we specifically used:

- Morphology given by the amplitude samples.
- Changes in T wave direction. It is measured indirectly by a value termed polarity P, and defined as:

$$P_i = \left|\frac{\max(h_i)}{\min(h_i)}\right|$$

It measures the ratio between the maximum value $\max(h_i)$ and the minimum $\min(h_i)$ of a heartbeat, which is influenced by the directions of R and T waves.

- RR interval duration defined as R_i.

Thus, in order to distinguish between two beats, the morphology, polarity, and RR interval, is employed. The heartbeat representation includes these new two features as $h_i = \{x_{i1}, x_{i2}, \ldots, x_{iL_i}, P_i, R_i\}$. Figure 3 shows the two classes of beats found in MIT register 210 using only polarity and RR interval duration where the good separability provided by these two features can be observed.

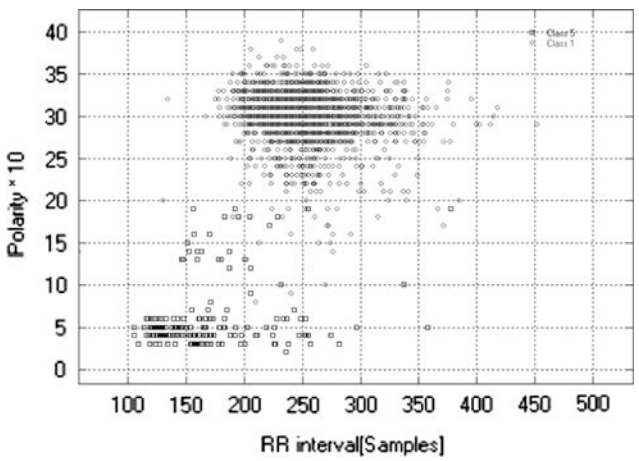

Fig. 3 Heartbeat class plot of MIT register 210. Separability of class 1 (normal beats, *squares*) and class 5 (VPC, *circles*) using only RR interval and polarity is clear

2.5 Feature selection

Heartbeat lengths L_i are usually too long for a complete morphology analysis of an entire Holter register. We reduce the number of amplitude samples using a non-uniform sampling method based on trace segmentation [3].

Length of the morphology feature sequences is set to 120 (time duration normalization), namely, $L_i = 120, \forall\ i$. The basic morphology of the heartbeat has to be kept while the computation time is reduced. Trace segmentation (TS) samples the original heartbeat at those points of main changes [3], and therefore morphology of waves P, Q, R, S, and T is properly kept for a wide range of lengths whereas baseline points are omitted. At this stage, the obtained output heartbeat set is termed $\widehat{H}$, with:

$$\widehat{H} = \left\{\widehat{h}_1, \widehat{h}_2, \ldots, \widehat{h}_N\right\},$$

where $\widehat{h}_i = \text{TraceSegmentation}(h_i, 120)$. Algorithm to compute the TS of a heartbeat is described in Algorithm 2.

2.6 Morphology matching

Morphology matching is based on DTW, and it is the dissimilarity measure used in the clustering algorithms. DTW is a pattern matching technique that stretches the time axis of the two heartbeats under analysis in a non linear way and provides a quantitative dissimilarity measure obtained from the matrix of cummulative distances, the dynamic programming matrix G_{DTW}. Throughout this matrix, an optimal alignment path between the two heartbeats is found and the last node of the matrix (usually normalized by the length of this path, L_{DTW}), provides the dissimilarity measure d_{DTW}. A detailed description of this morphology matching method can be found in [2, 3]. General steps are in Algorithm 3.

There are three ways to reduce the computational cost of DTW:

- Reduce the size of the dynamic programming matrix, that is, reduce L_i and L_j. In this case, it is accomplished by setting the length of all the heartbeats to $L_H = 120$ by means of trace segmentation, $L_H = L_i = 120, \forall\ i$, as described before.
- Reduce the number of cells in the dynamic programming matrix to be examined. This limits the search area to a smaller region. A graphical

example of G_{DTW}, whose search region has been limited, is shown in Fig. 4. Areas far from the diagonal are initialized with a great value in order to prevent the method from searching for the alignment at unlikely nodes. The alignment path is therefore bounded to be in the valley, that is, in the area close to the matrix diagonal. This path traverses the nodes of minimum slope. In this case the width of the valley is set to $2*L_H/3$.

- Reduce the number of times DTW is computed. The two other features, RR interval and polarity, may suffice if the two heartbeats are very different.

Algorithm 2 . Non-uniform sampling function

```
function TraceSegmentation(h, L)
begin
Total := Sum(|Derivative(h)|);
Interval := Total/L;
k := 0;
j := 0;
for i = 1 to Length(h) do
    begin
    k := k + |h[i + 1] − h[i]|;
    if k ≥ Interval ∗ j then
        begin
        ĥ[j] := h[i];
        j := j + 1;
        end
    end
end
```

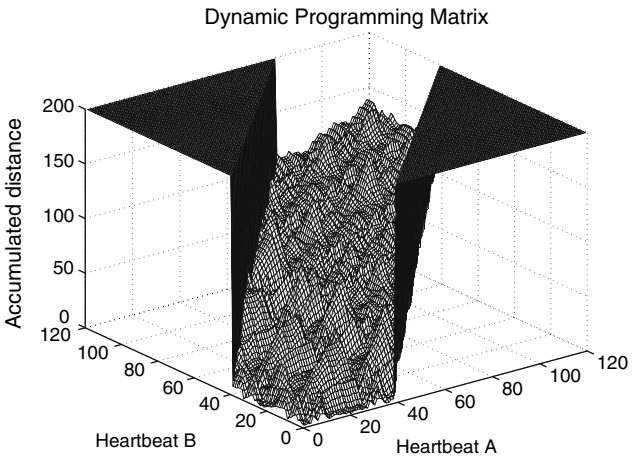

Fig. 4 Graphical representation of a dynamic programming matrix G when two heartbeats are aligned. Regions far from the diagonal have such a high cost (in theory, infinite, 200 in the example) that the alignment path is constrained to the central region, which is in addition the most likely region for usual alignments

2.7 Partitional clustering

The clustering problem is the problem of finding homogeneous groups of objects in a given set, according to certain criteria. In this case, given N heartbeats in $\widehat{H}$, the objective is to find a partition C of N_C clusters, $C = \{C_1, C_2, \ldots, C_{N_C}\}$ in such a way that heartbeats in a cluster are similar with regard to RR interval, polarity, and morphology.

Algorithm 3 . Computes G_{DTW} and therefore d_{DTW}

```
function d_DTW(ĥ_i, ĥ_j)                          {Two heartbeats input}
begin
Initialize(G_DTW);                                {Global constraints for G_DTW}
G_DTW[1, 1] := |ĥ_i[1] − ĥ_j[1]|;
for k_1 = 2 to L_i do
    for k_2 = 2 to L_j do
        begin
        D := |ĥ_i[k_1] − ĥ_j[k_2]|;               {Local distance}
        V := G_DTW[k_1, k_2 − 1] + D;
        H := G_DTW[k_1 − 1, k_2] + D;
        Diagonal := G_DTW[k_1 − 1, k_2 − 1] + 2 ∗ D;
        if Diagonal ≤ V & Diagonal ≤ H then
            G_DTW[k_1, k_2] := Diagonal
        else
            if H ≤ V & H ≤ Diagonal then
                G_DTW[k_1, k_2] := H
            else
                G_DTW[k_1, k_2] := V;
        end
d_DTW := G_DTW[L_i, L_j];
end
```

For this first stage of partition initialization, the *k*-means is used, with a relatively high number of clusters, and the final clustering is obtained using a hierarchical agglomerative approach. Every cluster is termed C^t_i, where the superindex t accounts for a general time index (clustering proceeds iteratively). Every cluster is a tuple that comprises the following fields:

- A list of heartbeats that at time or iteration t are included in the cluster, $\left\{\widehat{h}_{i1}, \widehat{h}_{i2}, \ldots, \widehat{h}_{ij}, \ldots, \widehat{h}_{iN_i}\right\}$.
- The average RR interval of the heartbeat list, R_i, as well as $R_{i_{\max}}$ and $R_{i_{\min}}$.
- The average polarity of the heartbeat list, P_i, as well as $P_{i_{\max}}$ and $P_{i_{\min}}$.
- A representative centroid z_i, obtained from the most centered heartbeat in the list, namely, the heartbeat whose normalized RR interval and polarity are the nearest to the average ones.

The objective of a partition initialization is to obtain a tradeoff between a raw classification where all the heartbeat types are represented, which assures global maximum convergence, and a reasonably low number of clusters.

In order to reach this objective, we based our initial partition on the qualitative information we already know about VPC: they are premature heartbeats, their RR interval is usually shorter, polarity between T and R waves often changes, and normal beats are majority in the heartbeat set. Therefore, if two heartbeats belong to the same class, their RR interval and polarity should be similar. Based on this idea, we apply loose bounds to decide whether two heartbeats may belong to the same class or not. Given two general heartbeats $\widehat{h}_i$ and $\widehat{h}_j$, these bounds are:

- The RR length ratio should be within an interval $b_{R_1} \leq \frac{R_i}{R_j} \leq b_{R_2}$ where the lower and higher bounds are set as $b_{R_1} = 0.75$ and $b_{R_2} = 1.25$. This interval is termed I_R.
- The polarity ratio should be within an interval $b_{P_1} \leq \frac{P_i}{P_j} \leq b_{P_2}$, where the lower and higher bounds are set as $b_{P_1} = 0.5$ and $b_{P_2} = 1.5$. This interval is termed I_P.
- Since it is difficult to fix distance bounds because the value obtained varies greatly, we applied a bound to the alignment path length L_{DTW} and used this measure as the dissimilarity instead of d_{DTW} (Algorithm 4). The value employed for this bound is $1.5L_H = 180$.
- To prevent the clustering from using too many clusters if the recording is noisy, the maximum number of initial clusters for the *k*-means algorithm $N_{C_{\max}}$ is set to 50. This number is typically reduced as the hierarchical clustering proceeds.

These bounds help to limit the search space to the most likely region, and therefore reduce the computational cost. They are very loose and conservative, and hence the exact value is not crucial.

Beats are classified to the minimum distance cluster (alignment path length), provided RR interval, polarity and alignment path length satisfy the constraints mentioned above, otherwise a new cluster is created and initialized with that heartbeat. In order to find the best centroids for each cluster, the algorithm iterates four times over the heartbeat set and at the end of each iteration, centroids are recomputed taking the most centered heartbeat of every cluster.

Algorithm 4 . Function that computes L_{DTW}

```
function DTWPathLength(G_DTW)
begin
i := L_i;
j := L_j;                                   {Last node}
L_DTW := 0;
while (i ≠ 1)or(j ≠ 1) do
   begin
   L_DTW := L_DTW + 1;
   H := ∞;                                  {Horizontal}
   V := ∞;                                  {Vertical}
   D := ∞;                                  {Diagonal}
   if (i − 1) ≥ 1 & (j − 1) ≥ 1 then
      D := G_DTW [i − 1, j − 1];
   if (i − 1) ≥ 1 then
      H := G_DTW [i − 1, j];
   if (j − 1) ≥ 1 then
      V := G_DTW [i, j − 1];
   if D ≤ H & D ≤ V & i > 1 & j > 1 then
      begin
      i := i − 1;
      j := j − 1;
      end
   else
      if V ≤ D & V ≤ H & j > 1 then j := j − 1
      else
         i := i − 1;
   end
DTWPathLength := L_DTW + 1;
end
```

The output of this stage is a new set of clusters $C^1 = \{C_1, C_2, \ldots, C_{N_C}\}$, $2 \leq N_C \leq 50$, where all the heartbeats in a cluster satisfy the bounds for RR interval, polarity, and L_{DTW}. Pseudocode of this stage is shown in Algorithm 5.

2.8 Hierarchical clustering

Once an stable initial partition has been found, an agglomerative hierarchical clustering [15] is conducted to join similar clusters. At each iteration, the dissimilarity among centroids of the clusters is computed. For the two nearest centroids, if L_{DTW} is below the bound, as well as the RR ratio and polarity, these two clusters are merged, and the parameters of the resulting cluster recomputed (Algorithm 6). With an updated partition, this process is repeated until no more clusters can be merged. The output of this stage is the final partition C^f with N_C clusters.

Algorithm 5 . k-means based clustering to obtain a complete initial partition

```
function GetInitialPartition(Ĥ)
begin
for Iter = 0 to Iter_max do
    begin
    for i = 1 to N do
        begin
        L_min := ∞;
        n := −1;
        for j = 1 to N_C do
            begin
            ρ_R := R_i / C_j.R;
            ρ_P := P_i / C_j.P;
            if ρ_R ∈ I_R & ρ_P ∈ I_P then
                begin
                L_DTW := DTWPathLength(Ĥ[i], z_i);
                if L_DTW < L_min then
                    begin
                    L_min := L_DTW;
                    n := j;
                    end
                end
            end
        if (L_min < 1.5 * L_H) or (N_C ≥ N_C_max) then
            AddHeartbeatToCluster(Ĥ[i], C_n);
        else
            if N_C < N_C_max then
                CreateNewCluster(Ĥ[i]);
        end
    if Iter < Iter_max − 1 then                {Centroid recalculation}
        begin
        for i = 1 to N_C do
            begin
            z_i := MostCenteredHeartbeat(C_i);
            C_i := φ;                          {Clears beat list}
            end
        end
    end
end
```

When the final partition C^f has been found, heartbeats in each cluster are matched with original heartbeats in H. Then heartbeats in the input signal $y[n]$ are labelled according to the clustering results, and the method finishes.

Algorithm 6 . Merges those clusters whose centroids satisfy the bounds for RR interval, polarity, and path length

```
procedure MergeClusters(C^1 = {C_1, C_2, ..., C_N_C})
begin
repeat
    L_min := ∞;
    k_1 := −1;
    Found := false;
    for i = 1 to N_C do
        for j = i + 1 to N_C do
            begin
            ρ_R := R_i / R_j;
            ρ_P := P_i / P_j;
            if ρ_R ∈ I_R & ρ_P ∈ I_P then
                begin
                L_DTW := DTWPathLength(z_i, z_j);
                if L_DTW < L_min then
                    begin
                    L_min := L_DTW;
                    k_1 := i;
                    k_2 := j;
                    end
                end
            end
    if k_1 ≠ −1 & L_min < 1.5 * L_H then
        begin
        Merge(C_k1, C_k2);
        Delete(C_k2);                          {N_C := N_C − 1}
        Found := true;
        end
until Found = false
end
```

3 Results

3.1 Validation database

Annotated registers from the MIT arrhythmia database [14] were used for validation since QRS locations and the types of heartbeat are identified. Additionally, these datasets have been used by other researchers to asses the performance of similar algorithms. This facilitates segmentation, performance assessment, and comparison.

The specific registers utilized are shown in Table 1. Name of the register, total number of beats, number of normal and VPC, and the average RR interval length and polarity, are included. Leads were processed independently. Only heartbeats of class 1 (normal) or class 5 (VPC) were kept in the set.

3.2 Experiments

We used the proposed algorithm to analyze the database described above. The algorithm was implemented as part of a software application and applied using a personal computer with 1 GB of main memory, and a Centrino Duo processor running at 1.66 GHz. Signals were loaded into the software application, and they underwent the stages described, from filtering to final clustering. No interaction with the user was required. Results obtained are shown in Table 2. Each column of the table provides the following information:

- *Clusters found:* The number of clusters the method found according to the restrictions described for the bounds.
- *Time:* Time in seconds elapsed from the preprocessing of the lead till the end of the clustering. Only the first lead was processed.
- *True positive* (*TP*): The heartbeat was normal (MIT label 1) and it was classified as normal beat.
- *True negative* (*TN*): The heartbeat was VPC (MIT label 5) and it was clasified as VPC beat.
- *False positive* (*FP*): A VPC hearbeat grouped in a cluster where normal heartbeats were dominant.
- *False negative* (*FN*): A normal heartbeat grouped in a cluster where VPC heartbeats were dominant.
- *SE*: Sensitivity, defined as $\mathrm{SE} = \frac{\mathrm{TN}}{\mathrm{TN+FP}}$. It measures the ratio of correctly grouped normal heartbeats.
- *SP*: Specificity, defined as $\mathrm{SP} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$. It measures the ratio of correctly grouped VPC heartbeats.
- *PPA*: Positive Predictive Accuracy, defined as $\mathrm{PPA} = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$. It measures the ratio of correctly grouped positives.

Table 1 Registers from the MIT arrhythmia database used in the experiments

Register	Total beats	N	VPC	$\bar{R}$	$\bar{P}$
100	2,240	2,239	1	287.02	4.08
102	103	99	4	285.08	1.93
104	165	163	2	276.56	2.71
105	2,567	2,526	41	252.58	2.74
106	2,027	1,507	520	320.56	2.56
108	1,756	1,740	16	368.10	0.88
114	1,863	1,820	43	346.34	0.98
115	1,953	1,953	0	332.80	2.05
116	2,411	2,302	109	269.51	2.80
119	1,987	1,543	444	327.02	3.70
121	1,862	1,861	1	348.89	3.88
122	2,476	2,476	0	346.34	0.88
123	1,518	1,515	3	427.99	1.59
200	2,569	1,743	826	250.34	1.33
201	1,823	1,625	198	338.91	4.11
202	2,080	2,061	19	307.57	3.91
203	2,973	2,529	444	218.10	1.79
205	2,642	2,571	71	244.77	5.85
208	2,578	1,586	992	221.23	3.90
209	2,622	2,621	1	225.82	1.77
210	2,617	2,423	194	245.90	2.79
213	2,861	2,641	220	200.45	2.28
215	3,360	3,196	164	193.25	1.37
217	406	244	162	247.80	1.90
219	2,146	2,082	64	298.20	2.54
221	2,427	2,031	396	267.74	2.74
223	2,502	2,029	473	250.62	2.78
228	2,030	1,668	362	316.59	3.26
230	2,256	2,255	1	288.04	1.38
231	316	314	2	601.43	5.62
233	3,061	2,230	831	211.20	2.38
234	2,703	2,700	3	236.98	4.35

$\bar{R}$ and $\bar{P}$ represent the average RR interval and polarity, respectively

- *CP*: Clustering performance, defined as $\mathrm{CP} = \frac{\mathrm{TN}+\mathrm{TP}}{\mathrm{TN}+\mathrm{FP}+\mathrm{TP}+\mathrm{FN}}$. It measures the ratio of correctly grouped heartbeats.

In addition to assessing the ability of the algorithm as a detector of VPCs, we also assessed its performance for other applications. Table 3 shows the results when using normal and atrial premature contraction (APC) heartbeats, Table 4 using VPC and APC, Table 5 using normal and right bundle branch block beats (RBBB), and Table 6 using left bundle branch block heartbeats (LBBB).

In order to better evaluate the performance under general situations, we also used registers with more than just two different heartbeat types. Registers used were 102, 104, 107, and 217, with heartbeat types: normal, VPC, paced, and fusion of paced and normal beats. Results were:

1. Register 102. $K = 4$, $t = 35$ s. VPC were correctly classified, the rest of types were considered the same, namely, normal, paced and fusion beat were classified together.
2. Register 104. $K = 7$, $t = 43$ s. VPC were correctly classified, fusion beats were considered as normal or paced.
3. Register 107. $K = 6$, $t = 34$ s. 57 VPC were incorrectly classified as paced, possibly because morphologies were not very different. No other types in this register.
4. Register 217. $K = 15$, $t = 39$s. Only 3 VPC were incorrectly classified as paced. The rest of types (normal, paced, and fusion) were classified together.

Finally, another experiment took place with heartbeats where R location was omitted in 1 or 10% of the heartbeats. As stated before, segmentation was intentionally not ideal, to better account for real cases, but no heartbeat was omitted until now. Results are shown in Table 7.

4 Discussion

The results of our assessment study indicate that our proposed algorithm has excellent performance. The

Table 2 Results of the experiments using only normal and VPC types

Reg.	*K*	*T* (s)	TP	TN	FP	FN	SE	SP	PPA	CP
100	5	40	2,239	1	0	0	100.0	100.0	100.0	100.0
102	3	1	99	4	0	0	100.0	100.0	100.0	100.0
104	3	2	163	2	0	0	100.0	100.0	100.0	100.0
105	21	53	2,525	41	0	1	100.0	99.9	100.0	99.9
106	7	37	1,507	520	0	0	100.0	100.0	100.0	100.0
108	20	42	1,740	13	3	0	81.2	100.0	99.8	99.8
114	11	34	1,820	43	0	0	100.0	100.0	100.0	100.0
115	4	38	1,953	0	0	0	100.0	100.0	100.0	100.0
116	20	46	2,302	107	2	0	98.1	100.0	99.9	99.9
119	4	26	1,543	444	0	0	100.0	100.0	100.0	100.0
121	11	35	1,861	1	0	0	100.0	100.0	100.0	100.0
122	3	35	2,476	0	0	0	100.0	100.0	100.0	100.0
123	3	24	1,512	3	0	3	100.0	99.8	100.0	99.8
200	10	51	1,743	821	5	0	99.3	100.0	99.7	99.8
201	9	28	1,625	198	0	0	100.0	100.0	100.0	100.0
202	7	36	2,061	19	0	0	100.0	100.0	100.0	100.0
203	30	85	2,479	417	27	50	93.9	98.0	98.9	97.4
205	14	51	2,571	71	0	0	100.0	100.0	100.0	100.0
208	15	49	1,585	991	1	1	99.8	99.9	99.9	99.9
209	2	51	2,621	1	0	0	100.0	100.0	100.0	100.0
210	23	61	2,421	188	6	2	96.9	99.9	99.7	99.6
213	6	46	2,641	188	32	0	85.4	100.0	98.8	98.8
215	7	69	3,195	158	6	1	96.3	99.9	99.8	99.7
217	9	7	244	159	3	0	98.1	100.0	98.7	99.2
219	9	43	2,082	64	0	0	100.0	100.0	100.0	100.0
221	6	40	2,031	396	0	0	100.0	100.0	100.0	100.0
223	14	49	2,020	468	5	9	98.9	99.5	99.7	99.4
228	20	42	1,668	362	0	0	100.0	100.0	100.0	100.0
230	4	38	2,255	1	0	0	100.0	100.0	100.0	100.0
231	3	4	314	2	0	0	100.0	100.0	100.0	100.0
233	7	60	2,229	827	4	1	99.5	99.9	99.8	99.8
234	5	43	2,700	3	0	0	100.0	100.0	100.0	100.0

The values of SE, SP, PPA, and CP are in percentage

Table 3 Results of the experiments using only normal and APC types

Reg.	*K*	*T* (s)	TP	TN	FP	FN	SE	SP	PPA	CP
103	6	33	2,082	0	2	0	0.0	100.0	100.0	99.9
112	2	40	2,537	0	2	0	0.0	100.0	100.0	99.9
117	3	21	1,533	1	0	1	100.0	99.9	100.0	99.9
220	4	34	1,953	87	7	1	92.5	99.9	99.6	99.6
222	24	76	1,964	120	88	98	57.6	95.2	95.7	91.8

The values of SE, SP, PPA, and CP are in percentage. TN and FP refer to APC

Table 4 Results of the experiments using only VPC and APC types

Reg.	*K*	*T* (s)	TP	TN	FP	FN	SE	SP	PPA	CP
118	4	45	2,163	25	71	3	26.0	99.8	96.8	96.7
207	5	2	86	107	0	0	100.0	100.0	100.0	100.0
232	10	29	432	1,382	0	1	100.0	99.7	100.0	99.9

The values of SE, SP, PPA, and CP are in percentage. TP and FN refer to VPC, and TN and FP refer to APC

algorithm was capable of performing accurate clustering without prior information about the number of clusters and without a training set. These results indicate that the algorithm may be used in a clinical environment.

Most of the registers (17) were processed without classification errors (Fig. 5). Others (7) only had errors in one heartbeat type, and the rest presented a low error percentage. Registers 115 and 122, with only normal heartbeats, were included in the test to analyse

Table 5 Results of the experiments using only normal and RBBB types

Reg.	*K*	*T* (s)	TP	TN	FP	FN	SE	SP	PPA	CP
212	4	42	919	1,822	3	4	99.9	99.5	99.6	99.7
231	4	21	314	1,253	1	0	99.9	100.0	99.6	99.9

The values of SE, SP, PPA, and CP are in percentage. TN and FP refer to RBBB

Table 6 Results of the experiments using only LBBB and VPC types

Reg.	*K*	*T* (s)	TP	TN	FP	FN	SE	SP	PPA	CP
109	6	41	2,491	34	4	1	89.4	99.9	99.8	99.8
111	4	38	2,123	0	1	0	0.0	100.0	99.9	99.9
207	17	32	1,447	104	1	10	99.0	100.0	99.3	99.2
214	11	40	2,003	256	0	0	100.0	100.0	100.0	100.0

The values of SE, SP, PPA, and CP are in percentage. TP and FN refer to LBBB

Table 7 Results of the experiments when QRS detection is incorrect (1% or 10% error detection)

Reg.	*K*	*T* (s)	TP	TN	FP	FN	SE	SP	PPA	CP
106	8	39	1,507	520	0	0	100.0	100.0	100.0	100.0
119	5	29	1,543	444	0	0	100.0	100.0	100.0	100.0
200	13	51	1,743	815	11	0	98.6	100.0	99.3	99.5
106	8	32	1,507	491	29	0	94.4	100.0	98.1	98.5
119	5	24	1,543	444	0	0	100.0	100.0	100.0	100.0
200	19	42	1,741	819	7	2	99.1	99.8	99.5	99.6

The values of SE, SP, PPA, and CP are in percentage

the performance when no VPC is found. Results for register 209 were ideal, a register with two hearbeat types, and with two corresponding clusters. Results for register 203 were the worst due to the noise and the lack of different QRS morphologies and RR lengths.

These results indicate that our algorithm has a performance comparable to other published algorithms [1, 5, 13]. This is a remarkable result since other algorithms typically require more user-specified parameters, a training set and more leads.

The number of clusters changed depending on the register structure (i.e. morphology, noise, outliers) from 2 to 30 clusters. The number of cluster detected were generally less than the fixed 25 or 18 clusters used in other works [6, 13]. Processing time varied accordingly with the number of clusters.

Results for other heartbeat types were not as good as with normal and VPC. The accuracy is still high when the morphology is different for the two types under test (LBBB, RBBB, VPC). The worst results were obtained for normal and APC, since their morphology is the same and the only difference is the prematurity of APC. These conditions are difficult to detect even for cardiologists.

Finally, our results indicate that the algorithm performs well even in situations where the R was not correctly detected. In these cases the number of clusters is usually greater in order to allocate the new morphologies found (two heartbeats as one), but the classification accuracy is still high.

5 Conclusion

We described a novel method to solve the problem of ECG beat clustering using an unsupervised approach. Our results demonstrate that our proposed algorithm has good performance and may be used in clinical practice.

The polarity measure defined has proven to be a fast and effective method to compare the morphology of two heartbeats, which usually is quite different in VPC and in normal beats. This measure could be used as a characterization feature in other ECG processing and analysis applications.

Ventricular extrasystoles detection is an important preprocessing step necessary in a variety of applications such as extrasystolic potentiation assesment, VE

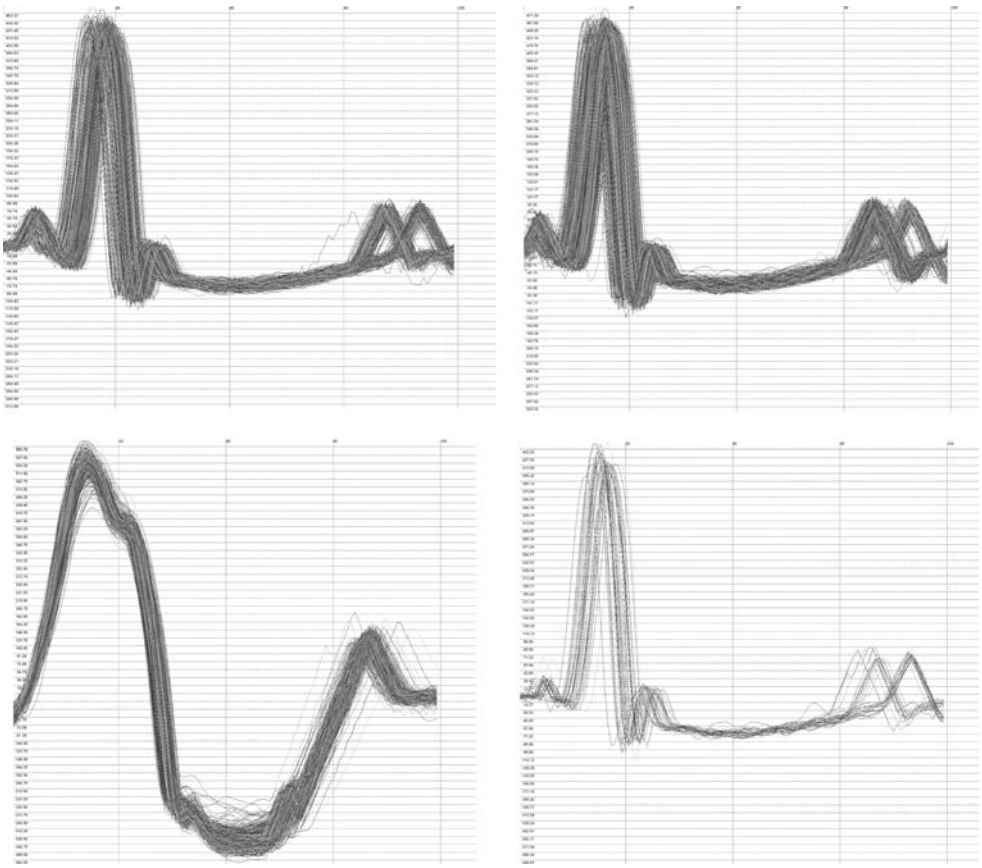

Fig. 5 Clusters obtained for register 119. In this case, and due to the loose bounds, there are more clusters (4), than necessary (2), but this method is aimed at obtaining a very high clustering performance with a reasonable low number of clusters, in a non supervised way. Besides, the method must work with easy cases, this one, as well as with difficult ones without parameter readjustment

per time unit measurement, classification according to relationship to normal beats (bigeminy, trigeminy, quadrigeminy, couplet and nonsustained VE), classification according to origin (unifocal or multifocal), and classification according to frequency (frequent or occasional).

References

1. Chazal P, O'Dwyer M, Reilly RB (2004) Automatic classification of heartbeats using ecg morphology and heartbeat interval features. IEEE Trans Biomed Eng 51(7):1196–1206
2. Cuesta-Frau D, Aboy M, McNames J, Goldstein B (2003) Morphology analysis of intracranial pressure using pattern matching techniques. In: The 25th annual conference of the IEEE engineering in medicine and biology society
3. Cuesta-Frau D, Pérez-Cortés JC, Andreu-García G (2003) Clustering of electrocardiographic signals in computer-aided holter analysis. Comput Methods Programs Biomed 72(3):179–196
4. Dave J, Gandhi G, Nazeer S, Jha S, Kocheril A (2005) Ventricular premature complexes. emedicine, http://www.emedicine.com ([Online])
5. Dokur Z, Olmez T (2001) Ecg beat classification by a novel hybrid neural network. Comput Methods Programs Biomed 66(2–3):167–181
6. Engin M (2004) Ecg beat classification using neuro-fuzzy network. Pattern Recogn Lett 25(15):1715–1722
7. Friedman HH (1989) Diagnostic electrocardiography and vectorcardiography, 3rd edn. McGrawHill, New York
8. Goletsis Y, Papaloukas C, Fotiadis DI, Likas A, Michalis LK (2004) Automated ischemic beat classification using genetic algorithms and multicriteria decision analysis. IEEE Trans Biomed Eng 51(10):1717–1725
9. Heartpoint (1997) ventricular arrhythmias. Heartpoint gallery [Online] http://www.heartpoint.com
10. Holter NJ (1961) New methods for heart studies. Science 134:1214
11. Kadambe S, Murray R, Boudreaux-Bartels GF (1999) Wavelet transform-based qrs complex detector. IEEE Trans Biomed Eng 46(7):838–848
12. Kohler BU, Henning C, Orglmeister R (2002) The principles of software QRS detection. IEEE Eng Med Biol Mag 21(1):42–57
13. Lagerholm M, Peterson C, Braccini G, Edendrandt L, Sornmo L (2000) Clustering ECG complexes using hermite functions and self-organizing maps. IEEE Trans Biomed Eng 47:838–848
14. Mark R, Moody G (1997) MIT-BIH arrhythmia database. http://ecg.mit.edu/dbinfo.html
15. Mingoti SA, Lima JO (2006) Comparing som neural network with fuzzy c-means, k-means, and traditional hierarchical clustering algorithms. Eur J Oper Res 174(3):1742–1759
16. Shyu L, Wu Y, Hu W (2004) Using wavelet transform and fuzzy neural network for vpc detection from the holter ecg. IEEE Trans Biomed Eng 51(7):1269–1273