



HAL
open science

Multi-Dynamics Analysis of QRS Complex for Atrial Fibrillation Diagnosis

Y. Trardi, Bouchra Ananou, Z. Haddi, M. Ouladsine

► **To cite this version:**

Y. Trardi, Bouchra Ananou, Z. Haddi, M. Ouladsine. Multi-Dynamics Analysis of QRS Complex for Atrial Fibrillation Diagnosis. 2018 5th International Conference on Control, Decision and Information Technologies (CoDIT), Apr 2018, Thessaloniki, France. 10.1109/CoDIT.2018.8394935 . hal-01893141

HAL Id: hal-01893141

<https://amu.hal.science/hal-01893141>

Submitted on 11 Oct 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-Dynamics Analysis of QRS Complex for Atrial Fibrillation Diagnosis

Y. Trardi, B. Ananou, Z. Haddi, and M. Ouladsine

Abstract—This paper presents an effective atrial fibrillation (AF) diagnosis algorithm based on multi-dynamics analysis of QRS complex. The idea behind this approach is to produce a variety of heartbeat time series features employing several linear and nonlinear functions via different dynamics of the QRS complex signal. These extracted features from these dynamics will be connected through machine learning based algorithms such as Support Vector Machine (SVM) and Multiple Kernel Learning (MKL), to detect AF episode occurrences. The reported performances of these methods were evaluated on the Long-Term AF Database which includes 84 of 24-hour ECG recording. Thereafter, each record was divided into consecutive intervals of one-minute segments to feed the classifier models. The obtained sensitivity, specificity and positive classification using SVM were 96.54%, 99.69%, and 99.62%, respectively, and for MKL they reached 95.47%, 99.89%, and 99.87%, respectively. Therefore, these medical-oriented detectors can be clinically valuable to healthcare professional for screening AF pathology.

I. INTRODUCTION

AF or A-fib is the most common type of cardiac arrhythmia, which has a severe impact on life quality and increases the risk of stroke, and heart failure [1]. Rapid and efficient method of diagnosis is required to create a flexible machine that can detect the AF occurrence in ECG signal. Further, different algorithms have been proposed in the literature for automatic AF detection and classification. Most reported studies are based only on inter-beat time series. These inter-beats are usually called RR segments by the cardiologist community, which can be computed by measuring the time difference between two successive R-peaks (detected in the QRS complex). The RR segments can be considered as a dynamic that describes the rhythmic behavior of the heartbeat signal. Indeed, we have addressed four dynamics of the QRS complex. This new methodology was first proposed by [2], which was focused on the analysis of different characterizations within the time series interval of the heartbeat signal. Further, we could assume that each dynamic is considered as a sensor that produces more knowledge about database characteristics. Several methods have been proposed in the literature for automatic cardiac arrhythmia detection and classification, such as Neural Network [3], Fuzzy C-Means clustering [4], Wavelet Transform and Support Vector Machine [5], Autoregressive Modeling [6], Bayesian methods [7], and many others. These reported studies focused on variables extracted from RR interval time series or the analysis of the ECG

signal. However, the current study proposes a multivariate analysis that aimed to couple fourteen linear and nonlinear functions with the four aforementioned dynamics. Finally, for an effective AF detection, we applied support vector machine (SVM) and multiple kernel learning (MKL) classification algorithms using a combination of 56-feature.

II. MATERIAL AND METHODS

A. Database treatment

The proposed methods in this paper was evaluated on PhysioBank free accessible databases, called Long-Term AF Database including 16078 AF and 18502 NSR episodes. The operation procedures performed in this study were separated into three parts as shown in (Fig. 1). The first part is ECG signal processing to extract the QRS complex; the second part is dedicated for data analysis; and finally, the last part is for the training of classification models.

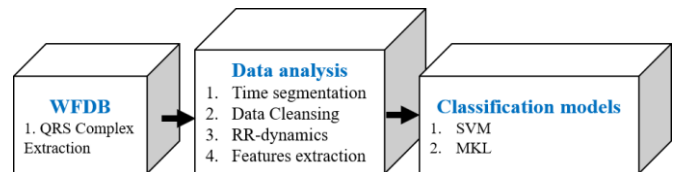


Figure 1. The data reprocessing and classification blocks

1) The QRS complex extraction

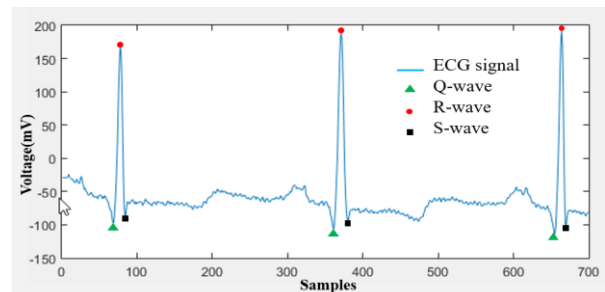


Figure 2. Thresholding peaks in signal

Firstly, WFDB (WaveForm DataBase) Software Package-Physionet is used to extract the QRS complex from each ECG recording. WFDB is a set of MATLAB functions and wrappers

for reading, writing, and processing files in the PhysioBank databases formats. The WFDB process is used to generate two vectors: the first one concerns R-peaks location in the QRS complex (Fig. 2) and the second shows the annotations of each interval.

2) Data analysis

We have developed a data reprocessing algorithm containing four operational blocks described by: time segmentation; cleaning process; RR dynamics calculation; and features extraction; as shown in (Fig. 3). This process takes the R-peaks interval as input and provides the following outputs: RR segments; a repertory called T contains the width of each segment extracted from the R-peaks signal; a repertory called Y contains the annotations affected to each segment. Generally, Y could have several arrhythmias that are exchanged. In our study, we targeted NSR and AF rhythms.

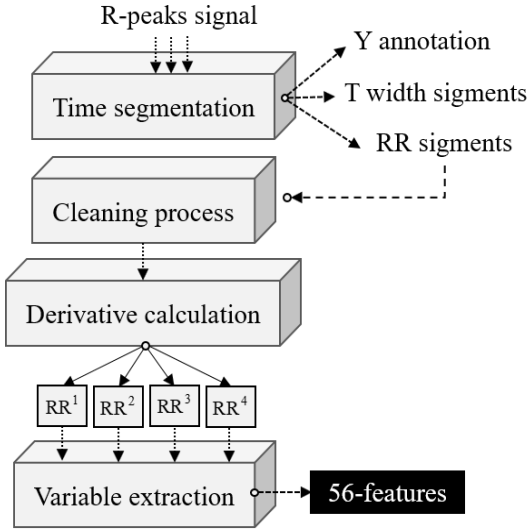


Figure 3. Data analysis process

a) Time segmentation

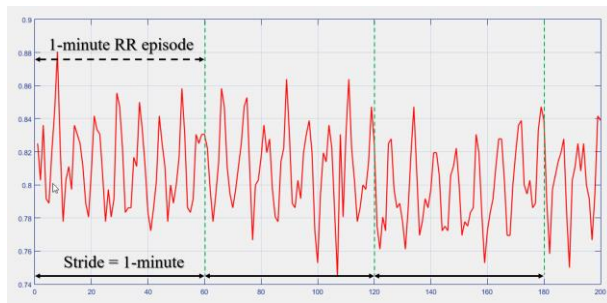


Figure 4. Illustration of 1-minute segmentation process

The R-peaks signal extracted by the WFDB process must be segmented, we have chosen a 1-minute window to divide it into a set of RR segment as defined by equation (1).

$$RR = R(2 \rightarrow N) - R(1 \rightarrow N-1) \quad (1)$$

Where N is the total number of components in the R-peaks signal.

Another important parameter that must be chosen is the slip pitch. Generally, the slip pitch must be lower or equal to the splitting window. In the current study, we choose a slip pitch equal to 1-minute, as showed in Fig. 4.

b) Cleaning process

Each 1-minute segment RR is subjected to some constraints that must be respected. Firstly, the components of the 1-minute segment RR are bounded. All $RR(n)$ values must be included in $[0.3s, 2s]$, which means that, between two successive R-peaks, we cannot have a duration greater than 2 seconds or less than 0.3 second. Secondly, the duration of 1-minute segment RR should not exceed $T \geq (60 + \varepsilon)$ seconds. In this study, we have fixed ε equal to 2 seconds. This constraint is of great importance when a timeout is encountered during the ECG recording (or missing peak's detection).

c) RR-dynamics

The dynamics employed in this investigation are described as follow:

- The first dynamic is redescribed as:

$$RR^1 = R(2 \rightarrow N) - R(1 \rightarrow N-1), \quad (2)$$

- Second dynamic:

$$RR^2 = \text{abs}(RR^1(2 \rightarrow N) - RR^1(1 \rightarrow N-1)) \quad (3)$$

- Third dynamic:

$$RR^3 = \text{abs}(RR^2(2 \rightarrow N) - RR^2(1 \rightarrow N-1)) \quad (4)$$

- Fourth dynamic:

$$RR^4 = \text{abs}(RR^3(2 \rightarrow N) - RR^3(1 \rightarrow N-1)) \quad (5)$$

d) Variable extraction

Various measures of complexity were developed to compare time series and distinguish regularity, chaotic and random behavior. Linear and nonlinear variables are used for the analysis of heartbeat time series signal.

The extracted features, used in this study, correspond to those listed in previous related works. All of them are applied to each of the four dynamics. For a better understanding, we use an index (i) to differentiate between dynamics, and we use the parameter N to represent the number of components in the i^{th} dynamics.

(1) Linear analysis

Mean: Mean value of RR components within each segment.

RMSSD: is a time-domain method used to quantify the heart variability (HRV). This refers to the root mean square of successive RR components within each segment. The equation to describe RMSSD is giving:

$$\text{RMSSD} = \sqrt{\frac{1}{N} \sum_{n=1}^N \text{RR}^i(n)} \quad (6)$$

MMedians: This refers to the median of medians, calculated by dividing the RR interval into three segments. Each segment was centralized by subtracting his own average value. Next, we calculate the median for each segment. And finally, a median of the three medians was calculated.

SDSD: This refers to the standard deviation of differences between the adjacent RR components within each segment.

Kurtosis: is the ratio of the fourth moment and the second moment squared.

$$\text{Kurtosis} = N \frac{\sum_{n=1}^N (\text{RR}^i(n) - \text{Mean})^4}{\left(\sum_{n=1}^N (\text{RR}^i(n) - \text{Mean})^2\right)^2} \quad (7)$$

Skewness: is the ratio of the third moment and standard deviation cubed.

$$\text{Skewness} = \sqrt{N} \frac{\sum_{n=1}^N (\text{RR}^i(n) - \text{Mean})^3}{\left(\sum_{n=1}^N (\text{RR}^i(n) - \text{Mean})^2\right)^{3/2}} \quad (8)$$

(2) Non-Linear analysis

Nine nonlinear methods were also investigated to characterize the studied arrhythmia. Four of them were based on the scatter plot of the RR segment [8].

VAI: Vector Angular Index is calculated as:

$$\text{VAI} = \sum_{n=1}^N |\theta_n - 45| / N \quad (9)$$

Where θ_n is the angle between the line plotted from every scatter point to the original point and the x-axis, N is the number of scatter points.

VLI: Vector length Index is calculated as:

$$\text{VLI} = \sqrt{\sum_{n=1}^N (l_i - L)^2} / N \quad (10)$$

Where l_i is the length between every scatter point and the original point, L is the mean of all the l_i , and N is the number of scatter points.

SD1: is the standard deviation calculated as:

$$\text{SD1} = \text{var}\left(\left|\text{RR}^i_{n+1} - \text{RR}^i_n\right|/\sqrt{2}\right) \quad (11)$$

SD2: is the standard deviation calculated as:

$$\text{SD2} = \text{var}\left(\left|\text{RR}^i_{n+1} - \text{RR}^i_n\right|/\sqrt{2}\right) - 2 \times \text{Mean}(\text{RR}^i) \quad (12)$$

Where $\text{RR}^i_n = \text{RR}^i(1 \rightarrow N-1)$, $\text{RR}^i_{n+1} = \text{RR}^i(2 \rightarrow N)$, and the $\text{Var}(\cdot)$ is the standard deviation.

The five remaining variables were based on different types of entropy which provide a valuable tool for quantifying the regularity of physiological time series. For each type of entropy, a set of certain common parameters is needed to be initialized: embedding dimension m, tolerance threshold r and time series length N.

Following recommendations of some works dealing with these parameters [9], m was set at 2 and r represents 20 or 25% of the standard deviation RR segment. In the present study, we have used the following entropy methods.

ApEn: Approximate entropy was developed by Pincus [9,10] as a measure of regularity to quantify levels of complexity within time series.

SampEn: like ApEn, Sample entropy is a measure of complexity. But it is different from ApEn mainly by two points: (1) SampEn does not count self-matches; (2) SampEn does not use a template-wise approach [11,12].

FuzzyEn: Fuzzy entropy, a new measure of time series regularity. Like the two existing related measures ApEn and SampEn, FuzzyEn is the negative natural logarithm of the conditional probability excluding self-matches and considering only the first N-m vectors of length m. There are three parameters that must be fixed for each calculation of FuzzyEn: m, r and n defined as the gradient of the boundary of the exponential function [11].

COSEn: called coefficient of sample entropy, has a high degree of accuracy in distinguishing AF from NSR in 12-beat calculations performance [13].

QSE: Quadratic entropy rate, based on densities rather than probability estimates. To normalize the value of r, SampEn was modified by dividing the probability by the length of the overall tolerance window $2r$ [13]. All these 14-linear and nonlinear features were applied on each dynamic. As results, a set of 56 features was formed. In particular, three features were eliminated because they contain many of indeterminate value components given by the three methods SampEn, QSE, and COSEn. Once the data preprocess is done, the next obvious step is classification methods. SVM and MKL based algorithms were developed for the classification task. We should notice that we have created our own algorithms based on the same theory given in each method.

B. Classification methods

1) Support Vector Machine (SVM)

Currently, support vector machines (SVM) have proven to be an extremely effective tool for solving learning problems such as classification or regression [14]. The SVM, introduced by Vladimir Vapnik in the 1990s is a category of supervised algorithms designed to constructs a hyperplane or a set of

hyperplanes in a high- or infinite- dimensional space [15]. In the classification problems, we are given a training data set of couple-subject: $\{(x_1, y_1), \dots, (x_n, y_n)\}$, where x_i a vector subject and $y_i = \pm 1$ a target indicating to which class belongs each subject x_i . The traditional problem of SVM algorithm is to establish a characteristic function able to identify correctly all the learning dataset: $D(x) = y_i$. In general, the decision function is written as:

$$D(x) = \text{sign}(w^T x + b), \quad (13)$$

Where the equation $w^T x + b$ is the characteristic function, with $w \in \mathbb{R}^p$ and $b \in \mathbb{R}$ are the parameters that defined the optimal hyperplane. For a given $C > 0$ [16] and [17] have suggested the following optimization problem:

$$\begin{cases} \min_{w, b, \varepsilon} \frac{1}{2} w^T w + C \sum_{i=1}^n \varepsilon_i \\ \text{subject to: } y_i (w^T \Omega(x_i) + b) \geq 1 - \varepsilon_i, \quad \varepsilon_i \geq 0, \quad \forall i \end{cases} \quad (14)$$

Where n is the total number of subjects, $\Omega(x_i)$ is a nonlinear function that maps the training patterns into high-dimensional feature spaces. The parameter ε_i is a slack variable and C is the penalty parameter of the error term, which must be adjusted in order to maintain the compromise between a better margin of separation and a better generalization of the decision. To solve (14) we adopt the Lagrangian formulation with constraints defined as follows:

$$\begin{cases} \min_{\alpha} \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^n \alpha_i \\ \text{subject to: } \begin{cases} 0 \leq \alpha_i \leq C \\ \sum_{i=1}^n \alpha_i y_i = 0; \forall i \end{cases} \end{cases} \quad (15)$$

Where the kernel function is $K(x_i, x_j) = \Omega^T(x_i) \Omega(x_j)$, which represents the scalar products in feature space, and α_i are Lagrange multipliers. Once the optimization is done, these Lagrange multipliers are produced and used to determine the parameters w and b that define the optimal separator hyperplane:

$$w = \sum_{s=1}^k \alpha_s y_s \Omega(x_s) \quad (16)$$

$$b = \frac{1}{k} \sum_{s=1}^k \left(y_s - \sum_{r=1}^n \alpha_r y_r K(x_r, x_s) \right) \quad (17)$$

Where each index (s) defines the active constraints at the optimum, where $y_s (w^T \Omega(x_s) + b) = 1$. These indices (s) are obtained by extracting the positions of the Lagrangian multipliers that are strictly greater than zero (called support vectors indices). Then, the classification decision for a test (\bar{x}) which was not included in the training set is given by:

$$y(\bar{x}) = \text{sign} \left(\sum_{s=1}^n \alpha_s y_s K(\bar{x}, x_s) + b \right) \quad (18)$$

2) Multiple Kernel Learning (MKL)

MKL was first proposed by Lanckriet et al, considered conic combinations of kernel matrices for binary classification, resulting to a convex quadratically constrained quadratic programming problem [18]. Most published studies on MKL are focused on two issues, (i) how to improve the classification accuracy of MKL, and (ii) how to improve the learning efficiency. Several approaches were proposed to learn an appropriate kernel combination, including ℓ_1 -norm [19], ℓ_p -norm [20], entropy-based [21], and mixed norms [22]. The reasons to use MKL is their ability to learn from a larger predefined set of kernels and parameters an optimal linear (or nonlinear) combination of kernels. In order to identify a robust resolution of a machine learning problem. As well, instead of creating a new kernel, MKL algorithm can be used to combine kernels already established.

In this study, we have applied the MKL theory for the binary classification, that was developed by [23]. For any problem of kernel algorithms, the solution of the learning problem is always given by (18). Where the kernel $K(\dots)$ is a convex combination of basis kernels given by:

$$K(x, x') = \sum_{m=1}^M d_m K_m(x, x'), \quad \text{for } d_m \geq 0, \quad \sum_{m=1}^M d_m = 1 \quad (19)$$

Where M is the total number of kernels. The problem of the data representation through the kernel is then transferred to the choice of weights d_m .

The used MKL algorithm is a weighted ℓ_2 -norm regularization, or the ℓ_1 -norm constraint on the vector d is a sparsity constraint that will force some weights d_m to be zero. In the MKL framework, the decision function $f(x) = \sum_m f_m(x)$ is a combination of different $f_m(x)$ functions each affected to a kernel K_m . Further, the solution of the primal MKL problem is calculated by solving the following convex equations:

$$\begin{cases} \min_{f_m, b, \varepsilon, d} \frac{1}{2} \sum_m \frac{1}{d_m} \|f_m\|^2 + C \sum_i \varepsilon_i \\ \text{subject to: } \begin{cases} y_i \left(\sum_m f_m(x_i) + b \right) \geq 1 - \varepsilon_i, \quad \varepsilon_i \geq 0 \\ \sum_m d_m = 1 \quad d_m \geq 0 \quad \forall m \end{cases} \end{cases} \quad (20)$$

To solve (20) we consider the following constrained optimization problem:

$$\min_d J(d) \text{ such that } \sum_m d_m = 1 \quad d_m \geq 0 \quad \forall m \quad (21)$$

$$\text{Where: } J(d) = \begin{cases} \min_{f_m, b, \varepsilon, d} \frac{1}{2} \sum_m \frac{1}{d_m} \|f_m\|^2 + C \sum_i \varepsilon_i \\ y_i \left(\sum_m f_m(x_i) + b \right) \geq 1 - \varepsilon_i \quad \varepsilon_i \geq 0 \quad \forall i \end{cases} \quad (22)$$

Then we give the Lagrangian formulation of (22) with the combined kernel as:

$$\begin{cases} \max_{\alpha} -\frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \sum_{m=1}^M d_m K_m(x_i, x_j) + \sum_{i=1}^n \alpha_i \\ \text{subject to: } \begin{cases} 0 \leq \alpha_i \leq C \\ \sum_{i=1}^n \alpha_i y_i = 0 \end{cases}; \quad \forall i \end{cases} \quad (23)$$

To solve (21) we use a simple gradient method. Then, to solve (23) we put all $d_m = 1/M$, and we follow the same structure process adopted in the SVM algorithm.

$J(d)$ is considered as the optimal objective value of (23). Because of the strong duality, $J(d)$ is also the objective value of the dual problem:

$$J(d) = -\frac{1}{2} \sum_{i,j=1}^n \hat{\alpha}_i \hat{\alpha}_j y_i y_j \sum_{m=1}^M d_m K_m(x_i, x_j) + \sum_{i=1}^n \hat{\alpha}_i \quad (24)$$

Where $\hat{\alpha}$ maximizes (23). After $J(d)$ has been calculated, we solve (21). We start by calculating the derivatives of $J(d)$ as if $\hat{\alpha}$ does not depend on d .

$$\frac{\partial J}{\partial d_m} = -\frac{1}{2} \sum_{i,j} \hat{\alpha}_i \hat{\alpha}_j y_i y_j K_m(x_i, x_j), \quad \forall m \quad (25)$$

Then we compute the reduced gradient of $J(d)$ as follow:

$$[\nabla_{\text{red}} J]_m = \begin{cases} \frac{\partial J}{\partial d_m} - \frac{\partial J}{\partial d_\mu} & \text{for } m \neq \mu \\ \sum_{m \neq \mu} \left(-\frac{\partial J}{\partial d_m} + \frac{\partial J}{\partial d_\mu} \right) & \text{for } m = \mu \end{cases} \quad (26)$$

Where d_μ is the greatest component of the vector d and μ its index. Next, the descent direction for updating d is given by:

$$D_m = \begin{cases} 0 & \text{if } d_m = 0 \text{ and } -\frac{\partial J}{\partial d_m} + \frac{\partial J}{\partial d_\mu} > 0 \\ -\frac{\partial J}{\partial d_m} + \frac{\partial J}{\partial d_\mu} & \text{if } d_m > 0 \text{ and } m \neq \mu \\ \sum_{\substack{m \neq \mu \\ d_m > 0}} \left(\frac{\partial J}{\partial d_m} - \frac{\partial J}{\partial d_\mu} \right) & \text{if } d_m > 0 \text{ and } m = \mu \end{cases} \quad (27)$$

Once the gradient of $J(d)$ and the descent direction D were computed, we start the process of updating d by using:

$$d \leftarrow d + \delta_{\max} D \quad (28)$$

Where δ_{\max} is the maximal admissible step size headed by D , and calculated by: $\delta_{\max} = -d_v / D_v$ (29)

$$\text{Where: } v = \arg \min_{\{m/D_m < 0\}} -d_m / D_m \quad (30)$$

If we replace the value of δ_{\max} in (28) we get

$$d \leftarrow d + (-d_v / D_v) \times D \quad (31)$$

The Eq. 31 in a matrix form is given by

$$\begin{bmatrix} d_1 \\ \vdots \\ d_v \\ \vdots \\ d_M \end{bmatrix} \leftarrow \begin{bmatrix} d_1 \\ \vdots \\ d_v \\ \vdots \\ d_M \end{bmatrix} + \begin{bmatrix} (-d_v / D_v) \times D_1 \\ \vdots \\ (-d_v / D_v) \times D_v \\ \vdots \\ (-d_v / D_v) \times D_M \end{bmatrix} \quad (32)$$

We can notice that component v of the vector d becomes null.

After the first update, now we compute $J^*(d)$ using an SVM solver with $K = \sum_m d_m^* K_m$, where $d^* = d + \delta_{\max} D$, and we check if the objective value decreases or not $\{J^*(d) < J(d)\}$. In the first iteration we set $J^*(d) = J(d)$. If the objective value decrease, d is updated using the formula $d = d + \delta_{\max} D$, which implies that $d_v = 0$ and we set $J(d) = J^*(d)$.

This procedure is repeated until $J^*(d)$ stops decreasing. To end up this process, we calculate an optimal step size δ by applying the golden search method on the interval between 0 and δ_{\max} , with an appropriate stopping criterion, such as the Armijo rule. Then the last adjustment is executed to compute the optimal value of d as $d = d + \delta D$.

Finally, the whole algorithm procedure is terminated when a stopping criterion is achieved. This stopping criterion can be either based on the duality gap, the KKT constraints, the variation of d between two consecutive steps, or, a maximal number of iterations. In this current study, we used MKL duality gap giving by:

$$\text{DyalGap} = \text{DualOne} - \text{DualTwo} \leq \varepsilon$$

$$\text{Where: } \begin{cases} \text{DualOne} = \max_m \sum_{i,j=1}^n \hat{\mu}_i \hat{\mu}_j y_i y_j K_m(x_i, x_j) \\ \text{DualTwo} = \sum_{i,j=1}^n \hat{\mu}_i \hat{\mu}_j y_i y_j \sum_m d_m^* K_m(x_i, x_j) \end{cases} \quad (33)$$

Consequently, we stop the process when $\text{DualGap} \leq \varepsilon$, where ε a tolerance threshold.

III. RESULTS

Tab.1 summarizes the performance obtained for SVM and MKL algorithms. In this study, two rhythms episodes of AF and NSR were extracted from Long-Term AF Database. SVM and MKL based-algorithms were optimized by using various configurations including the kernel function (Linear function, polynomial function, radial basis function), its corresponding adjustment coefficients, and the regularization parameter C. In order to get the optimal classification efficiency, setting of the aforementioned parameters must be conducted on a training dataset and validated on the test dataset. The best result for SVM was obtained by using a radial basis kernel function with σ equal to 10, and a C equal to 2400. However, for MKL the best selected configuration is obtained for C equal to 5800 and by using a combination of a teen radial basis kernel functions with σ equal 1 to 10. The data size used is represented by 8039 AF and 9251 NSR episodes are used in learning and validation; and 8039 AF and 9251 NSR episodes are used to test the models. From Tab.1, the MKL and SVM algorithms demonstrate excellent cardiac classification results. What makes this work very unique is the application of a multi-dynamic analysis of the heartbeat time series intervals. The coupling of four signals derived from R-peaks signal and applying fourteen linear and non-linear functions to feed the classification methods shown to be extremely effective.

TABLE I. PERFORMANCE MEASURES OF NSR AND AF DETECTION ALGORITHMS

	SVM		
	Sensitivity	Specificity	Positive predictivity
Learning	100%	100%	100%
Validation	99.70%	99.99%	99.99%
Test	96.54%	99.69%	99.62%
	MKL		
Learning	99.85%	99.96%	99.95%
Validation	99.56%	99.96%	99.95%
Test	95.47%	99.89%	99.87%

IV. CONCLUSION

In this paper, an effective automatic atrial fibrillation arrhythmia diagnosis was proposed, based on the combinations of multi-dynamics analysis of the QRS complex. Indeed, 14 linear and nonlinear functions were combined with four derivatives of R peaks signal to yield a set of 56 features. These features were used to discriminate the AF rhythm from NSR through two efficient algorithms namely SVM and MKL classifiers. Both methods have been proved to be successfully efficient to automatically detect an atrial fibrillation episode. The obtained results showed that MKL algorithm outperformed the SVM in terms of specificity and the positive predictivity. Therefore, these medical-oriented detectors can be of great importance to healthcare professional for AF diagnosis.

REFERENCES

- [1] S. Levy, G. Breithardt, R.W. Campbell, A.J. Camm, J.C. Daubert, M. Allessie, et al. Atrial fibrillation: current knowledge and recommendations for management. Working Group of Arrhythmias of the European Society of Cardiology, *European Heart Journal*, 19, (1998), 1294-320.
- [2] J.F. Pons, Z.Haddi, J.C. Deharo, A. Charaï, R. Bouchakour, M. Ouladsine & S. Delliaux. Heart rhythm characterization through induced physiological variables. *Scientific Reports*, 7, (2017), 5059.
- [3] Z. Haddi, J.F. Pons, S. Delliaux, B. Ananou, J.C Deharo, A. Charaï, R. Bouchakour, M. Ouladsine. A Robust Detection Method of Short Atrial Fibrillation Episodes, *Computing in Cardiology*, 44, (2017), 1-4.
- [4] N.A.H. Haldar, F.A. Khan, A. Ali, H. Abbas, Arrhythmia Classification using Mahalanobis Distance based Improved Fuzzy C-Means Clustering for Mobile Health Monitoring Systems, *Neurocomputing*, 220, (2017), 221-235.
- [5] S. Asgari, A. Mehrnia, M. Moussavi, Automatic detection of atrial fibrillation using stationary wavelet transform and support vector machine, *Computers in Biology and Medicine*, 60, (2015), 132-142.
- [6] K. Padmavathia, K.Sri Ramakrishnab, Classification of ECG Signal during Atrial Fibrillation Using Autoregressive Modeling, *Procedia Computer Science*, 46, (2015), 53-59.
- [7] R.J Martis, U. Rajendra Acharya, H. Prasad, C. Kuang Chua, C. Min Lim, Automated detection of atrial fibrillation using Bayesian paradigm, *Knowledge-Based Systems*, 54, (2013), 269-275.
- [8] R. Xiuhua, L. Changchun, L. Chengyu, W. Xinpei, L. Peng, Automatic detection of atrial fibrillation using R-R interval signal, *BMEI*, 2, (2011), 644 - 647.
- [9] L. S, X. Chen, J.K. Kanters, I.C. Solomon, K.H. Chon. Automatic selection of the threshold value R for approximate entropy. *IEEE Trans Biomed Eng*, 55, (2008), 1966-72.
- [10] S.M. Pincus, Approximate Entropy as a Measure of System Complexity, *Proc. Natl Academy Sci. USA*, 88, (1991), 2297-2301.
- [11] C. Weiting, W. Zhizhong, X. Hongbo, Y. Wangxin. Characterization of Surface EMG Signal Based on Fuzzy Entropy. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15, (2007).
- [12] J.S. Richman, J.R. Moorman, Physiological time-series analysis using approximate entropy and sample entropy, *American Journal of Physiology, Heart and Circulatory Physiology*, 278, (2000), 2039.
- [13] D.E. Lake, J.R. Moorman, 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*, 300 (2011), H319-25.
- [14] Z. Haddi, A. Amari, F. E. Annanouch, A. Ould Ali, N. El Bari, B. Bouchikhi, Potential of a Portable Electronic Nose for Control Quality of Moroccan traditional fresh cheeses, *Sensor Letters*, 9, (2011), 2229-2231.
- [15] A. Ben-Hur, D. Horn, H. Siegelmann, V. Vapnik, Support vector clustering, *Journal of Machine Learning Research*, (2), 125-137.
- [16] C. Cortes, V. Vapnik, Support Vector Networks, *Machine Learning*, 20, (1995), 273-297.
- [17] Boser, B. Guyon, I. Vapnik. V, A training algorithm for optimal margin classifiers, *Proceedings of the fifth annual workshop on Computational learning theory - COLT '92*, 144.
- [18] G. Lanckriet, N. Cristianini, P. Bartlett, and L.E. Ghaoui, Learning the kernel matrix with semi-definite programming, *J. Mach. Learn. Res*, 5, (2004), 27-72.
- [19] S. Sonnenburg, G. Rätsch, and C. Schäfer, A general and efficient multiple kernel learning algorithm. *NIPS*, (2006), 1273-1280.
- [20] M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien, lp-Norm multiple kernel learning, *J. Mach. Learn. Res*, 12, (2011), 953-997.
- [21] Z. Xu, R. Jin, S. Zhu, M. Lyu, and I. King, Smooth optimization for effective multiple kernel learning, in *Proc. AAAI Artif. Intell.*, 2010.
- [22] M. Kowalski, M. Szafranski, and L. Ralaivola, Multiple indefinite kernel learning with mixed norm regularization, in *Proc. 26th ICML*, Montreal, QC, Canada, (2009), 545-552.
- [23] A. Rakotomamonjy, F. Bach, S. Canu, Y. Grandvalet, SimpleMKL, *Journal of Machine Learning Research*, *Journal of Machine Learning Research*, 9, (2008), 2491.