CrossMark

ORIGINAL ARTICLE

Automatic detection of electrocardiographic arrhythmias by parallel continuous neural networks implemented in FPGA

Mariel Alfaro-Ponce[1] · Isaac Chairez[2] 🔵 · Ralph Etienne-Cummings[3]

Abstract In the developed world, heart diseases are the major cause of death among adults. Often, the sufferers of heart disease are not aware of their condition until a catastrophic medical event occurs. Therefore, early online detection and continuous monitoring of abnormal heart rhythms shall reduce this occurrence. There are four main types of arrhythmia: ventricular arrhythmia, supraventricular arrhythmia, premature beats and asynchronous arrhythmia. In this study, an algorithm for automatic detection of atrial premature contraction, supraventricular tachyarrhythmias, fusion of ventricular and normal beat (FUSION), isolated QRS-like artifact (ARFCT), ST change, T-wave change, premature or ectopic supraventricular beat and normal beat (NORMAL) using a continuous neural network (CoNN) is presented. This kind of continuous classifier offers an online detection of classical arrhythmia observed in electrocardiographic (EKG) signals. Typically, due to its complexity and recursive nature of arrhythmia classification algorithms, they are difficult to be implemented in real time. In this work, automatic signal classification was attained by implementing a parallel CoNN algorithm using fixed point arithmetic on a field-programmable gate array (FPGA). First, the classification algorithm using a floating-point MATLAB implementation was developed and validated. This procedure served as a benchmark for the fixed point FPGA implementation on a Xilinx Zinq board. The performance of the classification algorithm was evaluated by using a fivefold cross-validation method, achieving a 93.80% accuracy and a sensitivity (TPR) average of 98% when performing the classification of the entire set of EKG signal samples.

Keywords Signal classification · FPGA · Continuous neural networks · EKG arrhythmias · Parallel neural networks

1 Introduction

In North America and Europe, between 50 and 100 sudden unexpected cardiac failure deaths occur every year per a population of 100,000. This condition is even worse in Latin America. Heart diseases that lead to cardiac failure remain as the most important cause of death accounting for more than 17 million deaths per year [23]. These diseases produce abnormalities in the heart beat rhythm, or arrhythmias, which are caused by irregular (faster or slower) electrical impulses that coordinate the heart's normal operation [26]. Most of these abnormalities are not dangerous, but if they remain untreated, more complex arrhythmias may appear yielding to life threatening events.

Correct characterization of these atypical behaviors is important when trying to achieve the accurate classification of normal and abnormal heart activity. This fact is truly important considering the effect of correct and opportune diagnosis on the patient survival. Indeed, several studies have proven that significant progress has been made in the prevention of deaths resulting from the correct prediction of severe heart arrhythmias [15]. Since the nineteenth

🌀 Springer

century, electrocardiograph devices have allowed the recording of both regular and unusual electrocardiographic signals [10]. The constant evolution of clinical electrophysiology has provided detailed descriptions of the normal heart beat and some different types of abnormalities.

The emergence of modern technologies has transformed the way that physicians handle treatment and prevention of heart diseases. Therefore, automatic classification of EKG signals has received great attention given that the early detection of heart diseases can prolong life and enhance the life quality through appropriate treatment. Therefore, in recent years, there have been numerous research efforts analyzing the EKG signals by automatic signal processing algorithms. Indeed, there is a large number of publications documenting methods to classify certain arrhythmia in EKG signals [2, 20, 21, 31].

All these studies have used the fact that the state of cardiac health is generally reflected in the shape of the EKG waveform and heart beat rate. Indeed, it has been recognized that this waveform may contain important pointers to the nature of diseases afflicting the heart. The nature of the biological signals produced in the EKG waveform may impact the EKG at random times. Despite the variations of EKG waveform among different individuals, the signal alterations across patients often appear similar with the same type of arrhythmias. However, these variations may not show up all the time, but they can manifest at some specific period of time, thus appearing asynchronously.

These characteristics motivate the development of automatic waveform classifications to recognize the appearance of EKG abnormalities with high accuracy. This offers a reliable and affordable method to perform continuous mass screenings for cardiac abnormalities. The aim of this method is to obtain a set of characteristic shapes from the waveform that discriminate effectively one particular abnormal signal from another. Signal parameters, such as amplitude and phase, as well as signal shapes are typical characteristics used by the classification algorithms.

Over the years, several methods have been proposed to improve the accuracy and sensitivity of waveform classification strategies. These methods have included wavelet coefficient [7], autoregressive modeling [11] , radial base functions neural networks [19], self-organizing map, rough sets, fuzzy c-means clustering techniques [25] and many others [32]. These methods are implemented on personal computers but not on embedded hardware.

Software implementation uses the flexibility and full (float number) precision of powerful microprocessors with complex architecture algorithms; however, they are not portable. These computer algorithms have the disadvantage of being slower and with higher computational costs associated with operating system execution [21].

On the other hand, hardware implementation is limited in flexibility and usually offers lower precision, but has the advantage of processing data in parallel (as in field programming gate arrays or FPGA for short), making possible the implementation of complex algorithms in real time. In addition, they can be small and low power consuming, allowing them to be easily worn on the body.

Nowadays, FPGAs have increased their speed and circuit density making them suitable for signal processing in parallel. There are several reports describing the implementation of EKG parameter detection using FPGA devices [8]. Some of them have focused on the detection of certain sections of the EKG waveform [29], and others on the detection of abnormalities in the EKG wave [12], but most of them undergo preprocessing of the signal rather than working with the raw EKG [9]. Therefore, EKG signals classification demands some additional computer processes that may increase the computational complexity of the classifier. Furthermore, most of the waveform classifiers consider a time windowing process which limits the generalization capacity of signal classifiers because selecting the window size is still a major problem in designing pattern classifiers. Indeed, if the window size is not selected accurately, the classification accuracy can decrease significant even if the classifier methodology could be correct.

Classification problems occur when an object needs to be assigned into a predefined group or class based on a number of observed attributes related to that object. Artificial neural networks (NNs) provide an effective method to perform classification on continuous EKG signals. NN solutions are desirable because of their ability to self-adjusted as new information is provided to the system and their ability to approximate complex nonlinear functions. In the last decade, continuous neural network (CoNN), usually called differential NN, has emerged as a powerful tool to extend the classification capacities of systems that need this level of adaptability. Notice that continuous classification algorithms are not usual in the literature [4]. CoNN is a new type of NN described by a set of ordinary differential equations (ODEs). These ODEs are used to represent the set of signals monitored by the EKG amplifier. Generally, NN requires preliminary training stages. For instance, in conventional gradient descendant type, weight adaptation sensitivity of unknown systems is required during the so-called online training process [16]. This condition ensures the correct classification of signals into the different classes. By virtue of its parallel distribution, NN is generally robust, tolerant under the presence of faults and external noise, able to generalize well and capable of solving nonlinear problems [5].

In this paper, the implementation of CoNN as a real-time classifier for EKG signals is presented. The classifier

used the raw EKG signal that prevents losing information if some preliminary signal treatments are applied. Therefore, CoNN uses the EKG signal as input and a characteristic sigmoid function was selected as target that corresponds to signals where arrhythmia may appear. Therefore, a class of associative memory was constructed with the same CoNN structure but fixed weights obtained after applying a supervised training. An alternative option to use CoNN as signal classifier is to use the same EKG signal as the pattern that should be classified; therefore, a special class of continuous autoencoder system can be designed. To the authors knowledge, this is one of the first attempts to use EKG signal as it is to enforce the classification of arrhythmia presence. Moreover, the class of CoNN considered in this paper has not been used as signal classifier. The complete scheme for training as well as the classification was developed by applying the Lyapunov stability theory. This classifier based on a CoNN was also tested with both the software and hardware implementations to compare the results when working at full arithmetic precision and at lower precision. In particular, the hardware CoNN implementation was developed in a commercial FPGA Xilinx Zinq Board that also represented a contribution considering the previous attempts to implement CoNN in embedded systems.

In summary, the contributions of this paper are:

- The development of an automatic classifier for 8 different electrocardiography arrhythmia based on CoNN.
- A soft real-time implementation of the classifier on a field programming gate array.
- Detailed description on how to implement the training, generalization and validation processes for the pattern classifier based on CoNN.

1.1 Paper structure

The paper is organized as follows: Section 2 presents the mathematical support for the proposed classification algorithm which is based on the application of the CoNN implemented as a nonparametric identifier. Section 3 gives details on the EKG database used to test and validate the CoNN-based classifier. The fivefold cross-validation algorithm was considered to justify the application of the classification algorithm. Section 4 presents a comparison between the software and hardware CoNN implementations. Section 5 describes aspects of the FPGA evaluation board implementing the parallel CoNN signal classifier (VHDL code). Section 6 presents the results of the validation process of the CoNN classifier. Finally, the performance of the proposed classification CoNN on an FPGA

compared to other methods applied to the same database is described in Sect. 7.

2 CoNN as a classifier for EKG arrhythmia

Figure 1 shows a general analytic scheme of the CoNN-based classifier considered in this study. The first part performs a training algorithm associating the EKG signal with a predefined trajectory used to represent the specific class (among the N possible ones) where that particular EKG signal belongs. In a second stage, a set of N CoNN was fixed with weights obtained after training. This parallel structure, usually known as classifier ensemble, was used to evaluate/test the classifier performance. The selection of what signals were used in the training and testing phases was done according to the rules established in the k-fold cross-validation method [27]. The blocks labeled as LMSE describe the least mean square error calculus that are determined for the set of nine CoNN with fixed weights. These LMSE values are compared to select the minimum one (clearly this minimum value is attainable). This value defines the class where the tested signal belongs. This structure looks like a variation of the so-called associative memory concept [14].

2.1 CoNN online identifier

EKG signals are represented here as $u \in \mathbb{R}^m$. These signals are assumed to be continuous with respect to time. Therefore, the EKG signals classifier can be represented as

$$
\begin{aligned}
\frac{d}{dt}x^j(t) = {} & [A]^\top x^j(t) + \left[W_1^{0,j}\right]^\top \Psi_1(x^j(t)) \\
& + \left[W_2^{0,j}\right]^\top \Psi_2(x^j(t))u_i^j(t) + \xi(x(t),t) \\
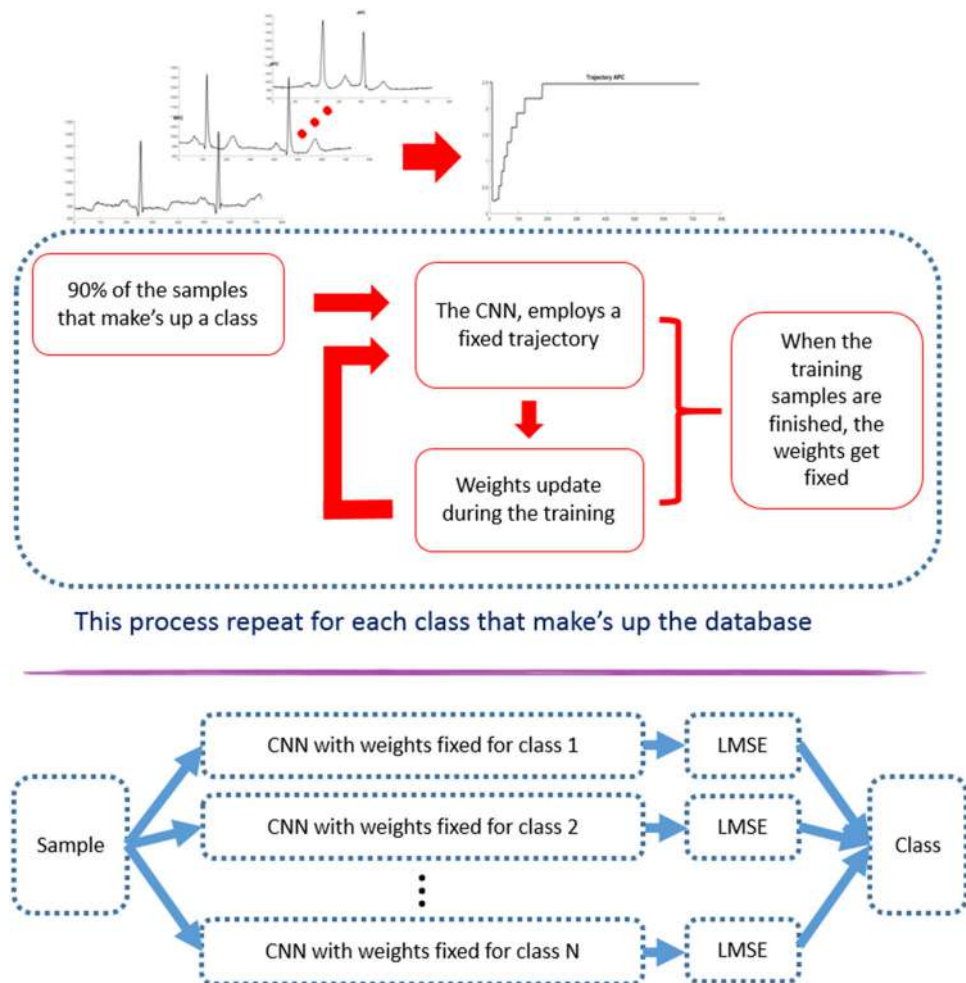& x(0) \text{fixed and bounded}
\end{aligned}
\tag{1}
$$

where $A \in \mathbb{R}^n, W_1^0 \in \mathbb{R}^{l_1}, W_2^0 \in \mathbb{R}^{l_2}$ are constant matrices used to approximate the EKG signals. The scalar $x \in \mathbb{R}$ defines a specific class characteristic to a particular EKG signal represented by u. The functions $\Psi_1 \in \mathbb{R}^{l_1}$ and $\Psi_2 \in \mathbb{R}^{l_2 \times m}$ define the set of activation functions used to design the NN structure. When the CoNN is used to classify the EKG signal using a reference function, to categorize the specific class where the EKG signal belongs, the class x was proposed to satisfy the structure of a sigmoid function, that is

$$
x^j(t) = \frac{a^j}{1 + e^{-c^j t}}
\tag{2}
$$

where a^j is the positive constant that characterizes the class and c^j is the positive constant to adjust the transition rate of the sigmoid function. This function represents the pattern

Fig. 1 CoNN process for the classification of EKG arrhythmias

to be approximated by the CoNN. Therefore, if the signal u_i^j belongs to the class j, then the corresponding signal x^j is characterized by both parameters, a^j and c^j. Clearly, if the type of arrhythmia changes during the training process, this couple of parameters also changes. For example, for ARFCT $a^1 = 1.5$, APC signals were characterized with $a^2 = 2.5$, FUSION beats $a^3 = 3$, for normal EKG signals, $a^4 = 4.5$; SVT used $a^5 = 5.0$. For ST change, $a^6 = 6.0$, SVPB $a^7 = 7.5$, T-wave change $a^8 = 9.0$, and finally, PVC signals were characterized with $a^9 = 10.0$.

An alternative option is using $x^j = N_i^{-1} \sum_{i=1}^{N_i} u_i^j$ where N_i is the number of EKG signals within each class (normal, APC, PVC and SVT). This particular form of applying CoNN is a variation of an autoencoder.

The term $\xi(x, t) : \mathbb{R}^{n+1} \to \mathbb{R}^n$ defines the approximation error of EKG signal associated with the finite number of activation functions used in the network structure. By assumption, the set of approximation errors represented by ξ satisfies the following definition: $|\xi| \le \xi_0 + \xi_1|x|$ with ξ_0

and ξ_1 positive scalars. Clearly, by including the term ξ_0, there is the possibility of including discontinuous functions. Therefore, uniqueness of solution must be understood in the sense of Filipov proposed in his seminal book. If $\xi_0 = 0$, then the uniqueness and existence of solution for the state x can be justified by the regular technique based on the application of the Lipchitz condition.

There are several theoretical results that justify the existence of finite constant matrices $W_1^{0,j}$, $W_2^{0,j}$ such that EKG signals can be presented as the NN structures presented in (1).

If different variations of EKG signals, represented by $u = u^j$, are considered, then different sets of matrices $W_i^{0,j}$ must exist, one per class of arrhythmia. Therefore, any feasible signal classifier must be capable of identifying the set of parameters $W_1^{0,j}$, $W_2^{0,j}$ when u^j is appearing as part of the EKG signal. The signal classifier proposed in this study is based on a set of parallel nonparametric identifier with similar structure to the approximation proposed in (1). This

condition is proposed to use the extended and well-established continuous parameter identification theory.

The set of CoNN used to perform the online signal classification satisfies the structure of the nonparametric identifier proposed in [28], that is

$$
\frac{d}{dt}\hat{x}^j(t) = [A]^\top \hat{x}^j(t) + \left[W_1^j(t)\right]^\top \Psi_1(\hat{x}^j(t))
$$
$$
+ \left[W_2^j(t)\right]^\top \Psi_2(\hat{x}^j(t))u^j(t) \tag{3}
$$
$$
\hat{x}^j(0)\text{fixed and bounded}
$$

where $W_1^j \in \mathbb{R}^{l_1}, W_2^j \in \mathbb{R}^{l_2}$. The vector $\hat{x}^j \in \mathbb{R}$ defines the state of class signal associated with the classifier. The set of time-varying weights, i.e., the matrices W_1^j and W_2^j, contains the adaptive parameters that are adjusted to produce the self-adaptive classification of the EKG signals in different classes. $\Psi_1 \in \mathbb{R}^{l_1}$ and $\Psi_2 \in \mathbb{R}^{l_2 \times m}$ define the set of activation functions used for the NN structure [28]. In this study, the activation functions were selected as sigmoid ones. That means that each component of these two vectorial functions was selected in that form. Usually, the correct adjustment of CoNN weights (W_i^j, $i = 1, 2$) provide the signal classification capacity. According to the results developed in [28], the so-called weight *updating* (learning) law is described by the following set of matrix differential equations

$$
\dot{W}_i^j(t) = -k_i^j P^j \Delta^j(t)\Psi_i^\top(\hat{x}(t)) - 2^{-1}k_i^j \tilde{W}_i^j(t), \quad i = 1, 2 \tag{4}
$$

The time-varying vectors $\Delta^j \in \mathbb{R}^n$, $\Delta^j = x^j - \hat{x}^j$ are the identification errors used during the training procedure. The parameters, k_i^j $i = 1, 2$, are the constant parameters used to adjust the learning rates. The time-varying terms $\tilde{W}_i^j$ represent the deviation from the time-varying weights W_i^j and $W_i^{0,j}$. The last constant parameters are obtained by a sequence off-line which is described in the next section.

Matrix P^j is the positive definite solution of the Riccati equations defined by

$$
2PA + P^j R^j P^j + Q^j = 0
$$
$$
R^j = W_1^{0,j}(\Lambda_2^j)^{-1}[W_1^{0,j}]^\top + W_2^{0,j}(\Lambda_4^j)^{-1}[W_2^{0,j}]^\top
$$
$$
+ \Lambda_1^j + \Lambda_3^j \tag{5}
$$
$$
Q^j = (\bar{f}_1^j + \lambda_{\max}(2\Lambda_2^j)l_1)I_{nxn} + Q_0^j
$$

Here $\Lambda_k^j \in \mathbb{R}^{n \times n}, k = 1, .., 4$ are positive definite matrices. The constant $\bar{f}_1^j$ represents the upper bound for the approximation error obtained as a consequence of fixing the number of weights in the CoNN structure. The matrices $W_1^{0,j}$ and $W_2^{0,j}$ are the initial conditions of weights used in the first signal within each class j. The matrix, Q_0^j, is a positive definite matrix used to provide the asymptotic

stability to the classifier structure. In fact, it must be selected (over a large set of possible values) just to ensure the existence of the solution for the Riccati equations. These results give the theoretical support to ensure that the CoNN algorithm may be used to classify the EKG signals. The classifier considers the evolution of the individual performance indices

$$
J^j(t = \mu T_k) := \int_{t=(\mu-1)T_k}^{t=\mu T_k} \left\|\Delta^j(t)\right\|^2 dt, \tag{6}
$$
$$
1 \le \mu \le N
$$

Therefore, the class $j^*(n)$ selected by the classifier is determined by $j^*(n) = j$ such that $J^{j*}(t = T) = \min_j J^j(t = T)$ is attained at each $t = nT_k$ where T_k is the size of the signal window to be analyzed and T is the total time of the signal considered to be studied. The selection of the period of time T_k is not a simple task because this value defines the number of samples that should be taken into account to obtain the classification. Some results can be consulted in the literature regarding how to select the best window size for different pattern recognition methods. Nevertheless, their application on the problem considered in this study is still a matter of investigation.

2.2 Off-line training of the CoNN

Learning laws proposed in (4) require the term $\tilde{W}_i^j(t)$ which depends on $W_i^{0,j}$. These constant values must be adjusted by preliminary off-line matrix identification algorithms. These algorithms use the least mean square method. To solve this part of the training procedure, a set of several EKG signals were selected to obtain $W_i^{0,j}$. Details of the matrix least mean square can be found in [1] where the convergence analysis as well as the identification algorithm is described.

The training process was completely developed on a personal computer implementing MATLAB 2014b. As usual in the k-fold cross-validation method, the number of folders was fixed to 5 to analyze the collected EKG signals that were used in this part of the off-line training method.

Two different numerical representations were evaluated: the first one used float variables (full precision) to represent all the weights, and the second one used 16-bit fixed point variables. This strategy was used for both the off-line and online training algorithms. We used off-line training produced weights to set the classifier connection matrix. On the other hand, the continuous training process proposed in (4) yields to the online classification of EKG signals.

As mentioned before, the weights $W_i^{0,j}$ contain the corresponding specific characteristic for each EKG class corresponding to normal and abnormal signals. To get the

weights that represent each class, 37 of the 74 samples of the ARFCT, 37 of the 75 samples of the APC, 37 of the 100 samples of the FUSION, 37 of the 100 samples of the NORMAL, 37 of the 75 samples of the SVT, 37 of the 100 samples of the STCH, 37 of the 100 samples of the SVPB, 37 of the 100 samples of the TCH and 37 of the 75 PVC were used in the off-line training process.

2.3 Validation of CoNN classification process

In order to validate the capacity of the CoNN to classify the EKG signals, the regular k-fold cross-validation (k-cv) method was implemented. Since the k-cv process is not an exhaustive cross-validation method, the data were split into (80/20) training and validation samples with some prede-fined ratio. Validation samples were used to estimate the prediction error. In the k-cv method, a set of sample S_n is uniformly and randomly partitioned into k-fold with similar sizes $\rho = \{\rho_1, ..., \rho_k\}$.

Let $S_i = S_n \backslash \rho_i$ be the complement data set of ρ_i. Then, all the samples belonging to S_i are used as training data. This particular cross-validation procedure is repeated k times (the number of folds), with each of the k subsamples used exactly once as the validation data. The complete set of k results obtained from the folds is averaged to produce a single estimation.

Indeed, the cross-validation algorithm $G(\cdot)$ induces a classifier from S_i, defined by $\psi_i = G(S_i)$, where the pre-diction error based on ρ_i is estimated. So the general pre-diction error $\hat{\epsilon}_k(S_n, \rho)$ based on the classifier $\psi = G(S_n)$ that uses the entire set of samples is estimated as follows [27]

$$\hat{\epsilon}_k(S_n, \rho) = \frac{1}{n} \sum_{i=1}^{k} \sum_{(x,c) \in \rho_i} 1(c, \psi_i(x))$$

where n is multiple of k, $1(i,j) = 1$ if $i \neq j$ and zero otherwise. This error is the average of errors committed by ψ_i in their corresponding partitions ρ_i. In particular, the so-called fivefold cross-validation was used to perform the validation procedure of the signal classifier proposed in this study. An additional general validation measure was per-formed by using a modified version of signal-to-noise-plus-distortion ratio (SNDR). This measure was following the next mathematical structure for each class

$$Q^j = 20 \log \left(\left[J^j(t = \mu T) \right]^{-1} \int\limits_{t=(\mu-1)T_k}^{t=\mu T_k} \left\| \Delta^j(t) \right\|^2 \mathrm{d}t, \right) \quad (7)$$

SNDR introduces a normalized method to characterize the classification accuracy. This measure can be considered an amplitude independent quality measure. So, their results can be trusted more than the one presented in (6).

3 Database description

To test the CoNN classification capabilities, the MIT-BIH Arrhythmia database and the European ST-T database were elected as information sources. Due to their availability and full detailed description of each beat, both databases are recommended in the American National Standard for Testing and reporting performance results of cardiac rhythm and ST segment. The MIT-BIH Arrhythmia data-base is described in [13, 24], and the European ST-T database is described in [13, 30]. Both databases have been used on several occasions by different research groups [6, 22] which allows their results for the comparison of the performances of the proposed classification algorithm based on CoNN.

The MIT-BIH Arrhythmia database contains 48 half-hour records of ambulatory EKG recordings obtained from 47 subjects. The EKG data were digitized at 360 samples per second per channel with an 11-bit resolution over a 10 mV range. Two or more cardiologists independently made annotations on each record. Disagreements were resolved to obtain the computer-readable reference annotations for each beat (approximately 110,000 annotations in all) included with the database.

The European ST-T database consists of 90 annotated excerpts of ambulatory ECG recordings from 79 subjects. Each record is two hours in duration, sampled at 250 samples per second with 12-bit resolution over a nominal 20 mV input range. Annotations were made by two car-diologists that worked independently to annotate each record beat by beat.

To test the proposed algorithm, eight types of arrhyth-mia (ARFCT, APC, FUSION, SVT, STCH, SVPB, TCH and PVC) and normal EKG rhythm samples were taken from both databases. For training and validation, only the fragments of signals that contain the arrhythmia episodes and normal EKG were considered. These signals are fil-tered by the regular analog filters sequence. No additional digital filters applied over the EKG signals.

In order to evaluate the classification capacities of the algorithm presented in this study, a set of evaluation tests was proposed. A set of 799 signals was prepared artificially according to the following procedure: consider a first signal S_i from a class C_i and a second one S_j from a class C_j, the hybrid signal S_{ij} was generated as the convex combination of $S_{ij} = \lambda S_i + (1 - \lambda S_j)$. Then, these signals were tested on the parallel arrangement of trained CoNN. For this particular analysis, $\lambda = 0.8$ meaning that 80% of a signal belong to a class C_i was combined with a 20% of a single belonging to a class C_j. A second round of analysis considered the analysis of borderline signals using $\lambda = 0.7$. After the total evalua-tion of classification process, the CoNN-based classifier

achieved a 93.8% of correct classification for both databases. This result shows the detectability capacity of the classifier proposed in this study.

To test the classification capabilities of the proposed identifier, the negative prediction value (NPV), positive prediction value (PPV), true positive rate (TPR) and specificity (SPC), where calculated in order to obtain the confusion matrix of the classifier. These values are calculated according to the following equations:

$$\text{NPV}(\%) = \frac{\text{TrueNegatives}}{(\text{TruePositives} + \text{TrueNegatives})} \times 100$$

$$\text{PPV}(\%) = \frac{\text{TruePositives}}{(\text{FalsePositives} + \text{TruePositives})} \times 100$$

$$\text{TPR}(\%) = \frac{\text{TruePositives}}{(\text{TruePositives} + \text{FalseNegative})} \times 100$$
(8)

$$\text{SPC}(\%) = \frac{\text{TrueNegatives}}{(\text{TrueNegatives} + \text{FalsePositive})} \times 100$$

where true positives are the total number of signals that were correctly classified in the corresponding class, true negatives are the total number of signals that were not in the class evaluated and they were clearly not classified in it, false negatives are the total number of signals belonging to the class evaluated, but they were considered as part of a different class, and false positives correspond to the number of signals that belong to a different class, but they were included in the evaluated class. The confusion matrix allows a more detailed analysis of classification capacities for the CoNN-based signal classifier.

We developed the study for all the databases described in the manuscript. We only explained the description of one single example just to avoid unnecessary repetition of the same method. However, all the databases were analyzed in the same way.

4 Fixed point implementation of the classifier

The CoNN algorithm was initially developed in full precision (based on floating-point variables). A second analysis used the same model but was implemented in lower precision (16-bit fixed point variables). Table 1 shows the nature of weight signals employed for both the full and lower precision. The notation $\langle \pm, \varpi_1, \varpi_2 \rangle$ means signed float represented by ϖ_1 integer digits with fixed number ϖ_2 of fractional ciphers.

Each CoNN was evaluated numerically during the training phase using the following set of parameters $k_1 =$

Table 1 CoNN input values for software and hardware

Inputs	Floating P	Fixed P
Sample	Double	$\langle \pm \rangle$
W1 ARFCT	Double	$\langle \pm, 16.14 \rangle$
W2 ARFCT	Double	$\langle \pm, 16.13 \rangle$
W1 APC	Double	$\langle \pm, 16.14 \rangle$
W2 APC	Double	$\langle \pm, 16.13 \rangle$
W1 FUSION	Double	$\langle \pm, 16.14 \rangle$
W2 FUSION	Double	$\langle \pm, 16.13 \rangle$
W1 NORMAL	Double	$\langle \pm, 16.13 \rangle$
W2 NORMAL	Double	$\langle \pm, 16.12 \rangle$
W1 SVT	Double	$\langle \pm, 16.13 \rangle$
W2 SVT	Double	$\langle \pm, 16.12 \rangle$
W1 STCH	Double	$\langle \pm, 16.12 \rangle$
W2 STCH	Double	$\langle \pm, 16.11 \rangle$
W1 SVPB	Double	$\langle \pm, 16.12 \rangle$
W2 SVPB	Double	$\langle \pm, 16.9 \rangle$
W1 SVPB	Double	$\langle \pm, 16.11 \rangle$
W2 SVPB	Double	$\langle \pm, 16.7 \rangle$
W1 TCH	Double	$\langle \pm, 16.11 \rangle$
W2 TCH	Double	$\langle \pm, 16.5 \rangle$
W1 PVC	Double	$\langle \pm, 16.11 \rangle$
W2 PVC	Double	$\langle \pm, 16.4 \rangle$

The meaning of $P = \langle \pm I.F \rangle$ represents the number of positive/negative () digits used to represent the integer (I) and floating (F) sections of numbers used in the FPGA evaluation of the classifier proposed in this study

$0.5, k_2 = 0.7, A = -2.2, P^1 = P^2 = P^3 = P^4 = P^5 = P^6 = P^7 = P^8 = P^9 = 3.3$, and the weights were initialized with the following values $W_1(0) = 0.01$ and $W_2(0) = 0.01$. The same set of parameters was used in either evaluation: in the personal computer or the embedded system. For all the experiments, the analysis time T was selected as 2.16 s.

The lower-precision representations presented in Table 1 were chosen to keep the off-line classification quality obtained when the float representation was considered. One may notice that this table shows the actual simplification obtained when the classifier is implemented in an embedded system instead of the personal computer. Therefore, it is expect to obtain an efficient pattern classifier based on CoNN not only from the effectiveness of classification but the computational cost also. Different lower-precision representations were chosen in order to satisfy the classification purposes described above.

A simplified version of the algorithms used to evaluate both the training and validation processes as well as the implementation in the FPGA device is defined as follows:

Algorithm 1: Training process

Data: Load the database sample and the X trajectory
Result: Training weights $W_1^j(t)$ and $W_2^j(t)$ for each class j
Initialization: Set the values for A, K_1, K_2, P and initial values for W_1 and W_2;
for $1 \leq j \leq$ *Number of classes* **do**
    for $1 \leq i \leq X_{Tr,i}^j$ **do**
        • Generate the values of parameters in the activation functions σ_i^j and ϕ_i^j for sample $S_{Tr,i}^j$
        • Using the class information $x_i^j(t)$ assigned to $S_{Tr,i}^j$, determine $\Delta_i^j(t) = x_i^j(t) - \hat{x}_i^j(t)$
        • Solve the learning laws for $W_{1,i}^j(t)$ and $W_{2,i}^j(t)$ based on $\Delta_i^j(t)$
        • Solve the estimation of dynamic class $\hat{x}_{i+1}^j$
    end
    Save the values of $W_1^j(t)$ and $W_2^j(t)$
end

Algorithm 2: Validation process

Data: Load the database sample and final weights $W_1^j(t)$ and $W_2^j(t)$ for each class j
Result: Classification efficiency for each class j
Initialization: Set the values for A, K_1, K_2, P and values for W_1^j and W_2^j obtained after the training process;
Separate the set of signals in the database accordingly to the rules of the k-fold cross validation. For each stage in the method, execute the following procedure: **for** $1 \leq i \leq X_{Tr,i}^j$ **do**
    for $1 \leq j \leq$ *Number of classes* **do**
        • Generate the values of parameters in the activation functions σ_i^j and ϕ_i^j for sample $S_{Tr,i}^j$
        • Using the EKG register information $u_i^j(t)$ assigned to $S_{Tr,i}^j$, determine $\hat{x}_i^j(t)$
        • Evaluate $\Delta_i^j(t)$ integrated in a time window form $T - h$ to T where h is the length of EKG register
    end
    Select the minimum value of $\Delta_i^j(t)$ with respect to j Assign the class accordingly to the specific class j corresponding to the minimum value calculated above
end

EKG signals were repeatedly tested in the classifier presented in (3). The training process used the set of signals according to the distribution presented in the fivefold cross-validation method available in MATLAB. The weights obtained in each training sequence was recorded and used in the next trial. With the whole set of EKG signals considered as training patterns, the final set of weights was used to perform the validation stage. These final trained weights were specified according to the class of abnormality of each EKG signal. One may notice that after the training process, the EKG signals belonging to the training set were reproduced by the detector with an accuracy of 99% Fig. 2. The signals showed in this figure are just one example of how the CoNN implemented during the training process can track the EKG signals.

Two sets of training processes were executed: the first one used the set of weights produced when full precision numbers were used to execute the training process, while the second one applied the set of weights represented by fixed numbers.

These two sets of training processes were simulated on a personal computer. Both training processes reproduced the EKG signals with the same quality (Fig. 3). The similar performance of both types of classifiers (with full and low precision numbers) justifies the possibility of implementing the signal classifier in an embedded system with fixed point precision, such as the FPGA system.

The mean square error of each CoNN was evaluated for each class of arrhythmia, and the same evaluation was done to compare the performances between the identifiers running on either floating or fixed point representations. Similar performances were confirmed using the mean square error information. Also, a transient period not longer than 0.5 s was produced in all the training evaluations.

5 FPGA implementation

To implement the signal classifier in the embedded FPGA system, a fixed point representation of both the detector state and the identification error was developed. Table 2
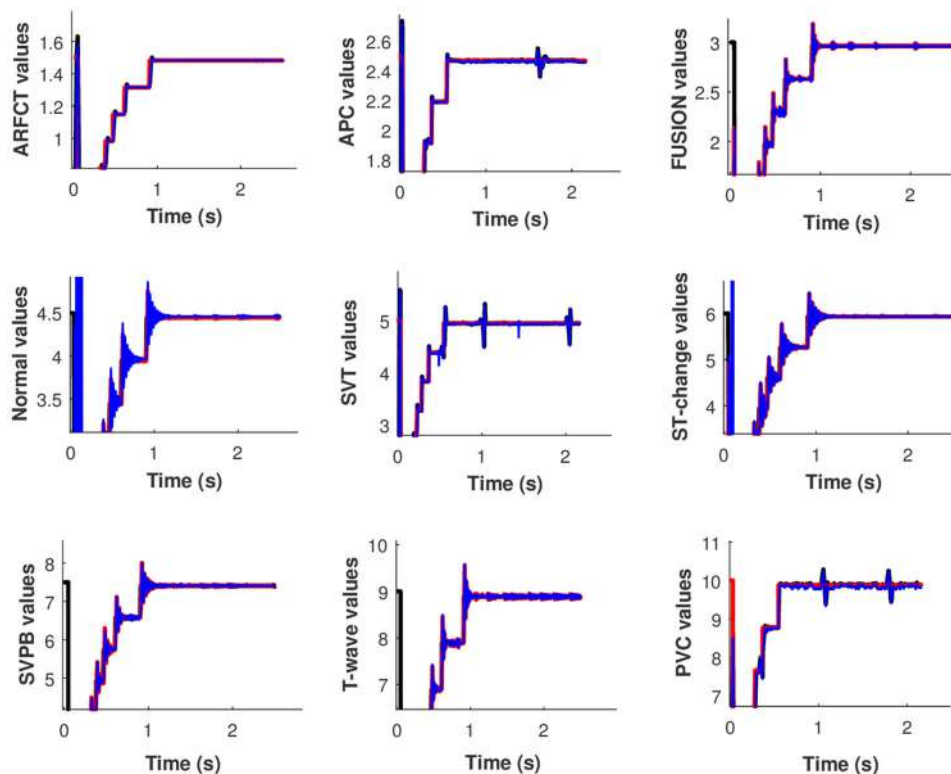
Fig. 2 Performance of the CoNN for ARFCT, FUSION, SVT, ST change, SVPB, T-wave, PVC arrhythmias and normal EKG waveform, when classification sample is correct. The *black line* is the desired output, the *red line* is the full precision NN output, and the *blue line* is the low-precision NN output

shows the values employed to implement the low precision representation of the detector. The second column in this table demonstrates that all variables executed in MATLAB were used as floating-point variables. Nevertheless, the third column demonstrates how the variable representation was simplified to a signed ($\pm$) fixed point variable with the 16 bits to represent the integer part and variable number of bits (showed in the third column of the same table after the dot symbol). This analysis was necessary to attain the FPGA implementation.

The implementation of the classification algorithm in the FPGA system obeyed the following algorithm:

The detector implemented in the FPGA system was compared online with a similar CoNN structure implemented on a personal computer with double representation of numbers. This comparison was feasible considering the online Ethernet monitoring system of the Xilinx Zedboard FPGA development board.

The CoNN identifier was implemented on a Xilinx Zedboard development board. This board is based on the Zynq^{TM}-7000 AP SoC xc7z020-CLG484-1 processor. The memory resources of this development board include a 512MB DDR3, 256Mb Quad-SPI Flash and a 4GB external

---

Algorithm 3: Implementation process

Data: Load the database sample and final weights $W_1^j(t)$ and $W_2^j(t)$ for each class j
Result: Classification efficiency for each class j evaluated in embedded devices

- Initialization: Set the values for A, K_1, K_2, P and values for W_1^j and W_2^j and update them into the FPGA system by using the *targetupdater* to setup the *Zynq hardware*;
- Select the code generation target (in the platform section, select the FPGA evaluation kit)
- Check the classifier design and convert it to fixed point equivalent numerical simulation
- Set the target interface (AXI-4 Lite interface)
- Build the project which can be executed into the FPGA system
- Transfer the transformed code into the target device (FPGA)
- Run the numerical simulation in external mode by transferring each signal into the FPGA device which is executing the classifier with fixed weights
- Evaluate the classification process accordingly to the rules derived from the k-fold cross validation process

Fig. 3 Closer view of the performance of the neural network for APC (*top*, *left-hand side*), STVA (*top*, *right-hand side*), PVC (*bottom*, *right-hand side*) arrhythmias and normal EKG waveform (*bottom*, *left-hand side*). On *x*-axis, samples refers to the number of acquired EKG measurements
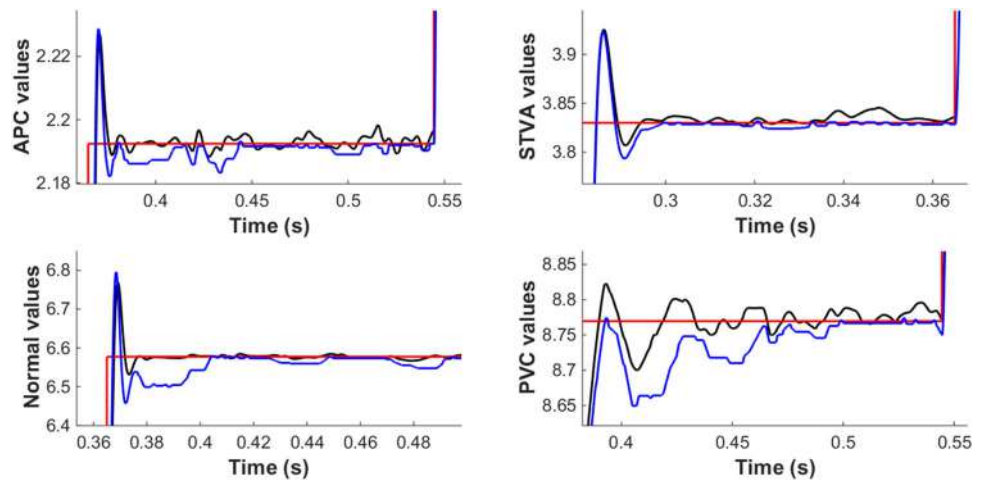

Table 2 CoNN output values for software and hardware

Outputs	Floating P	Fixed P
Xe ARFCT	Double	$\langle\pm,16.13\rangle$
Delta ARFCT	Double	$\langle\pm,16.4\rangle$
Xe APC	Double	$\langle\pm,16.13\rangle$
Delta APC	Double	$\langle\pm,16.4\rangle$
Xe FUSION	Double	$\langle\pm16.12\rangle$
Delta FUSION	Double	$\langle\pm,16.4\rangle$
Xe NORMAL	Double	$\langle\pm16.12\rangle$
Delta NORMAL	Double	$\langle\pm,16.4\rangle$
Xe SVT	Double	$\langle\pm16.11\rangle$
Delta SVT	Double	$\langle\pm,16.4\rangle$
Xe STCH	Double	$\langle\pm,16.11\rangle$
Delta STCH	Double	$\langle\pm,16.4\rangle$
Xe SVPB	Double	$\langle\pm,16.10\rangle$
Delta SVPB	Double	$\langle\pm,16.4\rangle$
Xe TCH	Double	$\langle\pm,16.10\rangle$
Delta TCH	Double	$\langle\pm,16.4\rangle$
Xe PVC	Double	$\langle\pm,16.10\rangle$
Delta PVC	Double	$\langle\pm,16.4\rangle$

Table 3 Summary of registers used in the FPGA device

Type of register	Used	Total
Slice registers	471	10,6400
Slice LUTS	50284	53,200
Slice LUT–flip-flop pairs	50291	53,200

All the EKG signals from the database were evaluated using the training algorithm in the FPGA system. They were done according to the same distribution characterized by the tenfold cross-validation algorithm. The identifier evaluation for each EKG signal was simulated in the FPGA system spending approximately 2.16s each.

6 Performance quality measurements

The full database was tested on the FPGA. Table 4 summarizes the training results obtained from the FPGA system. One would notice that a similar correct classification percentage was achieved when float number precision was used. This confirms the possibility of success when using the detector in the selected, low-precision embedded system.

In order to evaluate the classification capacities of the algorithm implemented in the FPGA for this study, a set of evaluation tests were proposed. A set of 300 signals was prepared artificially according to the following procedure: consider a first signal S_i from a class C_i and a second one S_j from a class C_j, the hybrid signal S_{ij} was generated as the convex combination of $S_{ij} = \lambda S_i + (1 - \lambda S_j)$. Then, these signals were tested on the parallel arrangement of trained CoNN. In particular, for this particular analysis, $\lambda = 0.7$ meaning that 70% of a signal belong to a class $C_i$was combined with a 30% of a single belonging to a class C_j,

SD card. The communication systems of this board include the USB-JTAG programming, 10/100/1000 Ethernet, USB OTG 2.0 and USB-UART.

The programming language used to implement the 16-bit fixed point representation of CoNN was VHDL. The corresponding code was generated using the MATLAB HDL Coder 3.5 and ISE Xilinx 14.4. An auxiliary AXI-4 interface was implemented to perform the Ethernet communication between the computer and the development board to follow online the time evolution of the CoNN-based classifier. Table 3 shows the number of total registers used to implement the identifier.

Table 4 Training and validation results for float number precision

	ARFCT	APC	FUSION	NORMAL	SVT	STCH	SVPB	TCH	PVC
Samples	74	75	100	100	75	100	100	100	75
Training	15	15	20	20	15	20	20	20	15
TPR	100%	100%	98.96%	98.98%	97.26%	95.6%	92.31%	99%	100%
SPC	75%	100%	33.33%	100%	100%	66.67%	88.89%	25%	50%
PPV	98.59%	100%	97.95%	100%	100%	96.7%	98.82%	96.9%	98.64%
NPV	100%	100%	50%	66.67%	50%	60%	53.33%	50%	100%

the CoNN-based classifier implemented in the FPGA achieved a 72.3% of correct classification for the database. This result shows the detectability capacity of the classifier proposed in this study.

Tables 5 contains the results of NPV, PPV, TPR and SPC when the evaluation of classifier was executed over the set of borderline signals considering that $\lambda = 0.8$. According to the accuracy results as well as the predictive analysis (with all results above 90%), the classifier proposed in this study seems to be a reliable method to classify certain characteristics in EEG signals.

Figure 4 depicts the comparison between the CoNN identifier evaluated on a PC (Processor i7, 8 GB RAM and 64-bit operating system) and that on the Xilinx FPGA Zedboard evaluation system. The signal classifier was implemented as a composition of four parallel identifiers with fixed set of weights. These weights were recovered from the files recorded during the training process executed on the PC system.

To perform a continuous evaluation of the CoNN implement of parallel classifier, an artificial EKG signal was developed. This signal, which was not in the database, used partial information of signals obtained from the database. The signal contained data from the EKG signals in the following order: PVC, normal, APC and SVT.

The results of the classification system based on the parallel structure of CoNN detector with fixed weights are detailed in Fig. 4. The first signal showed at the top of the figure demonstrates the signal constructed artificially. It is clear that the error obtained in each classifier with fixed

weights is less when the corresponding part of the signal contains the corresponding type of signal.

This part of the classifier evaluation was only tested in the FPGA development board. The red arrow points to the segments where the corresponding classifier represents the corresponding type of arrhythmia. The blue lines show the time when the artificial signal changes from one type of arrhythmia to the other. This evaluation also details the result of the continuous nature of the classifier proposed in this study.

The classifier proposed in this study also considers a different type of evaluation. It is not unusual for an incorrect signal to appear in the electrophysiological signal due to an incorrect measuring technique. It is even possible for a different bio-potential to appear when acquiring the EKG signal. Therefore, the classifier was tested on the PC and the FPGA with an electromyographic signal as the input.

In all the cases, the error grew to a point beyond a predefined threshold (2.0), which indicated that the input signal was not one of the expected waveforms. This kind of evaluation is not typically done by other researchers. Finally, a comparison between the results presented in this study and others introduced in different previous studies is presented. Only the arrhythmia classification implemented on an FPGA is presented in Table 5. This comparison is valid because the closeness between the classifier evaluated on the PC and the FPGA has already been proven.

Even when other work do not consider the same type of arrhythmias for classification purposes, most of them take into account PVC as part of the classifier evaluation. The parallel CoNN classifier evaluated on the FPGA showed a better performance than previous results (Table 6). The described algorithm attained a total correct classification of 97.62% for all the different signals included in the database.

In order to elucidate the training process efficiency, the quality measurement introduced in Eq. (7) was evaluated. This value was obtained during the training as well as after the testing procedures. Table 7 shows these values obtained in both phases of the classification process.

The training process took 300 s long, while the validation was 60 s long. This is a very long period for training and validation. This is an opportunity for developing more efficient numerical methods to solve the pattern classification problem by the method proposed in this study. On

Table 5 NPV, PPV, TPR and SPC results for the borderline signals with $\lambda = 0.8$ I

	$A^*_{70\%}\_P^*_{30\%}$	$P^*_{70\%}\_S^*_{30\%}$	$S^*_{70\%}\_N_{30\%}$	$N_{70\%}\_A^*_{30\%}$
NPV	78.57%	75.71%	74.24%	84.5%
PPV	21.42%	24.28%	25.75%	15.49%
TPR	94.82%	96.36%	89.09%	96.77%
SPC	88.23%	85%	85%	84.61%

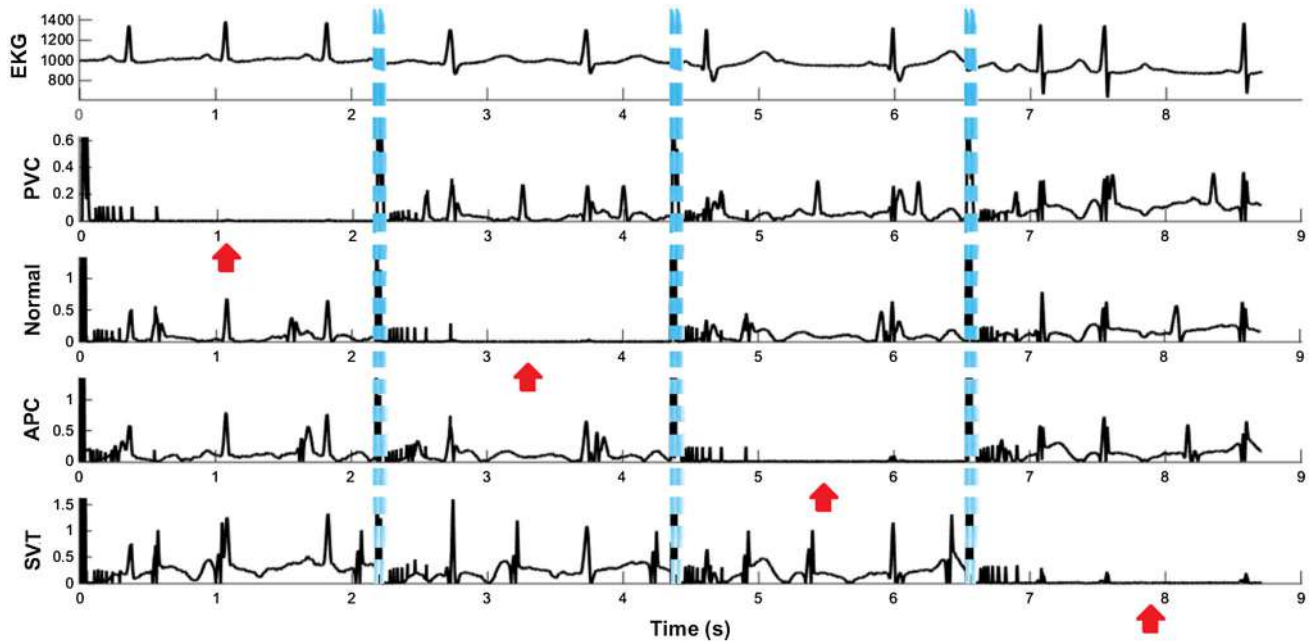* A, APC

* P, PVC

* S, STV

* N, Normal

Fig. 4 From the *top* to the *bottom*, a EKG artificial signal generated from patterns containing arrhythmias and normal EKG waveform. The second trace is the error for detection of PVC events. The third trace is the error for detection of normal EKG rhythms. Next is the error for detection of APC events, and final is the error trace for detection of STV

Table 6 Comparison of several classifier performances on MIT-BIH database

	FPGA	Type	Accuracy (%)
Cvikl et al. [8]	Virtex-II	PVC	92.36
Chang et al. [7]	Virtex- 4	PVC	94.73
Jiang et al. [18]	Virtex-II	VEB	98.1
		SVEB	96.6
Jewajinda et al. [17]	Virtex- 5	VEB	97.7
		SVEB	96.5
Armato et al. [3]	Virtex- 4	PVC	95.43
This study	Zedboard[a]	APC	98.6
		SVT	94.6
		PVC	97.3
		N	99.99

VEB ventricular ectopic beats, *SVEB* supraventricular ectopic beats), *N* Normal

[a] For the full description, refer to Sect. 5

Table 7 Signal-to-noise-plus-distortion ratio applied to the DNN output

	Trajectory floating	Trajectory fixed	Floating/fixed
ARFCT	dB	dB	dB
APC	98.64dB	90dB	198dB
FUSION	dB	dB	dB
Normal	136.93dB	117dB	178dB
SVT	111.3dB	88.4dB	167dB
STCH	dB	dB	dB
SVPB	dB	dB	dB
TCH	dB	dB	dB
PVC	167.22dB	134.4dB	195dB

the other hand, the processing time in the FPGA system was 3.4 s which is closer to the actual period selected for each signal.

7 Conclusions

From the clinical point of view, automatic online detection of different types of EKG arrhythmia may contribute to immediate and better treatment of different heart diseases.

This study proposes a method to perform EKG signal classifications online based on the application of parallel CoNN. This structure was first trained on a PC simulation system to prepare a set of weights which were used to design the particular CoNN used to detect multiple different arrhythmia waveforms in parallel.

The classifier was then implemented on an FPGA evaluation board and was validated using the fivefold cross-validation method. The classification TPR attained by the classifier developed in this study was the best (98.01%) among a selected set of similar classifiers, also implemented in FPGA, described previously in some similar studies. This value was obtained as the average of the accuracy percentage obtained for each class considered in this study.

The classification technique developed in this study can be extended to different problems such as (a) electroencephalographic signal analysis that can be executed online. This application can be used in the implementation of brain machine interfaces focused on assisting patients with movement limitations, (b) electromyographic pattern classifications that can used as input information in the control loop of active orthosis that can used in different sections of the human body or (c) online voice pattern classifier that can be used in security and medical devices among others.

Compliance with ethical standards

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Alfaro-Ponce M, Arguelles-Cruz A, Chairez I (2014) Adaptive identifier for uncertain complex nonlinear system based on continuous neural network. IEEE Trans Neural Netw Learn Syst 25(3):483–494
2. Alickovic E, Subasi A (2015) Effect of multiscale PCA denoising in ECG beat classification for diagnosis of cardiovascular diseases. Circuits Syst Signal Process 34(2):513–533
3. Armato A, Nardini E, Lanata A, Valenza G, Mancuso C, Scilingo E, De Rossi D (2009) An fpga based arrhythmia recognition system for wearable applications. In: Ninth international conference on intelligent systems design and applications, 2009. ISDA '09. IEEE, pp. 660–664
4. Arqub OA, Abo-Hammour Z (2014) Numerical solution of systems of second-order boundary value problems using continuous genetic algorithm. Inf Sci 279:396–415
5. Benvenuto N, Piazza F (1992) On the complex backpropagation algorithm. IEEE Trans Signal Process 40(4):967–969
6. Castillo O, Melin P, Ramrez E, Soria J (2012) Hybrid intelligent system for cardiac arrhythmia classification with fuzzy k-nearest neighbors and neural networks combined with a fuzzy system. Expert Syst Appl 39:2947–2955
7. Chang C, Lin C, Wei M, Lin K, Chen S (2014) High-precision real-time premature ventricular contraction (pvc) detection system based on wavelet transform. J Signal Process Syst 77(3):289–296
8. Cvikl M, Zemva A (2010) FPGA-oriented HW/SW implementation of ecg beat detection and classification algorithm. Digit Signal Process 20(1):238–248
9. Faezipour M, Saeed A, Bulusu S, Nourani M (2010) A patient-adaptive profiling scheme for ecg beat classification. IEEE Trans Inf Technol Biomed 14(5):1153–1165
10. Fay L, Misra V, Sarpeshkar R (2009) A micropower electrocardiogram amplifier. IEEE Trans Biomed Circuits Syst 3(5):312–320
11. Ge D, Srinivasan N, Krishnan SM (2007) The application of autoregressive modeling in cardiac arrhythmia classification. In: Advances in cardiac signal processing, ch. 8. Springer, Berlin, pp. 209–224
12. Ghorbanian P, Ghaffari A, Jalali A, Nataraj C (2010) Heart arrhythmia detection using continuous wavelet transform and principal component analysis with neural network classifier. Comput Cardiol 2010:669–672
13. Goldberger AL, Amaral L, Glass L, Hausdorff J, Ivanov P, Mark R, Mietus G, Moody JE, Peng C-K, Stanley H (2000) Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. Circulation 101(23):e215–e220
14. Hinton G, Anderson J (1989) Parallel models of associative memory. In: Hinton G, Anderson J (eds). Lawrence Erlbaum Associates
15. Huikuri H, Mkikallio T, Raatikainen P, Perkimki J, Castellanos A, Myerburg R (2003) Prediction of sudden cardiac death: appraisal of the studies and methods assessing the risk of sudden arrhythmic death. Circulation 108:110–115
16. Hwang HJ, Kim S, Choi S, Im CH (2013) Eeg-based brain-computer interfaces: a thorough literature survey. Int J Brain Comput Interact 29(12):814–826
17. Jewajinda Y, Chongstitvatana P (2013) A parallel genetic algorithm for adaptive hardware and its application to ecg signal classification. Neural Comput Appl 22:1609–1626
18. Jiang W, Kong S (2007) Block-based neural networks for personalized ecg signal classification. IEEE Trans Neural Netw 18(6):1750–1761
19. Kannathal N, Rajendra U, ChooMin L, Suri JS (2007) Classification of cardiac patient states using artificial neural networks. In: Advances in cardiac signal processing, ch. 7. Springer, Berlin, pp. 187–208
20. Khalaf A, Owis M, Yassine I (2015) A novel technique for cardiac arrhythmia classification using spectral correlation and support vector machines. Expert Syst Appl 42:8361–8368
21. Khazaeea A, Ebrahimzadeha A (2013) Heart arrhythmia detection using support vector machines. Intell Autom Soft Comput 19(1):1–9
22. Lawrence S, Burns I, Back A, Tsoi C, Giles L (2012) Neural network classification and prior class probabilities. In: Neural networks: tricks of the trade, 2nd ed, ch. 14. Springer, pp. 295–309
23. Mendis S, Puska P, Norrving B (2011) Global Atlas on cardiovascular disease prevention and control. In: Mendis S, Puska P, Norrving B (eds) World Health Organization in collaboration with the World Heart Federation and the World Stroke Organization. ISBN 978 92 4 156437 3
24. Moody G, Mark R (2001) The impact of the mit-bih arrhythmia database. IEEE Eng Med Biol Mag 20(3):45–50
25. Ozbay Y, Ceylan R, Karlik B (2005) A fuzzy clustering neural network architecture for classification of ecg arrhythmias. Comput Biol Med 26(2006):376–388
26. Perna F, Leo M (2012) Epidemiology, classification and description of cardiac arrhythmias. In: Sports cardiology, ch. 14. Springer, Milano, pp. 155–177
27. Rodriguez J, Perez A, Lozano J (2010) Sensitivity analysis of k-fold cross validation in prediction error estimation. IEEE Trans Pattern Anal Mach Intell 32(3):569–575
28. Sanchez E, Yu W (2001) Differential neural network for robust nonlinear control identification, state estimation and trajectory tracking. Ed. Worls Scientific Publishing, Singapore
29. Shukla A, Macchiarulo L (Aug 2008) A fast and accurate FPGA based QRS detection system. In: Engineering in medicine and biology society, 2008. EMBS 2008. 30th annual international conference of the IEEE. IEEE, pp. 4828–4831
30. Taddei A, Distante G, Emdin M, Pisani P, Moody G, Zeelenberg C, Marchesi C (1992) The european st-t database: standard for evaluating systems for the analysis of st-t changes in ambulatory electrocardiography. Eur Heart J 13:1164–1172
31. Thomas M, KrDas M, Ari S (2015) Automatic ECG arrhythmia classification using dual tree complex wavelet based features. AEU-Int J Electron Commun 69(4):715–721
32. Zhang C, Tae-Wuk B (2012) VLSI friendly ECG QRS complex detector for body sensor networks. IEEE J Emerg Sel Top Circuits Syst 2(1):52–59