Received: October 8, 2020. Revised: November 10, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A QRS-Detection Algorithm for Real-Time Applications

Akram Jaddoa Khalaf[1]* **Samir Jasim Mohammed[1]**

[1]*Department of Electrical Engineering, Faculty of Engineering, University of Babylon, Iraq*
* Corresponding author's Email: eng.akram@uobabylon.edu.iq

Abstract: The QRS detection algorithm is substantial for healthcare monitoring and diagnostic applications. A low error detection without adding more computation is a big challenge for researchers. The proposed QRS detection algorithm is a simple, real-time, and high-performance hybrid technique based on decision tree and artificial neural networks (ANN). In this study, the five stages algorithm is designed, implemented, and evaluated for wearable healthcare applications. The first stage is filtering the original ECG signal to reduce the noise and baseline wandering. After that, a maximum or minimum moving-window for positive or negative peaks respectively is searching R-peaks for any expected value and finding the Q and S corresponding to this R-peak. Only these values from all ECG samples are passed to the next stage for feature extraction to reduce the algorithm computation. Stage four is excluded any unlikely points using the mean of the slope and level based on a simple decision tree. Finally, artificial neural networks are designed to classify the rest point for QRS detection using ANNs for each peak polarity to improve the network's performance by separating the data as a positive or negative peak. The algorithm is evaluated based on MATLAB using the MIT-BIH Arrhythmia Database, and the results show a low error rate detection of 0.25%, high sensitivity of 99.86%, and high predictivity of 99.89%. We develop a new approach for real-time QRS detection with low resources and high efficiency compared with other approaches.

Keywords: Biomedical, ECG, MIT-BIH, QRS Detection, Real-time, Signal processing.

1. Introduction

The electrocardiograph (ECG) is the process of reading the heart electrical signal over time. This signal shows the heart functionality for the cardiologist to diagnose cardiac diseases. Globally, the most common death is cardiovascular diseases [1]. So, this field developed significantly. For different biomedical applications like heart rate measuring, unnormal diagnosing, biometric identification, and movement recognition, the ECG is an efficient non-invasive tool [2]. The normal heart rhythm contains P, QRS complex, and T distinct waves [3-6]. In ECG signal processing, QRS-complex detection is essential for most applications. Heart rate can measure from the R-peaks time sequence after the detection of QRS waves. This analysis is not easy because the noise affected the ECG signal and physiological variability for patients; moreover, the similarity between some cases of T wave with QRS-complex characteristics [4, 7]. Muscular noise, electrode motion effects, power-line interference, and baseline wandering are the noise sources that affect the ECG signals.

The applications for real-time biomedical signal processing development are rapidly increasing because it's related to human life and the development of wearable devices. The Internet of things (IoT) and cloud play a significant to improve many applications; healthcare applications are at the top of these applications [8]. The QRS detection field seems to be saturated because a lot of algorithms are proposed. Despite that, the development of wearable devices and the healthcare system make a new challenge to increase the algorithm efficiency, performance, and portability [4, 5]. And to reduces the memory storage and transmit low rate date for wearable devices [7]. Many advanced techniques and tools are still not used for future improvement [9].

Different databases are widely used to evaluate the different application that proposed in many

DOI: 10.22266/ijies2021.0228.33

studies for ECG signals analysis [2]. The ECG databases are available now, such as MIT-BIH databases, QT database, and The American Heart Association database. The algorithm performance evaluation is done using a global data set like the MIT-BIH Arrhythmia Database [10], which contains several types of ECG signals with variable signal noise [7].

At the last time, many researchers have been developing various algorithms, methods, and techniques based on different concepts. Each approach depends on the application is designed, so some methods are used for real-time applications, and the others are used for offline applications. QRS detection methods are an essential field for wearable healthcare applications to reduce the data transmitting, speed processing time, and reduce the memory used. Linear filter, nonlinear processing, and decision are the three main stages used for most QRS detection algorithm [11]. These methods are based on filters design [4, 7, 12, 13], Short Time Fourier Transform [3], Hilbert Transform [14], Wavelet [15, 16], neural network (NN) [17, 18], energy detection [19, 20], entropy [18], Bayesian framework [8], and decision rules [21].

Pan and Tompkins [7] algorithm based on filtered the ECG signal to maximize the QRS frequency band and reduce the other frequencies. So, digital filters are designed for QRS-complex improvement. After that, it is preprocessing the filtered signal to extract slope information. Two thresholds are applied to detect the QRS and search back the missing QRS based on the RR interval accepted ranges. The error detection for this method is moderate. Two adaptive thresholds are applied to detect the QRS, searching back for missing QRS based on RR normal intervals and using a complex calculation for all signal samples, which are adding more computation. The following researchers improved this algorithm. Castells-Rufas and Carrabina [4] present a novel nonlinear filter to minimal computational reason. A moderate accuracy for this technique, less computation, and the same [7] computational detection model. Excluding the derivative filter and based on new decision techniques Yakut and Bolat are proposed a new method [21]. The first stage is contained two filters, square, and normalization. The second stage is the decision stage. A satisfying error detection rate, more computation because it is searching back the missing QRS based on normal durations, and using a complex calculation. A complex detection for positive and negative slopes using two stages as preprocessing and decision making presented by Adnane and others [12]. Squaring, normalizing, and differentiation are used for the first stage before the decision rules depending on RR durations. So, the complexity of the operation is increasing the execution time.

On the other hand, Zidelmal and others [15] are designed a discreet wavelet transform considering energy levels, and Sharma and others [16] presented a tunable-Q wavelet transform. These methods consist of a complex operation with moderate detection errors for offline applications. Zhang and others [13] are presented Kalman filter extract features for decrease computational cost, storage, fast response, and less accuracy with two adaptive threshold systems. Shaik and others [3] proposed an adaptive threshold technique based on Short Time Fourier Transform with moderate accuracy and initial training.

Moreover, Moraes and others [19], moderate detection sensitivity is evaluated using two channels ECG detection method, two detections in parallel cross detection as mainly, and energy detection as secondary. So, Chin and others [8] developed an efficient real-time and accurate QRS complex detector algorithm based on the Bayesian framework that contains a complex repeated multiply and accumulates calculations.

Finally, Xue and others [17] are developed an artificial neural network based on an adaptive whitening filter that updates the matched filter to improve the QRS detection. After squaring and moving average for all samples. Cai and Hu [18] Convolutional and Squeeze-and-Excitation networks (CNN), and hybrid Convolutional and Recurrent Neural Network (CRNN) based on deep learning methods are proposed. All ECG-samples are processing to these neural network methods and high neurons number that increasing the computation.

All these methods stages are processed all ECG samples without reducing them, so improving the performance will add more complexity. The samples squaring add more computation. So, they did not deal with QRS polarity, and that will reduce the detection accuracy. Because of the squaring process, they did not deal with QRS polarity.

Our algorithm developed a novel real-time hybrid approach based on a simple decision tree and artificial neural networks (ANN) classifier. The original ECG signal is preprocessing using a single moving average window to reduce the noise and artifacts. The ECG-signal is abbreviated by only some features extracted from the candidate's QRS-points after maximum or minimum searching window. So, the processing number for the next stages is reduced. Furthermore, the data are separated based on polarity to enhanced the detection performance with less computation.

These features are pass from a simple decision that removes the unlikely value to reduce the data before the ANN. The ANN classifies the separated features for positive peaks and negative peaks with two simple and high accurate separated networks.

In section 2, we describe the proposed algorithm and each stage in detail. Then, section 3 demonstrates the algorithm evaluation using the MIT-BIH-arrhythmia-database with performance results and discussion. Finally, in section 4, we summarize the conclusion.

2. Proposed algorithm

The algorithm based on five-stage is designed to improve the QRS- detection to be implemented for real-time wearable applications. ECG signal filtering using a simple moving average filter without any more filters to reduce the noise and baseline wander. The second stage searching window the R-peak, Q, and S using window for maximum and minimum values. These QRS points pass to stage three to extract features that are used for the next stages. Stage four is a simple decision using one of the features to pass any QRS near expected for the final decision stage based on a neural network. Stage five is the final stage that is using the overall features to inspect the QRS for detection purposes. The QRS detection algorithm block diagram is shown in Fig. 1. Each algorithm stage is described in the following subsection.

2.1 Pre-processing

The original ECG signal *s(t)* suffers baseline wander, power line interference, electrode, and muscle noise. A moving average with *N* window filters the signal to remove the noise and smooth the signal, as shown in Fig. 2. We used a moving average window with *N*=150 samples for ECG at a sampling rate of 360 samples per second. Eq. (1) shows the moving average filter for *N* samples.

$$m(t) = s(t) - \frac{1}{N}\sum_{i=1}^{N} s\left(t - \frac{N}{2} + i\right) \tag{1}$$

2.2 Peak search

The major point for any detection algorithm is finding R-peak. Searching the filtered signal *m(t)* for maximum or minimum value using *N*=90 single window for the signal rate 360 sample per second (depending on the maximum QRS width for the normal case is 150ms [7]) to find the positive-peak or negative-peak as present in Eq. (2). Where the absolute value of the max greater than the absolute of the min, the peak is positive and vice versa. These two types of peaks values will be as predicted R-peak. After that, search the Q and S points related to the predicted R-peak to get the predicted QRS (QRS^*) three points and eliminate the residual points (the signal samples that are not an R-peak or QS points). Therefore, the only signal points pass to the next stage are QRS^*.

For 360 samples, an average of 7 QRS^* can found that will reduce the original samples to three pairs for each QRS^* (7×6=42 points). Only 42 values are used after this stage instead of 360 samples to reduce computation. The rest of the samples is eliminated because it is far away to be QRS points. So, the computation for the next stages will be less than the convolutional methods. On the other hand, the convolutional methods process all ECG samples and add more complexity. Fig. 3 and Fig. 4 show the QRS-predicted points and the max or min values for the moving window.

$$r(n,t) = \begin{cases} +\,QRS^* & m(t) = max \;\&\; |max| > |min| \\ -\,QRS^* & m(t) = min \;\&\; |max| < |min| \\ \;\;- & otherwise \end{cases} \tag{2}$$

Where: $r(1-3,t)$: x − axis values
$r(4-6,t)$: y − axis values

This technique is used to speed the algorithm and eliminate the useless signal samples for the detection process. The *r(t)* contains x-values and y-values for the three QRS points for positive or negative waves to increase the accuracy using separate wave values for each QRS polarity. In contrast, the convolutional methods square the samples add more computation and decrease the performance.

2.3 Feature extraction

The features are extracted from the current QRS^* and some relations with the preceding QRS^* points. After finding all predicted points that could be actual or detected QRS, these points contain enough information that is used to detect the correct QRS points. The slope is the most feature used for detection in many algorithms. Moreover, the proposed algorithm extracts more features like the slope from QR, the slope from RS, and the level from Q to R and the level from R to S, and the time for QR, RS, and QS. These features will improve detection performance. In contrast, the existing methods based on threshold detection are used the slope feature only.

From the three QRS^* pair (x, y) points by using features selection methods and testing these features, fourteen features are extracted, as shown in Fig. 5 to

Received: October 8, 2020. Revised: November 10, 2020.

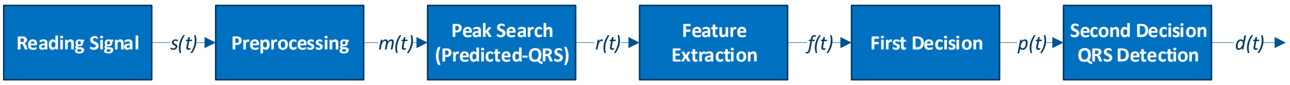

Figure. 1 The proposed QRS-detection algorithm block diagram

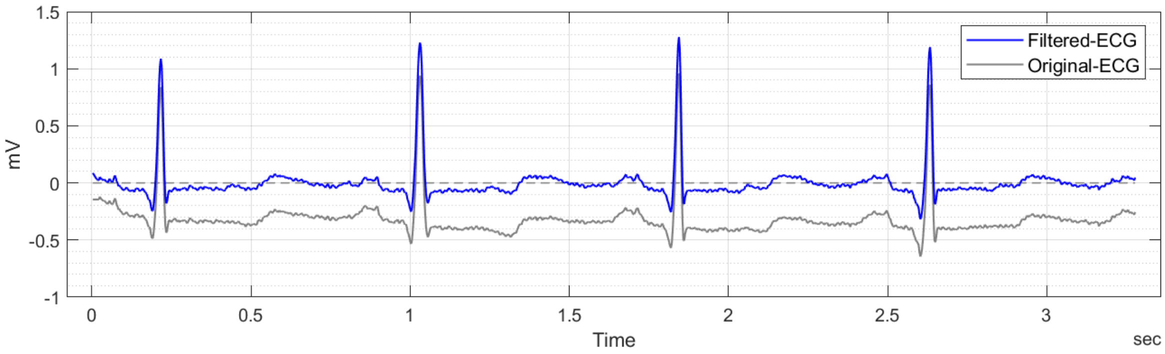

Figure. 2 The original and filtered signals

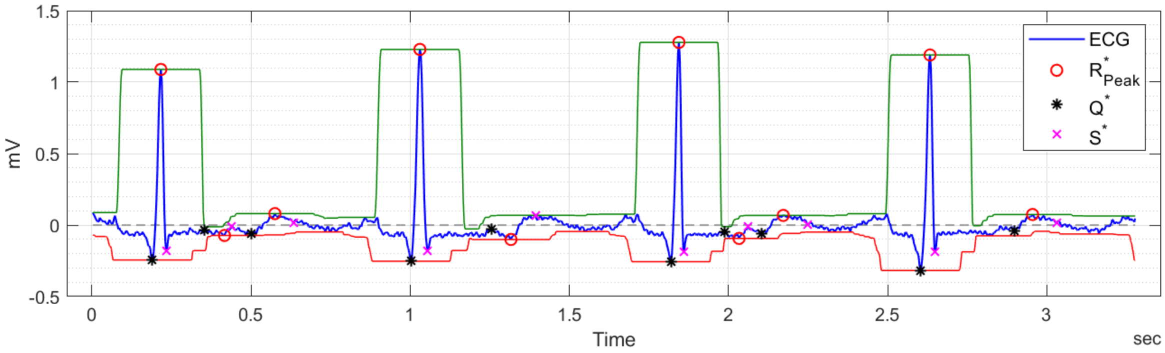

Figure. 3 Predicted QRS (QRS*) and the min-max window

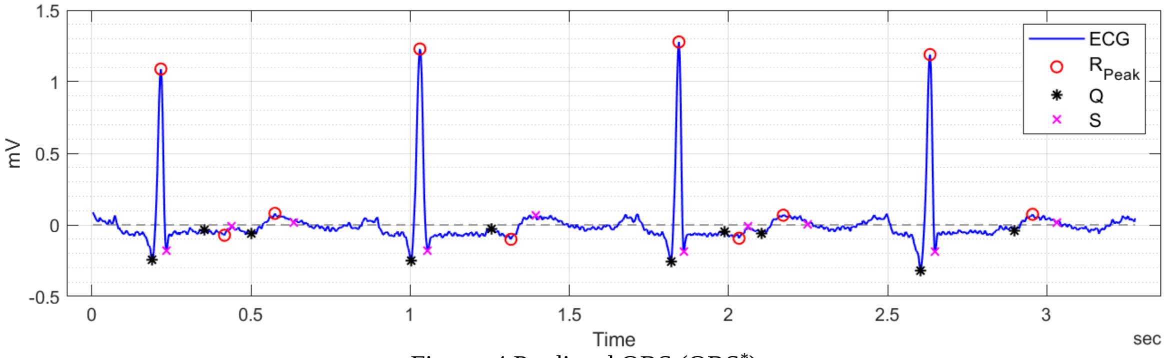

Figure. 4 Predicted QRS (QRS*)

improve the classification process's accuracy. These fourteen features started from *f* (1, *t*) to *f* (14, *t*), and its calculation equations are:

- (1-3) Q, R, and S levels (y-values for each of the three points)

$$f(1,t) = r(4,t) \quad (3)$$

$$f(2,t) = r(5,t) \quad (4)$$

$$f(3,t) = r(6,t) \quad (5)$$

- (4-5) *R-10_slope* and *R+10_slope* (the slope of fixed time from R point to t-10 and t+10)

$$f(4,t) = \frac{r(5,t) - r(5,t-10)}{20} \times fs \quad (6)$$

$$f(5,t) = \frac{r(5,t+10) - r(5,t)}{20} \times fs \quad (7)$$

- (6) *QRS_level* = *QR_level* + *RS_level* (y- value from R point to Q + from R point to S)

$$f(6,t) = r(5,t) - r(4,t) + r(5,t) - r(6,t) \quad (8)$$

- (7) *QRS_slope*= *QR_slope* +*RS_slope* (the slope value from Q point to R + the slope from S point to R)

$$f(7,t) = \left(\frac{r(5,t)-r(4,t)}{r(2,t)-r(1,t)} - \frac{r(6,t)-r(5,t)}{r(3,t)-r(2,t)}\right) \times fs \quad (9)$$

- (8-9) *QR_time* and *RS_time* (the x_values from Q point to R point and from R point to S point)

$$f(8,t) = r(2,t) - r(1,t) \quad (10)$$

$$f(9,t) = r(3,t) - r(2,t) \quad (11)$$

- (10) QRS_{dec} =*QRS_level* × *QRS_slope* (Eqs. (8) and (9) are used for first decision stage)

$$f(10,t) = f(6,t) \times f(7,t) \quad (12)$$

- (11) *QS_time* (the x_values from Q point to S point)

$$f(11,t) = f(8,t) + f(9,t) \quad (13)$$

- (12) m_QRS_{dec}: The mean for 15 values of the QRS_{dec} used as a threshold for the first decision stage start at five values for initializing algorithm.

$$f(12,t) = \frac{1}{N}\sum_{i=1}^{N=(5-15)} f(10,t-i+2) \quad (14)$$

- (13) The mean for three values of the QRS_{dec} (current, previews, and next values) that is used to eliminate noise detection.

$$f(13,t) = \frac{1}{3}\sum_{i=1}^{3} f(10,t-i+2) \quad (15)$$

- (14) The mean for three values of the *QRS_level* (current, previews, and next values) that is also used to eliminate noise detection.

$$f(14,t) = \frac{1}{3}\sum_{i=1}^{3} f(6,t-i+2) \quad (16)$$

The features are normalized using the following Eq. (17) because these features are calculated from a variable ECG signal. The amplitude of ECG signals changed depending on device connectivity, personal age, and some Heart diseases.

$$f(t) = \frac{2\times(f(t)-f_{min})}{f_{max}-f_{min}} - 1 \quad (17)$$

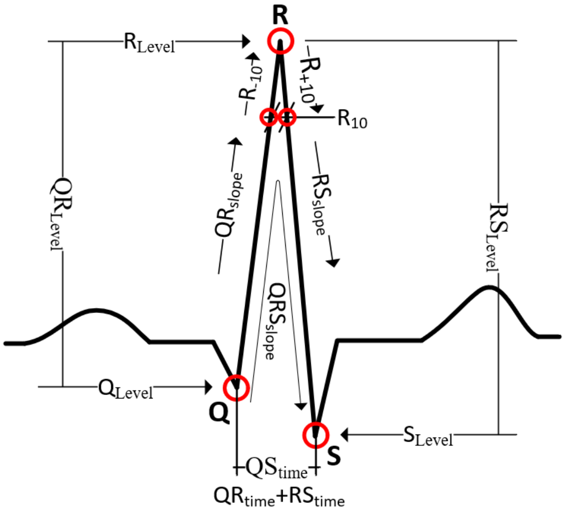

Figure. 5 The features of QRS*

2.4 First decision

The QRS level and slope are the best features that can be used to separate the QRS predicted or detected. So that each predicted (QRS^*) level and slope named (QRS_{dec}) that achieve the lowest mean of the QRS* level and slope named (m_QRS_{dec}) can be QRS and pass to the next stage. On the other hand, the QRS* level and slope (QRS_{dec}) that not achieving the condition are eliminated and excluded to be QRS. This stage is such a threshold in the other algorithm. Eq. (2) shows the first condition.

$$p(t) = \begin{Bmatrix} f(t) & QRS_{dec} > (0.2 \times m\_QRS_{dec}) \\ 0 & QRS_{dec} \le (0.2 \times m\_QRS_{dec}) \end{Bmatrix} \quad (18)$$

Where *p(t)*: the features that pass the first condition.

The simple decision tree using one feature (m_QRS_{dec}) to select the QRS^* that passes the lowest threshold condition and excludes the QRS^* that not passes the condition. The excluded QRS^* values are far away to be actual points. So, increasing the algorithm speed by reducing the number of values that are processing to the final stage using a simple decision stage.

2.5 Second decision (QRS detection)

A simple two-layer feed-forward artificial neural network (ANN) is designed with multi hard limiting hidden neurons and single linear output neurons. The network is trained with the backpropagation algorithm for 70% training data and 30% testing data. We design the ANN using network growing based on trial and error [22, 23], started at one hidden neuron, and, after that, adding more neurons to reach twice

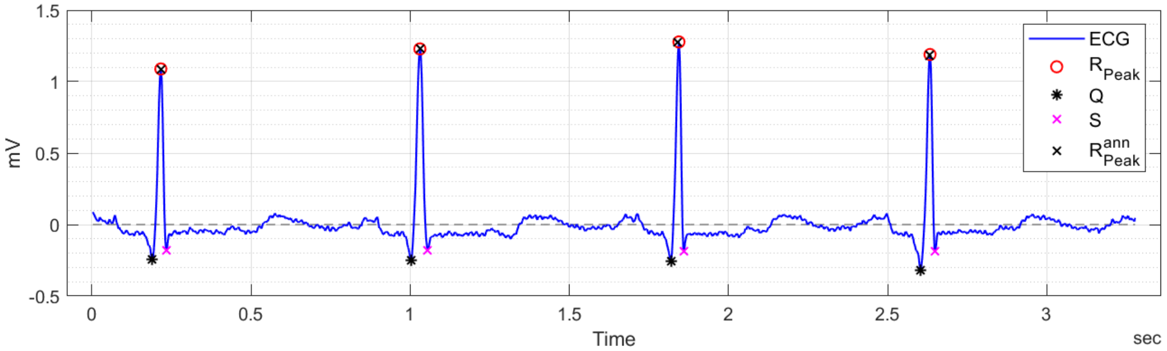

Figure. 6 The detected QRS and R-peak from the original annotation file

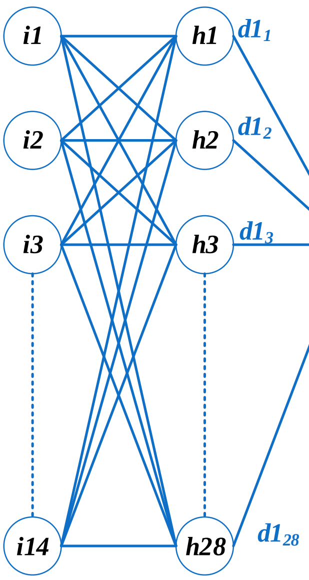

Figure. 7 The designed artificial neural network

the input number for the hidden layer to improve the classifier performance. Also, for two decision classifiers, a single output neuron is used. On the other hand, for each *QRS** positive and *QRS** negative, a particular full neural network is designed to improve the accuracy without adding more complexity. So, the algorithm computation will be simple than the other methods.

The classifier is designed using twenty-eight neurons in hidden layers and a single output neuron, as shown in Fig. 7. The features are feed to the first layer, which contains twenty-eight neurons fully connected and applies the Eqs. (19)-(22).

Where: *i*: 1, 2, ……14 no. of inputs (features)

j: 1, 2, ……28 no. of hidden neurons

$$net1_j = (b1_j + \sum_{i=1}^{14} w1_{i,j} \times f_i) \qquad (19)$$

$$d1_j(net1_j) = \left\{\begin{matrix} +1 & net1_j > 0 \\ -1 & net1_j \leq 0 \end{matrix}\right\} \qquad (20)$$

$$net2 = b2 + \sum_{j=1}^{28} w2_j \times d1_j \qquad (21)$$

$$d2(net2) = \left\{\begin{matrix} 1 & net2 > 0 \\ 0 & net2 \leq 0 \end{matrix}\right\} \qquad (22)$$

Eqs. (19) and (20) calculate the 28 hidden neurons output. Then Eq. (21) determines the net value for the output neuron. The output *d2(net2)* for Eq. (22) is QRS final detection; if the value is 1, the QRS detection is true and 0 for not a QRS detection.

From the preview's equations, the multiplication occurs in the ANN for low numbers of neurons (28 neurons only) Eqs. (19) and (21). Moreover, the number of values that processing is less than the original samples. Therefore, the proposed algorithm is low computation. The QRS detection and the original R-peak annotation file is shown in Fig. 6 as a final result for these stages. The overall algorithm flowchart and a sample ECG detection signal are described in Fig. 8.

3. Evaluation and implementation

The algorithm performance evaluated using MIT-BIH-arrhythmia-database [10] with variable ECG signals used for testing and evaluating because it contains a variable ECG signal and variable noise. Furthermore, it's widely used from many other approaches for comparison purposes. This database contains 48 records with 30 minutes, two channels, 360 samples per second, and 11-bit resolution. The database contains positive and negative QRS, high and low QRS, baseline wander, power line interference, muscle noise, regular and irregular heartbeats, and normal and unnormal wave. All mentioned previously makes this database is suitable for evaluation reasons. Therefore, we used this database by taking the first ECG channel from the database.

The proposed algorithm was implemented using the MATLAB program for software evaluation. The results considered the calculation of the True Positive (TP), False Negative (FN), and False Positive (FP) values that are the TP is the correct detection beats, FN is the fail detection beats, and FP is the wrong

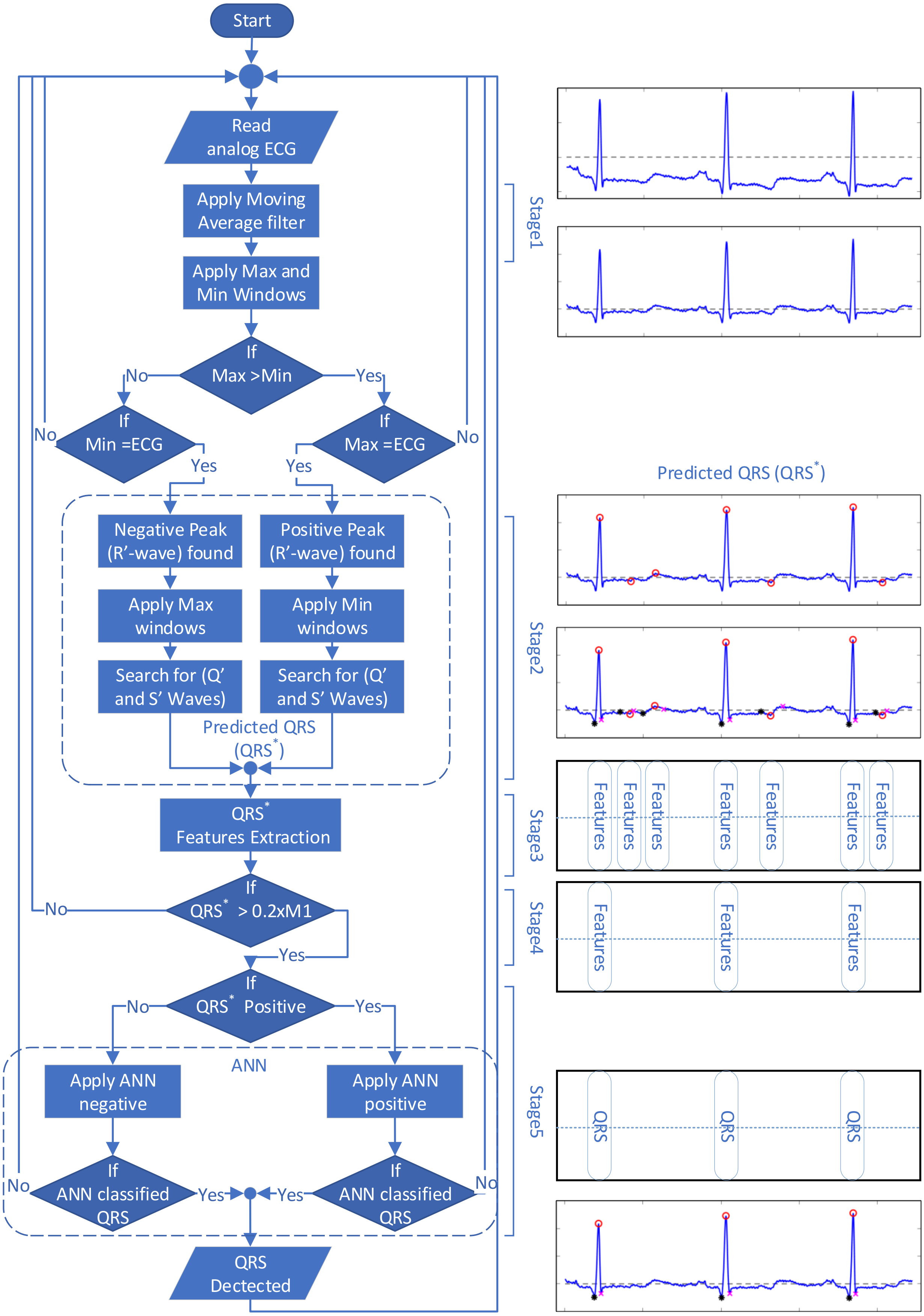

Figure. 8 The overall algorithm flowchart

Table 1. The database detection results

Records No.	TP (Beats)	FP (Beats)	FN (Beats)	FP+FN=FT (Beats)	S (%)	PP (%)	DER (%)
100	2273	0	0	0	100	100	0
101	1865	2	0	2	100	99.89	0.107
102	2187	0	0	0	100	100	0
103	2084	0	0	0	100	100	0
104	2229	1	0	1	100	99.96	0.045
105	2572	15	5	20	99.81	99.42	0.778
106	2027	0	0	0	100	100	0
107	2137	0	1	1	99.95	100	0.047
108	1763	2	5	7	99.72	99.89	0.397
109	2532	0	0	0	100	100	0
111	2124	0	1	1	99.95	100	0.047
112	2539	0	0	0	100	100	0
113	1795	0	0	0	100	100	0
114	1879	1	3	4	99.84	99.95	0.213
115	1953	0	0	0	100	100	0
116	2412	0	18	18	99.26	100	0.746
117	1535	0	0	0	100	100	0
118	2278	1	0	1	100	99.96	0.044
119	1987	0	0	0	100	100	0
121	1863	0	1	1	99.95	100	0.054
122	2476	0	0	0	100	100	0
123	1518	0	0	0	100	100	0
124	1619	1	0	1	100	99.94	0.062
200	2601	1	1	2	99.96	99.96	0.077
201	1963	0	8	8	99.59	100	0.408
202	2136	1	3	4	99.86	99.95	0.187
203	2980	25	23	48	99.23	99.17	1.611
205	2656	0	2	2	99.92	100	0.075
207	1860	30	11	41	99.41	98.41	2.204
208	2955	7	19	26	99.36	99.76	0.880
209	3005	2	0	2	100	99.93	0.067
210	2650	1	9	10	99.66	99.96	0.377
212	2748	1	0	1	100	99.96	0.036
213	3251	0	2	2	99.94	100	0.062
214	2262	2	3	5	99.87	99.91	0.221
215	3363	0	0	0	100	100	0
217	2208	1	2	3	99.91	99.95	0.136
219	2154	0	0	0	100	100	0
220	2048	0	0	0	100	100	0
221	2427	1	1	2	99.96	99.96	0.082
222	2483	1	2	3	99.92	99.96	0.121
223	2605	0	1	1	99.96	100	0.038
228	2053	12	6	18	99.71	99.42	0.877
230	2256	0	0	0	100	100	0
231	1571	0	0	0	100	100	0
232	1780	7	0	7	100	99.61	0.393
233	3079	0	0	0	100	100	0
234	2753	0	3	3	99.89	100	0.109
overall	109494	**115**	**130**	**245**	**99.881**	**99.895**	**0.224**

Table 2. Methods comparison for certain records

Rec. No.	*TP (B)*	Our Method			[3]			[4]			[7]			[13]			[15]			[21]		
		FP (B)	*FN (B)*	*FT (B)*	*FP (B)*	*FN (B)*	*FT (B)*	*FP (B)*	*FN (B)*	*FT (B)*	*FP (B)*	*FN (B)*	*FT (B)*	*FP (B)*	*FN (B)*	*FT (B)*	*FP (B)*	*FN (B)*	*FT (B)*	*FP (B)*	*FN (B)*	*FT (B)*
104	2229	1	0	**1**	12	12	**24**	6	6	**12**	1	0	**1**	113	2	**115**	12	18	**30**	9	4	**13**
105	2572	15	5	**20**	35	12	**47**	40	34	**74**	67	22	**89**	45	5	**50**	15	44	**59**	12	16	**28**
108	1763	2	5	**7**	21	8	**29**	34	52	**86**	199	22	**221**	79	115	**194**	25	35	**60**	34	15	**49**
114	1879	1	3	**4**	42	71	**113**	5	14	**19**	3	17	**20**	3	2	**5**	1	4	**5**	3	5	**8**
116	2412	0	18	**18**	2	23	**25**	1	30	**3**	3	22	**25**	3	22	**25**	4	15	**19**	11	8	**19**
200	2601	1	1	**2**	6	3	**9**	12	12	**24**	6	3	**9**	44	2	**46**	11	15	**26**	2	4	**6**
203	2980	25	23	**48**	17	74	**91**	19	182	**201**	53	30	**83**	80	27	**107**	25	36	**61**	22	25	**47**
208	2955	7	19	**26**	1	66	**67**	4	26	**30**	4	14	**18**	4	21	**25**	5	25	**30**	9	14	**23**
222	2483	1	2	**3**	5	7	**12**	0	1	**1**	101	81	**182**	4	9	**13**	2	2	**4**	3	1	**4**
228	2053	12	6	**18**	19	3	**22**	7	12	**19**	25	5	**30**	184	4	**188**	27	57	**84**	30	15	**45**

detection beats.

The following Eqs. (23)-(25) are calculated for entirely database records only excepted the ventricular flutter beats in record 207 that are not included to evaluate the algorithm performance and compare the results with the other methods. Table 1 demonstrates the proposed detection algorithm results for the MIT-BIH database.

$Sensitivity\ (S)$:

$$(S)\% = \frac{TP}{TP+FN} \times 100 \tag{23}$$

$Positive\_Predicitvity\ (PP)$:

$$(PP)\% = \frac{TP}{TP+FP} \times 100 \tag{24}$$

$Detection\_Error\_Rate\ (DER)$:

$$(DER)\% = \frac{TP}{TP+FN} \times 100 \tag{25}$$

The results show a promising QRS detection with high overall sensitivity and predictivity 99.86% and 99.89%, respectively, and with a low detection error rate of 0.224%. Moreover, it achieved a 99.75% average detection accuracy. A MATLAB 2020a 64-bit in 2.6 GHz Intel Ci7 computer is used for performance evaluation. The average computation time for the database of 30-minutes record is (1.196sec), which is a low computation algorithm [13, 20-21].

The database has difficult records for existing QRS-detection methods because it is suffering from different types of noise, artifacts, irregular beats, and unnormal waves. The most detection errors occur from these records. Table 2 demonstrates the proposed algorithm, and some other algorithms are compared to the detection results errors (FP, FN) for specific records.

Table 3. Comparison of methods performance

Ref. No.	S (%)	PP (%)	DER (%)
Our	**99.881**	**99.895**	**0.224**
[16]	99.833	99.881	0.287
[8]	99.858	99.850	0.293
[21]	99.832	99.834	0.334
[15]	99.642	99.824	0.535
[12]	99.769	99.642	0.590
[7]	99.762	99.565	0.675
[4]	99.442	99.679	0.883
[3]	99.559	99.516	0.929
[13]	99.240	99.383	1.387

These results are based on the technique of searching all potential QRS points (QRS*) and separate these points depending on the QRS polarity. The algorithm improves the performance for negative QRS as in records (108,114) comparing with other methods. Detect the true QRS using two simple decisions based on the selected features. In addition to the slope, the level and the remaining features play a significant role in reducing error detection. Some of these features show in Eqs. (15) and (16) are reduced FN for high-frequency and noisy records (105, 200, 203, 222). The premature ventricular contraction, these features are detected with low errors like records (116, 200, 203, 208, 228).

In this paper, we present a hybrid technique base on decision tree and artificial neural networks for QRS detection. It consists of a particular five stages compared to the included three stages of the other methods. The most stages of other methods are included because it consists of more than one operation in each stage. On the other hand, our algorithm stage consists of one operation. There are several distinct features for the proposed algorithm:

1. The algorithm is based on searching all possible QRS points (QRS^*) using the max or min moving window and remove the other point to reduce the computation. So, the ECG samples are reduced by the QRS^* points only to increase the speed.
2. The QRS^*s are positive and negative depending on the peak polarity to improve the accuracy. Furthermore, there are no squaring samples to reduce the multiplication.
3. The features extracted from QRS^* are not depending on slope only. Moreover, depending on the level between these points and the time to improve the detection.
4. Two simple ANN's are used for QRS^* positive and QRS^* negative with high classification accuracy because it depends on the polarity.
5. The low computation is from less data, no squaring all samples, and no back search for missing peaks base on RR interval calculation that is used for the conventional methods.

Finally, the overall evaluation performance comparison of the proposed algorithm with some other algorithm using the same MIT-BIH Arrhythmia Database are summarized in Table 3. These results describe, the performance of our algorithm is achieved better than the other methods.

4. Conclusion

We have proposed a novel QRS detection algorithm using a hybrid technique base on a simple decision tree and artificial neural network. It searches all predicted QRS values and select the actual values using two decision stages without any of the following costly computational processes: back search for missing values, adaptive decision, RR interval range checking, starting learning, complex filter, squaring all data, and complex calculation for all signal samples. So, the features are extracted with simple calculation from some signal points (predicted-QRS) and eliminating the most samples that not near QRS to simplify the processing. Moreover, the first decision is to use the level values with the slope to improve accuracy. The ANN data are separating based on the peak polarity. Therefore, each ANN can classify the data more accurately and less complicated to achieve better performance.

The average speed for each 30-minutes record of the proposed algorithm is 1.196sec. Also, the computation of the proposed algorithm is low. The high-performance is achieved based on a low detection error rate of 0.224% compared with the existing methods. Furthermore, it has a high QRS detection sensitivity (99.86%), high predictivity (99.89%), and average detection accuracy of 99.75%. Therefore, the algorithm has a promising result compared to the proposed methods. It can be implemented for a wearable application that requires a real-time and high accuracy algorithm.

As future work, the algorithm will be implemented using a suitable programing language and hardware based on IoT prototype wearable device with relevant features.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, Akram Jaddoa Khalaf and Samir Jasim Mohammed; methodology, Akram Jaddoa Khalaf and Samir Jasim Mohammed; software, Akram Jaddoa Khalaf; validation, Akram Jaddoa Khalaf, and Samir Jasim Mohammed; formal analysis, Akram Jaddoa Khalaf; investigation, Akram Jaddoa Khalaf; resources, Akram Jaddoa Khalaf and Samir Jasim Mohammed; data curation, Akram Jaddoa Khalaf; writing—original draft preparation, Akram Jaddoa Khalaf; writing—review and editing, Akram Jaddoa Khalaf and Samir Jasim Mohammed; visualization, Akram Jaddoa Khalaf; supervision, Samir Jasim Mohammed; project administration, Akram Jaddoa Khalaf and Samir Jasim Mohammed; funding acquisition, Akram Jaddoa Khalaf .

References

[1] S. Mendis, P. Puska, and B. Norrving, *Global Atlas on Cardiovascular Disease Prevention and Control*, Geneva: World Health Organization in Collaboration with The World Heart Federation and The World Stroke Organization, 2011.

[2] S. Berkaya, A. Uysal, E. Gunal, S. Ergin, S. Gunal, and M. B. Gulmezoglu, "A survey on ECG analysis", *Biomedical Signal Processing and Control*, Vol. 43, pp. 216-235, 2018.

[3] B. Shaik, G. Naganjaneyulu, T. Chandrasheker, and A. Narasimhadhan, "A Method for QRS

Delineation Based on STFT Using Adaptive Threshold", *Procedia Computer Science*, Vol. 54, pp. 646-653, 2015.

[4] D. Castells-Rufas and J. Carrabina, "Simple real-time QRS detector with the MaMeMi filter", *Biomedical Signal Processing and Control*, Vol. 21, pp. 137-145, 2015.

[5] M. Elgendi, M. Jonkman, and F. DeBoer, "Frequency Bands Effects on QRS Detection", *BIOSIGNALS*, Vol. 2003, pp. 428-431, 2010.

[6] M. Gusev and E. Domazet, "Optimal DSP bandpass filtering for QRS detection", In: *41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, pp. 303-308, 2018.

[7] J. Pan and W. Tompkins, "A Real-Time QRS Detection Algorithm", *IEEE Transactions on Biomedical Engineering*, Vol. BME-32, No. 3, pp. 230-236, 1985.

[8] W. Chin, C. Chang, C. Tseng, Y. Huang, and T. Jiang, "Bayesian Real-Time QRS Complex Detector for Healthcare System", *IEEE Internet of Things Journal*, Vol. 6, No. 3, pp. 5540-5549, 2019.

[9] B. Kohler, C. Hennig, and R. Orglmeister, "The principles of software QRS detection", *IEEE Engineering in Medicine and Biology Magazine*, Vol. 21, No. 1, pp. 42-57, 2002.

[10] G. Moody and R. Mark, "The impact of the MIT-BIH Arrhythmia Database", *IEEE Engineering in Medicine and Biology Magazine*, Vol. 20, No. 3, pp. 45-50, 2001.

[11] O. Pahlm and L. Sörnmo, "Software QRS detection in ambulatory monitoring - a review", *Medical & Biological Engineering & Computing*, Vol. 22, No. 4, pp. 289-297, 1984.

[12] M. Adnane, Z. Jiang, and S. Choi, "Development of QRS detection algorithm designed for wearable cardiorespiratory system", *Computer Methods and Programs in Biomedicine*, Vol. 93, No. 1, pp. 20-31, 2009.

[13] Z. Zhang, Q. Yu, Q. Zhang, N. Ning, and J. Li, "A Kalman filtering based adaptive threshold algorithm for QRS complex detection", *Biomedical Signal Processing and Control*, Vol. 58, p. 101827, 2020.

[14] D. Benitez, P. Gaydecki, A. Zaidi, and A. Fitzpatrick, "A new QRS detection algorithm based on the Hilbert transform", *Computers in Cardiology (Cat. 00CH37163)*, Vol. 27, pp. 379-382, 2000.

[15] Z. Zidelmal, A. Amirou, M. Adnane, and A. Belouchrani, "QRS detection based on wavelet coefficients", *Computer Methods and Programs in Biomedicine*, Vol. 107, No. 3, pp. 490-496, 2012.

[16] A. Sharma, S. Patidar, A. Upadhyay, and U. Acharya, "Accurate tunable-Q wavelet transform based method for QRS complex detection", *Computers & Electrical Engineering*, Vol. 75, pp. 101-111, 2019.

[17] Q. Xue, Y. Hu, and W. Tompkins, "Neural-network-based adaptive matched filtering for QRS detection", *IEEE Transactions on Biomedical Engineering*, Vol. 39, No. 4, pp. 317-329, 1992.

[18] W. Cai and D. Hu, "QRS Complex Detection Using Novel Deep Learning Neural Networks", *IEEE Access*, Vol. 8, pp. 97082-97089, 2020.

[19] J. Moraes, M. Freitas, F. Vilani, and E. Costa, "A QRS complex detection algorithm using electrocardiogram leads", In: *Computers in Cardiology*, Memphis, TN, USA, pp. 205-208, 2002.

[20] M. Manikandan and K. Soman, "A novel method for detecting R-peaks in electrocardiogram (ECG) signal", *Biomedical Signal Processing and Control*, Vol. 7, No. 2, pp. 118-128, 2012.

[21] Ö. Yakut and E. Bolat, "An improved QRS complex detection method having low computational load", *Biomedical Signal Processing and Control*, Vol. 42, pp. 230-241, 2018.

[22] K. Sheela and S. Deepa, "Review on Methods to Fix Number of Hidden Neurons in Neural Networks", *Mathematical Problems in Engineering*, Vol. 2013, pp. 1-11, 2013.

[23] S. Xu and L. Chen, "A novel approach for determining the optimal number of hidden layer neurons for FNN's and its application in data mining", In: *Proc. of the 5th International Conf. on Information Technology and Applications (ICITA '08)*, pp. 683-686, 2008.

Received: October 8, 2020. Revised: November 10, 2020.

Appendix A

Table 4. The notations list

Symbol	Description
$+QRS^*$	Positive predicted QRS
$-QRS^*$	Negative predicted QRS
B	Beats
$b1$	Hidden layers biases
$b2$	Output bias
$d1(net1_j)$	Hidden output for the j'th neuron
$d2(net2)$	The output for output neuron
f_{max}	The maximum value for each feature
f_{min}	The minimum value for each feature
fs	Sampling frequency
h_j	Hidden layer j'th neuron
m_QRS_{dec}	The mean for 15 values of the QRS_{dec}
N	Number of samples per window
$net1_j$	Hidden nets for the j'th neuron
$net2$	Net for output neuron
$o1$	Output neuron
QR_level	The y- value from R point to Q point
QR_slope	The slope from Q point to R
QR_time	The x_values from Q point to R point
QRS^*	Predicted QRS
QRS_level	QR_level + RS_level
QRS_slope	QR_slope + RS_slope
QRS_{dec}	*QRS_level* × *QRS_slope*
QS_time	The x_values from Q point to S point
R+10_slope	The slope from peak R(t) to R(t+10)
R-10_slope	The slope from peak R(t) to R(t-10)
RS_level	The y- value from R point to S point
RS_slope	The slope from R point to S point
RS_time	The x_values from R point to S point
t	Time
$w1$	Inputs weights
$w2$	Hidden layer weight