QRS detection in single-lead, telehealth electrocardiogram signals: benchmarking open-source algorithms

Florian Kristof[a], Maximilian Kapsecker[a,b], Leon Nissen[b], James Brimicombe[c], Martin R. Cowie[d], Zixuan Ding[c], Andrew Dymond[c], Stephan M. Jonas[b], Hannah Clair Lindén[e], Gregory Y. H. Lip[f,g], Kate Williams[c], Jonathan Mant[c], Peter H. Charlton[c,*] and on behalf of the SAFER Investigators

[a]*TUM School of Computation, Information, and Technology, Technical University of Munich, Garching bei München, 85748, Germany*
[b]*Institute for Digital Medicine, University Hospital Bonn, Bonn, 53127, Germany*
[c]*Department of Public Health and Primary Care, University of Cambridge, Cambridge, CB1 8RN, United Kingdom*
[d]*School of Cardiovascular Medicine & Sciences, Faculty of Lifesciences & Medicine, King's College London, London, United Kingdom*
[e]*Zenicor Medical Systems AB, Stockholm, 113 59, Sweden*
[f]*Liverpool Centre for Cardiovascular Science at University of Liverpool, Liverpool John Moores University and Liverpool Heart & Chest Hospital, Liverpool, United Kingdom*
[g]*Danish Center for Health Services Research, Department of Clinical Medicine, Aalborg University, Aalborg, Denmark*

ARTICLE INFO

Keywords:
signal processing
atrial fibrillation
beat detection
heartbeat

ABSTRACT

Background and Objectives: A key step in electrocardiogram (ECG) analysis is the detection of QRS complexes, particularly for arrhythmia detection. Telehealth ECGs present a new challenge for automated analysis as they are noisier than traditional clinical ECGs. The aim of this study was to identify the best-performing open-source QRS detector for use with telehealth ECGs.

Methods: The performance of 18 open-source QRS detectors was assessed on six datasets. These included four datasets of ECGs collected under supervision, and two datasets of telehealth ECGs collected without clinical supervision. The telehealth ECGs, consisting of single-lead ECGs recorded between the hands, included a novel dataset of 479 ECGs collected in the SAFER study of screening for atrial fibrillation (AF). Performance was assessed against manual annotations.

Results: A total of 12 QRS detectors performed well on ECGs collected under clinical supervision (F_1 score $\geq$ 0.96). However, fewer performed well on telehealth ECGs: five performed well on the TELE ECG Database; six performed well on high-quality SAFER data; and performance was poorer on low-quality SAFER data (three QRS detectors achieved F_1 of 0.78-0.84). The presence of AF had little impact on performance.

Conclusions: The Neurokit and University of New South Wales QRS detectors performed best in this study. These performed sufficiently well on high-quality telehealth ECGs, but not on low-quality ECGs. This demonstrates the need to handle low-quality ECGs appropriately to ensure only ECGs which can be accurately analysed are used for clinical decision making.

1. Introduction

The electrocardiogram (ECG) is one of the most widely used physiological measurement techniques, providing detailed information on heart function. Traditionally ECG measurements have been confined to clinical settings. However, recently it has become possible to measure the ECG in telehealth settings using handheld devices or smartwatches (1; 2). This presents the opportunity to conduct health assessment beyond the clinical setting, with potential applications including remote health monitoring, personalized diagnosis, rehabilitation, and screening for atrial fibrillation (AF). Indeed, the recent COVID-19 pandemic has acted as a strong catalyst for innovation in this area (3). However, the increasing use of wearable and telehealth technologies also presents new challenges.

A key challenge is that telehealth ECGs can be of lower quality than those collected in clinical settings, and so ECG analysis algorithms must be able to handle increased noise levels. Telehealth ECGs can be of lower quality for several reasons (4): the ECG is often measured further away from the heart (such as at the hands rather than the chest); devices typically use dry electrodes rather than the more conductive adhesive electrodes; and there is less quality control since measurements are taken by a non-expert user without clinical supervision. Therefore, there is a need to understand how well ECG analysis algorithms perform in the telehealth setting.

QRS detection is a fundamental task in ECG analysis. QRS complexes indicate ventricular depolarisation, *i.e.* the electrical impulse which causes the the heart to pump blood into the circulation. QRS detection is widely used for heart rate and rhythm monitoring, and heart rate variability analysis. Furthermore, QRS detection is frequently the first step towards extraction of more detailed ECG features such as QT intervals and P-waves. A range of QRS detection algorithms have been proposed (5; 6), most of which were developed using ECGs collected in clinical settings ((4) being a notable

*Corresponding author
pc657@medsch1.cam.ac.uk (P.H. Charlton)
ORCID(s): 0000-0002-3907-0749 (M. Kapsecker); 0009-0009-3884-0850 (L. Nissen); 0000-0001-7457-2552 (M.R. Cowie); 0009-0002-2337-3398 (Z. Ding); 0000-0002-3687-6165 (S.M. Jonas); 0000-0002-7566-1626 (G.Y.H. Lip); 0000-0002-9531-0268 (J. Mant); 0000-0003-3836-8655 (P.H. Charlton)

exception). Therefore, there is a need to assess their performance with telehealth ECGs. QRS detectors should firstly be accurate, correctly identifying QRS complexes. They should ideally remain accurate in the presence of pathologies such as AF (which results in an irregular heart rhythm), and in the presence of noise. In addition, QRS detection algorithms should also be stable with low execution times to ensure they are suitable for rapid and long-term analyses.

Previous studies have compared the performance of QRS detection algorithms across databases recorded in different settings. Liu *et al.* assessed ten QRS detectors across five datasets including one telehealth dataset (5). The algorithms, chosen for their computational efficiency, achieved F_1 scores of >99% on high-quality signals, ≤ 80% for low-quality signals, and ≥ 94% during pacing and in the presence of arrhythmias. The study concluded that an optimized knowledge-based algorithm (7) performed best. Llamedo and Martinez assessed six QRS detectors on 12 databases covering five categories: normal sinus rhythm, arrhythmia, ST and T morphology changes, stress, and long-term monitoring (6). The study concluded that the *gqrs* algorithm performed best. Research in (8) assessed 12 QRS detectors across five publicly available datasets. The study concluded that the neurokit (*nk*) algorithm performed best when considering both accuracy and execution time. Previous work in this area addresses known algorithms and benchmark ECG databases, but there is a lack of knowledge about the latest algorithms and their application to new telehealth databases, especially their performance on self-recorded ECGs.

The aim of this study was to identify the best-performing open-source QRS detector for use with telehealth ECGs. The performance of 18 algorithms was assessed on multiple datasets including a novel dataset collected using handheld devices during screening for AF. Performance was assessed primarily in terms of the accuracy of QRS detection (quantified using the F_1 score), and also in terms of the execution time and error rate of algorithms. The findings address the gap in knowledge about how well QRS detection algorithms perform in telehealth and settings. They are particularly relevant given the rapid introduction of single-lead ECG technology in consumer devices such as smartwatches, and clinical devices such as handheld ECG recorders.

2. Methods

2.1. QRS detection algorithms

The 18 QRS detectors assessed in this study are summarised in Table 1 (with source links provided in Appendix A.1). The QRS detectors were identified through a search for open-source algorithms. The majority of algorithms were found in either in the 'NeuroKit' (8) or 'ecgdetector' (9) Python packages. Some algorithms were available in both packages with slightly different implementations, in which case the faster implementation was used. Python implementations were used where available to provide a fair comparison of algorithm execution times. In four cases Python implementations were not available and Matlab implementations were used instead (*jqrs*, *rdeco*, *rpeak*, and *unsw*). Six additional algorithms were identified but not used in this study due to one of the following reasons: (i) no Python or Matlab implementation was available; (ii) the available implementation only accepted particular sampling frequencies; (iii) the available implementation predominantly led to errors; or (iv) the execution time was substantially longer than that of other algorithms. Further details are provided in Appendix A.2.

2.2. Datasets

The performance of QRS detectors was assessed using six datasets, including datasets collected in inpatient, outpatient, and home settings. The datasets are summarised in Table 2 and described in the following paragraphs. Full source links for the datasets are provided in Appendix A.3.

2.2.1. MIT-BIH Normal Sinus Rhythm Database (SIN)

The MIT-BIH Normal Sinus Rhythm Database (SIN) contains 18 24-hour ECG recordings from patients referred to the Arrhythmia Laboratory at Boston's Beth Israel Hospital, who were found not to have significant arrhythmias (16). The first two hours of each recording were used in this study. The subjects consisted of 13 women and 5 men, aged 20 to 50. Each recordings contains two ECG channels of unknown leads, the first of which was used in this analysis.

2.2.2. MIT-BIH Arrhythmia Database (ARR)

The MIT-BIH Arrhythmia Database (ARR) contains 48 30-minute ECG recordings from 47 patients referred to the same Arrhythmia Laboratory (29; 16). This dataset consists of 23 recordings which were selected at random from a larger dataset and a further 25 recordings which were manually selected to include examples of significant but uncommon arrhythmias. The subjects included 22 women and 25 men aged 23 to 89. The first ECG channel in each recording was analysed, which was the modified limb lead II in most cases.

2.2.3. PhysioNet/Computing in Cardiology Challenge 2014 Datasets (HIGH and LOW)

The PhysioNet/Computing in Cardiology Challenge 2014 datasets consist of 10-minute ECG recordings from patients and healthy volunteers (30; 16). The two publicly available datasets were used in this study: (i) the Training Set (HIGH), which contains 100 recordings which are generally of high quality; and (ii) the Augmented Training Set (LOW), which contains 100 recordings that are generally of low quality. Each record in these datasets contains a single ECG lead. The LOW dataset contains the following leads: lead II (78 records); lead III (5), lead AVF (3), lead AVL (1), and no lead label (13). No lead labels are provided in the HIGH dataset.

2.2.4. TELE ECG Database (TELE)

The TELE ECG Database contains 250 30-second lead-I ECG recordings from home-dwelling patients suffering from chronic obstructive pulmonary disease and/or congestive heart failure (32; 4). Recordings were acquired without

Table 1
QRS detection algorithms

Abbreviation	Description	Reference(s)
christ	**christov:** Detect QRS complexes as points exceeding an adaptive threshold consisting of the sum of: (i) steep-slope threshold (a linear reduction from 200ms to 1200ms 200ms after a QRS); (ii) adaptive integrating threshold (increases in the presence of electromyogram noise); and (iii) adaptive beat expectation threshold (a linear reduction between 2/3 and 1 mean RR-interval after a QRS).	(10)
engz	**engzee, sqrs:** Detection of QRS complexes as points exceeding a threshold based on a filtered signal: (i) removal of baseline wandering; (ii) application of a threshold to detect R-peaks; (iii) application of a refractory period to prevent multiple detection of a single R-peak.	(11; 12)
fnvg	**FastNVG**: A natural visibility graph (NVG) based R-peak detector: (i) representing the ECG signal as a graph using NVG; (ii) calculation of a node metric in the graph domain for weighting the signal to emphasize R-peak positions (iii) applying thresholding as with pan-tomp.	(13; 14)
fwhvg	**FastWHVG:** An R-peak detector based on the horizontal visibility graph (HVG): The method is similar to the fnvg algorithm, but is computationally more efficient since the HVG is a subset of the NVG.	(13; 14)
gamb	**gamboa:** Detection using amplitude histogram and critical points: (i) signal normalization using the amplitude histogram; (ii) detection of critical points in the first derivative exceeding a threshold; (iii) elimination of false beats through constraints on detected ECG signal beats; (iv) Computation of the mean ECG wave to obtain QPRS features.	(15)
gqrs	**gqrs:** The ECG beat detection algorithm initiates with: (i) employing a trapezoid low-pass filter to the signal, followed by a QRS matched filter convolution. (ii) The parameters of recent intervals and peak thresholds are adjusted without recording QRS locations. (iii) Sample detection occurs, identifying larger samples and peaks that surpass the QRS threshold, thus marking them as QRS complexes. If no peak is detected, the system lowers the peak detection threshold. (iv) Primary and secondary peak identification differentiates between peak types based on neighborhood size and relevance to a previous primary peak or associated T-wave.	(16)
hamilt	**hamilton, eplimited:** Detect QRS complexes using filtering, differentiation, rectification, and a moving window method: (i) applying low-pass and high-pass filtering to the signal; (ii) calculating the signal's derivative; (iii) rectifying the signal and utilizing a moving window of 80 ms; (iv) detecting QRS complexes following a predefined rule set.	(17; 18)
jqrs	**jqrs:** QRS detection enhanced by sliding window and custom filter: (i) window-based peak energy detector; (ii) band-pass filter with QRS matched filter (Mexican hat); (iii) reject detections based on heuristic during flat lines; (iv) search-back procedure for suspected missed beats.	(19; 20; 21)
kali	**Kalidas and Tamil, Stationary Wavelet Transform (swt):** Peak detection using Stationary Wavelet Transform: (i) resample signal to 80 Hz for real-time processing; (ii) compute 2-level SWT using 'db3' wavelet; (iii) square and MWA to enhance QRS peaks; (iv) threshold-based peak detection; (v) detect missed beats based on RR intervals; (vi) determine actual R-peak location within 0.10 seconds.	(22; 8)
mart	**martinez, wavedet, Continuous Wavelet Transorm (CWT):** The algorithm functions by executing a continuous wavelet transformation of the ECG signal across five distinct scales: (i) Each scale calculates a standard deviation, epsilon, from the transformed signal and peaks exceeding this epsilon are identified. (ii) The algorithm then filters these identified peaks across each scale, keeping only those closely associated with preceding scale peaks. (iii) It locates R-peaks by pinpointing zero-crossings in scale one within a specified range.	(23)
nab	**nabian:** Usage of sliding window with adaptive thresholds and domain knowledge to detect PQRST points: (i) Filter using Elliptic, Gaussian, or Butterworth (default: Elliptic); (ii) Detect potential R-peaks using global maxima in sliding window; (iii) Eliminate R-peaks below amplitude threshold; (iv) Find missing R-peaks using R-R interval; (v) Detect PQST points using predefined R-based locations.	(24)
nk	**neurokit:** Usage of signal smoothing and gradients to detect QRS complexes: (i) Computing the gradient and average gradient threshold of the highpass-filtered raw ECG signal. (ii) Identifying the start and end of QRS complexes by comparing the signal's smoothed gradient with the gradient threshold. (iii) Ignoring QRS complexes that are too short by setting a minimum length. (iv) Identifying R-peaks within each QRS. (v) Ensuring peaks identified are not too close together by enforcing a minimum delay.	
pan-tomp	**pan tompkins:** Filtering of the signal to segment the QRS complex: (i) low-pass and high-pass filtering; (ii) derivative of the signal; (iii) squaring of the derivative to amplify the QRS complex; (iv) adaptive thresholding with a refractory period using a moving window approach.	(25)
rdeco	**r-deco:** QRS detection using an envelope-based method: (i) using the difference between the lower and upper envelopes to flatten the signal; (ii) limiting the search range by considering segments whose value is higher than the 80 ms later value and whose upward slope lasts longer than 80 ms; (iii) selecting the segments of maximal value; (iii) defining the R peaks using the Pan-Tompkin adaptive thresholding method; (iV) eliminating false detections by performing a 50 ms backward search for each peak.	(26)
rpeak	**rpeakdetect:** Periodic adjustment of thresholds and parameters to detect QRS complexes using sensitivity-appropriate filtering: (i) cascaded low-pass and high-pass filtering to reduce signal noise; (ii) approximation of a derivative and application of an amplitude squaring operation; (iii) use of a moving window integrator with adaptive thresholds to segment the locations of QRS complexes.	(25; 18)
two-avg	**two average, elgendi:** Application of statistical thresholds and moving averages to generate blocks of interest: (i) bandpass filtering; (ii) integration of moving averages to generate blocks of interest; (iii) rejection of blocks smaller than the QRS complex length of the healthy adult and detection of the R peak as the maximum value of the remaining blocks.	(27)
unsw	**unsw:** Application of an adaptive threshold to a feature enriched signal: (i) filtering the signal with detrending, median filtering, and bandpass filters; (ii) calculating the QRS feature using the differentiated and filtered signal; (iii) smoothing the QRS features signal frequencies using the fundamental frequency as the lower bound; (iv) Adaptive threshold calculation using windows of different lengths on the filtered QRS feature signal; (v) Identification of possible QRS complexes using a peak-through detector on the filtered QRS feature signal; (vi) Rejection of erroneous QRS complexes.	(4)
wqrs	**wqrs:** Transformation to a curve-length signal to apply an adaptive threshold: (i) low-pass filtering; (ii) non-linear scaling of the signal to amplify the QRS complex and reduce noise; (iii) an adaptive threshold reveals the onset and duration of the QRS complex.	(28)

Table 2
Datasets

Abbreviation	Description	No. Beats	No. Recordings	Recording Duration (min)	Total Time (min)	Sampling Frequency (Hz)	Source
			Supervised ECG recordings				
SIN	Recordings from patients without arrhythmias: excerpts from long-term recordings collected at an Arrhythmia Laboratory.	185,253	18	120	2,160	500	MIT-BIH NSR database (16)
ARR	Recordings from patients with and without arrhythmias: excerpts from 24-hour ambulatory recordings collected at an Arrhythmia Laboratory.	112,599	48	30	1,440	360	MIT-BIH arrhythmia database (29; 16)
HIGH	High-quality recordings from patients and healthy volunteers: collected from multimodal devices such as bedside monitors.	72,315	100	10	1,000	250	2014 PhysioNet/CinC challenge training set (30; 16)
LOW	Low-quality recordings from patients and healthy volunteers: collected from multimodal devices such as bedside monitors.	78,518	100	10	1,000	360	2014 PhysioNet/CinC challenge augmented training set (30; 16)
			Unsupervised, telehealth ECG recordings				
TELE	Single-lead, telehealth ECGs from home-dwelling patients: collected by patients without supervision using a device which records the ECG from the hands.	5,932	250	0.50	125	500	Harvard dataverse TELE database (4)
SAFER	Single-lead, telehealth ECGs from home-dwelling AF screening participants: collected by participants without supervision using a handheld device. Split into subsets according to presence of AF (AF or non-AF) and ECG quality (HIGH or LOW):	18,279	479	0.50	239.5	500	SAFER Feasibility Study (private) (31)
	- SAFER-AF-HIGH	8,456	183	0.50	91.5	500	
	- SAFER-nonAF-HIGH	7,065	199	0.50	99.5	500	
	- SAFER-nonAF-LOW	2,758	97	0.50	48.5	500	

clinical supervision using the TeleMedCare Health Monitor (TeleMedCare Pty. Ltd. Sydney, Australia). The device records an ECG from the hands using dry metal electrodes. This dataset contains 221 ECGs randomly selected from 120 patients, and an additional 29 ECGs specifically selected to represent poor-quality data. The dataset contains manual annotations of QRS complexes. One ECG in the dataset lasted longer than 30s, and was truncated to 30s for this study.

2.2.5. SAFER ECG Dataset (SAFER)

The SAFER ECG Dataset contains 479 30-second lead-I ECG recordings from home-dwelling subjects aged 65 and over, collected in an AF screening study (the SAFER Feasibility Study, ISRCTN 16939438) (31).

ECG recordings were acquired without clinical supervision using the Zenicor EKG-2 device shown in Figure 1 (Zenicor Medical Systems AB, Sweden). The device records an ECG from the thumbs using dry metal electrodes. This dataset contains: 183 high-quality ECGs exhibiting AF (denoted SAFER-AF-HIGH) collected from 48 subjects (13 female and 35 male); 199 high-quality ECGs from subjects without AF (SAFER-nonAF-HIGH) collected from 199 participants (100 female and 99 male); and 97 low-quality ECGs from subjects without AF (SAFER-nonAF-LOW) collected from 97 subjects (49 female and 48 male). ECG quality was assessed using the Cardiolund ECG Parser algorithm (Cardiolund AB). R-peaks were manually annotated specifically for this study. The presence of AF was determined as described in (31): (i) using the Cardiolund algorithm to identify ECGs with potential abnormalities; and (ii) expert reviewers manually reviewing ECGs to identify AF (as described in (31; 33)). To provide further details, ECGs were classified as AF and non-AF based on *ad hoc* review by two cardiologists. An ECG was classified as AF if either: (i) both cardiologists agreed that the ECG contained AF; or (ii) one cardiologist made an AF diagnosis and the other provided no diagnosis. An ECG was classified as non-AF if either: (i) the Cardiolund algorithm didn't identify abnormalities in the ECG, and the cardiologists did not identify an arrhythmia, and the participant was not diagnosed with AF; or (ii) both cardiologists agreed that the ECG didn't contain an arrhythmia.

Figure 1: The handheld Zenicor-EKG device used to record 30-second ECGs in the SAFER ECG Dataset.

2.3. Ethics Statement

The SAFER Feasibility Study in which the SAFER ECG dataset was acquired was approved by the London Central NHS Research Ethics Committee (18/LO/2066). All participants gave written informed consent to participate in the study. The study was conducted in accordance with the Declaration of Helsinki. Ethical approval was not required for the use of the remaining datasets as these were pre-existing, anonymised datasets.

2.4. Statistical analysis

The performance of QRS detectors was primarily assessed using the F_1 score (following a precedent in (34; 35)). The F_1 score is the harmonic mean of the sensitivity (SEN) and positive predictive value (PPV). These three statistics were calculated as follows from: the number of reference QRS complex annotations (n_{ref}, corresponding to the number of actual positives, P); the number of QRS complexes identified by an algorithm (n_{alg}, corresponding to the number of predicted positives, *i.e.* true positives + false positives, TP+FP); and the number of QRS complexes which were correctly identified ($n_{correct}$, corresponding to the number of true positives, TP).

$$SEN(\%) = \frac{TP}{P} \times 100 = \frac{n_{correct}}{n_{ref}} \times 100 \quad (1)$$

$$PPV(\%) = \frac{TP}{TP + FP} \times 100 = \frac{n_{correct}}{n_{alg}} \times 100 \quad (2)$$

$$F_1(\%) = \frac{2 \times PPV \times SEN}{PPV + SEN} \times 100 \quad (3)$$

$n_{correct}$ was calculated as the number of reference QRS complex annotations for which at least one QRS complex was identified by an algorithm within ± 75ms of the reference QRS annotation as shown in Figure 2.

A threshold of ± 75ms was chosen to classify QRS detections as correct or not for the following reasons. QRS complexes typically last <120ms in health (36), although can last longer in disease (37). ± 75ms was identified as a conservative threshold which would classify any QRS detections lying on a QRS complex as correct, whilst classifying any detections on other ECG waves (such as p- or t-waves) as incorrect. This was based on the assumptions that: a QRS complex lasts up to approximately 150ms; the R-wave is located approximately in the centre of a QRS complex; and reference QRS annotations are at the locations of R-waves. To investigate the suitability of this threshold, we assessed the performance of the QRS detectors on the telehealth (TELE and SAFER) datasets for thresholds ranging from ± 1 to 140ms. The results (shown in Appendix A.4) show that for all QRS detectors performance was poorer at low thresholds, with performance generally approaching a maximum between 20 and 100ms (such as ≈ 20ms for *nk* and *unsw*, and ≈ 60-80ms for *pan-tomp* and *two-avg*). Therefore, a threshold of ± 75ms appeared a reasonable choice. In comparison, previous work in this area has used tolerances of 50 ms (5) and ± 150 ms (19).

F_1 scores are reported using the median and interquartile range of the F_1 score for each ECG window.

Two additional performance measures were used: algorithm error rate and execution time. Error rates were defined as the percentage of 30s ECG segments in which an algorithm encountered an error and did not return identified QRS complexes. Execution times were assessed as the median time taken for an algorithm to process each 30s ECG segment. The analysis was performed on a MacBook Air (M1, 2020, 16 GB RAM, 8 cores) without parallelization. The assessment was run in Visual Studio Code 1.73.0, using Python 3.9, and calling MATLAB R2022a for QRS detectors written in MATLAB code.

The two-sided Mann-Whitney U test was used to test for statistically significant differences between F_1 scores at the 95% significance level. A Bonferroni correction was used to account for the multiple comparisons (a comparison for each beat detector). This test was used as the distributions were neither normally distributed nor dependent on each other. Comparisons were made between: (i) supervised and telehealth ECGs; (ii) high- and low-quality ECGs; (iii) AF and non-AF ECGs; and (iv) female and male subjects. Comparisons between female and male subjects were made on the SIN, ARR and SAFER datasets, but not on the HIGH, LOW and TELE datasets as to the best of our knowledge they do not contain information on gender.

3. Results

3.1. Algorithm performance

The performance of the algorithms is presented in Figure 3 using the F_1 score. When using a F_1 score of ≥ 0.96 to identify good performance, a total of 12 out of 18 algorithms performed well on ECGs collected under clinical supervision (ARR, HIGH and LOW, and SIN). The exceptions were *engz*, *gamb*, *jqrs*, *mart*, *nab* and *rpeak*. Fewer algorithms performed well on telehealth ECGs: five algorithms performed well on the TELE dataset (*gqrs*, *nk*, *rdeco*, *two-avg*, and *unsw*); six algorithms performed well on high-quality SAFER data (*fnvg*, *fwhvg*, *nk*, *rdeco*, *two-avg*, and *unsw*); and performance was considerably poorer on low-quality SAFER data, with only three algorithms scoring ≥ 0.78 (*fnvg*, *nk*, and *unsw*), and none scored higher than 0.84.

Therefore, overall the *nk* and *unsw* algorithms performed best, with consistently high F_1 scores on datasets of supervised ECG recordings, and the highest F_1 scores on self-recorded ECGs (TELE and SAFER datasets).

Additional results for the positive predictive value (PPV) and sensitivity (SEN) are provided in Figures 8 and 9 in Appendix A.5. These metrics show that: *gamb* performed poorly because of a low PPV, indicating that it falsely detected additional QRS complexes; and *mart*, and *engz* had

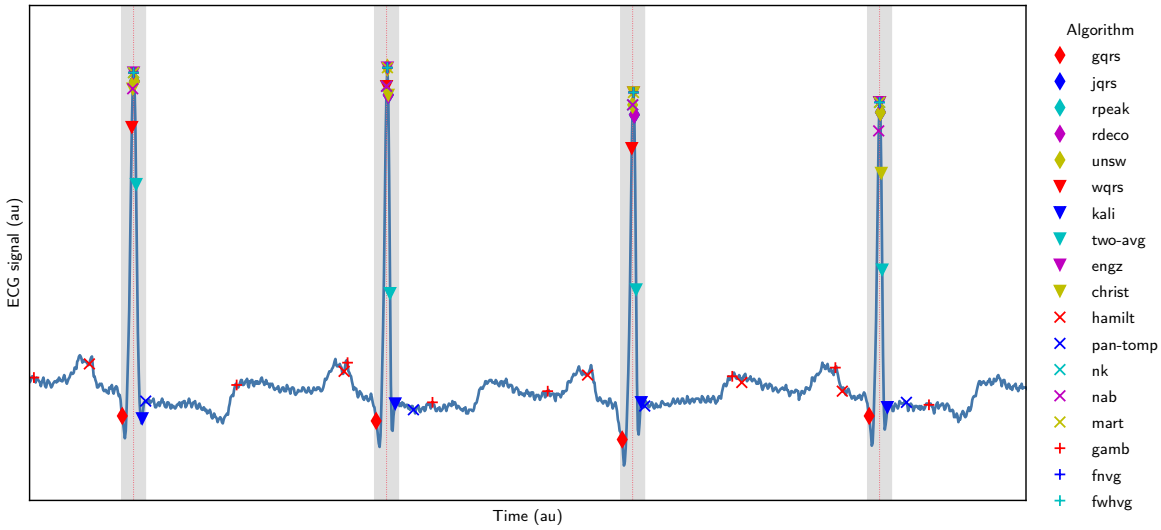


Figure 2: Assessing whether QRS complexes were correctly identified: An ECG signal is shown with dotted red lines marking reference R-peak annotations, grey areas showing the tolerance of ± 75ms around these annotations within which QRS complexes are deemed to be correctly identified, and markers for the R-peaks identified by the 18 QRS detectors used in this study.

a low SEN, indicating that they frequently missed QRS complexes.

Figure 4 shows the error rates of each QRS detector (indicating the proportion of ECG windows for which the QRS detector algorithms failed to execute - *i.e.* encountered an error). Most QRS detectors had no or very few errors. The best-performing algorithms had 0.0% errors on all datasets (*nk* and *unsw*). The *engz* and *gamb* algorithm implementations frequently produced errors, and some errors were encountered for *gqrs*, *jqrs*, *rdeco* and *rpeak* algorithms. Of particular note, the *gamb* algorithm exhibited higher error rates on SAFER data, including error rates of ≥99% for *gamb* (in keeping with a previous study (8)). This was due to the algorithm's use of a fixed amplitude threshold which was often not met for SAFER ECGs.

Figure 5 shows the median execution time of each QRS detector. The fastest QRS detector, *rpeak*, had an execution time of 1.1 ms (i.e. 0.004% of the signal duration). Of the best-performing QRS detectors (*nk* and *unsw*), *nk* had a short execution time of 2.7 ms (0.009% of the signal duration), whereas *unsw* was slower at 37.1 ms (0.124% of the signal duration). Four QRS detectors had much longer execution times (*christ*, *engz*, *gqrs*, and *wqrs*), although we note that C code implementations are available for some of these that would have led to shorter execution times. Most QRS detectors had similar median and mean execution times: the mean execution time was between 87 and 124% of the median for all QRS detectors except *christ*, whose median execution time was substantially longer (373% of the median), primarily due to exceptionally high runtimes on the SAFER-nonAF-LOW dataset.

3.2. Comparison between supervised and telehealth ECGs

For most QRS detectors, the performance of QRS detectors was higher on supervised ECG recordings than on unsupervised, telehealth ECGs. A total of 17 (out of 18) QRS detectors had a significantly higher F_1 score on the supervised SIN dataset than the unsupervised SAFER-nonAF-HIGH dataset (*mart* showed no sigificant difference). Similarly, 16 QRS detectors had a significantly higher F_1 score on the supervised ARR dataset than the unsupervised SAFER-AF-HIGH dataset (*hamilt* and *nk* showed no sigificant difference). Referring to Figure 3: some QRS detectors performed below average on unsupervised ECGs despite having performed well on supervised ECGs: *gqrs* achieved F_1 scores of ≥0.98 on supervised ECGs (SIN, ARR, HIGH, LOW), but ≤0.70 on SAFER; and *rpeak* achieved ≥0.81 on supervised ECGs, but ≤0.53 on TELE and SAFER datasets.

The results for positive predictive value (PPV) and sensitivity (SEN) (in Appendix A.5) show that most QRS detectors which performed poorly on self-recorded ECGs had a low PPV, indicating false positive QRS detections. In addition, some QRS detectors had low sensivities, indicating unrecognized QRS complexes (e.g. *engz*, *gamb*, *jqrs*, *mart*, *nab*, and *wqrs*).

Algorithm errors predominantly occurred in unsupervised telehealth ECGs (see Figure 4).

3.3. The impact of signal quality

Low signal quality was associated with poorer performance of QRS detectors in the telehealth setting. The F_1 scores for all QRS detectors except *gamb* were significantly

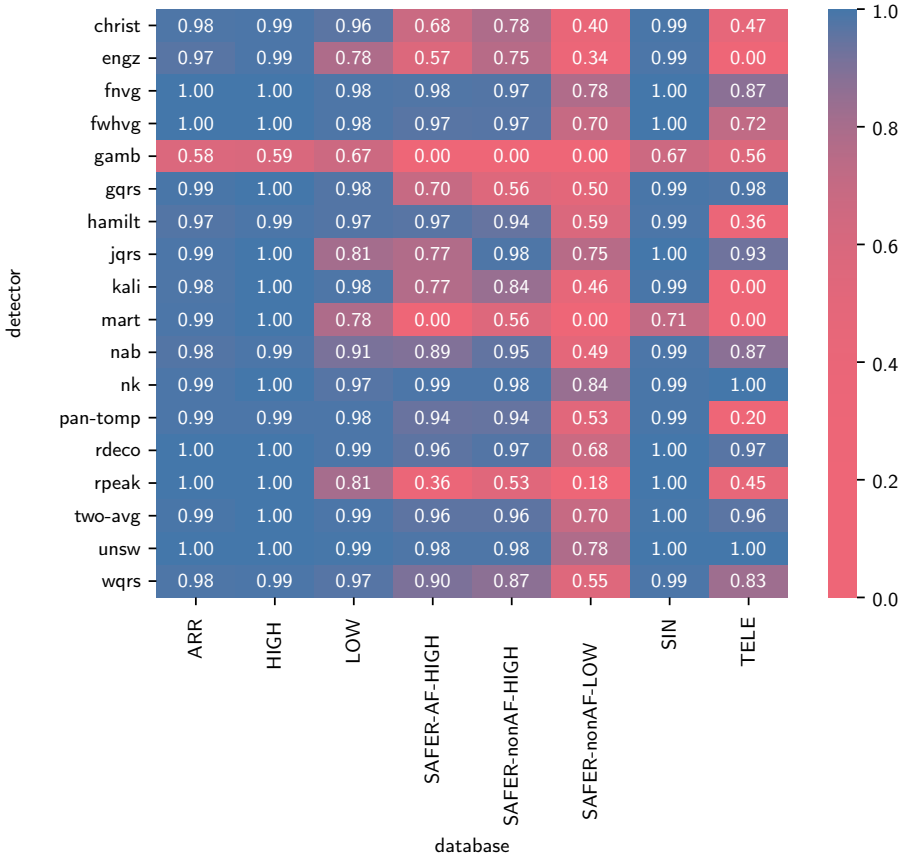


Figure 3: The performance of QRS detectors, expressed as the F_1 score. Results are shown for the 18 QRS detectors (on the y-axis) and the six datasets (x-axis).
Dataset definitions: ARR - MIT-BIH Arrhythmia Database; *HIGH* - PhysioNet/Computing in Cardiology Challenge 2014 training set; *LOW* - PhysioNet/Computing in Cardiology Challenge 2014 augmented training set; *SAFER-AF-HIGH* - SAFER ECG Dataset subset of high-quality ECGs exhibiting AF; *SAFER-nonAF-HIGH* - SAFER ECG Dataset subset of high-quality ECGs not exhibiting AF; *SAFER-nonAF-LOW* - SAFER ECG Dataset subset of low-quality ECGs not exhibiting AF; *SIN* - MIT-BIH Normal Sinus Rhythm Database; *TELE* - TELE ECG Database.

lower on low-quality unsupervised ECGs (SAFER-nonAF-LOW) than high-quality unsupervised ECGs (SAFER-nonAF-HIGH). For instance, the best-performing QRS detectors (*nk* and *unsw*) performed well on high-quality unsupervised ECGs (TELE, SAFER-nonAF-HIGH, SAFER-AF-HIGH) with F_1 scores of $\geq$0.97, but performed less well on low-quality unsupervised ECGs (SAFER-nonAF-LOW) with F_1 scores of $\leq$0.84. Indeed, all remaining QRS detectors showed F_1 scores of $\leq$0.78 on low-quality ECGs (SAFER-nonAF-LOW) in the unsupervised telehealth environment.

Signal quality had a smaller but nonetheless significant impact on QRS detectors when using supervised ECGs. Almost all algorithms performed well on high-quality ECGs (the SIN, ARR, and HIGH datasets) with F_1 scores of $\geq$0.97 (except *gamb* and *mart*), and most of these algorithms continued to perform relatively well on low-quality supervised ECGs (the LOW dataset) with F_1 scores of $\geq$0.97 (except *engz*, *gamb*, *jqrs*, *mart*, *nab* and *rpeak*). The small differences in F_1 scores between HIGH and LOW were significant for all QRS detectors except *wqrs* and *hamilt*.

3.4. Other influencing factors

The presence of arrhythmia did not have a large effect on F_1 scores for either supervised ECGs (comparing ARR and SIN) or unsupervised ECGs (comparing SAFER-AF-HIGH and SAFER-nonAF-HIGH) (see Figure 3). Whilst the differences were mostly small, F_1 scores were significantly lower during arrhythmias in ARR compared to SIN for 6 out of 18 QRS detectors, and in SAFER-AF-HIGH compared to SAFER-nonAF-HIGH for 8 QRS detectors. Amongst the best performing QRS detectors (*nk* and *unsw*), the only significant difference was for *nk* in the comparison of SAFER-AF-HIGH and SAFER-nonAF-HIGH, although this difference was small with median F_1 scores of 0.99 and 0.98 on the datasets.

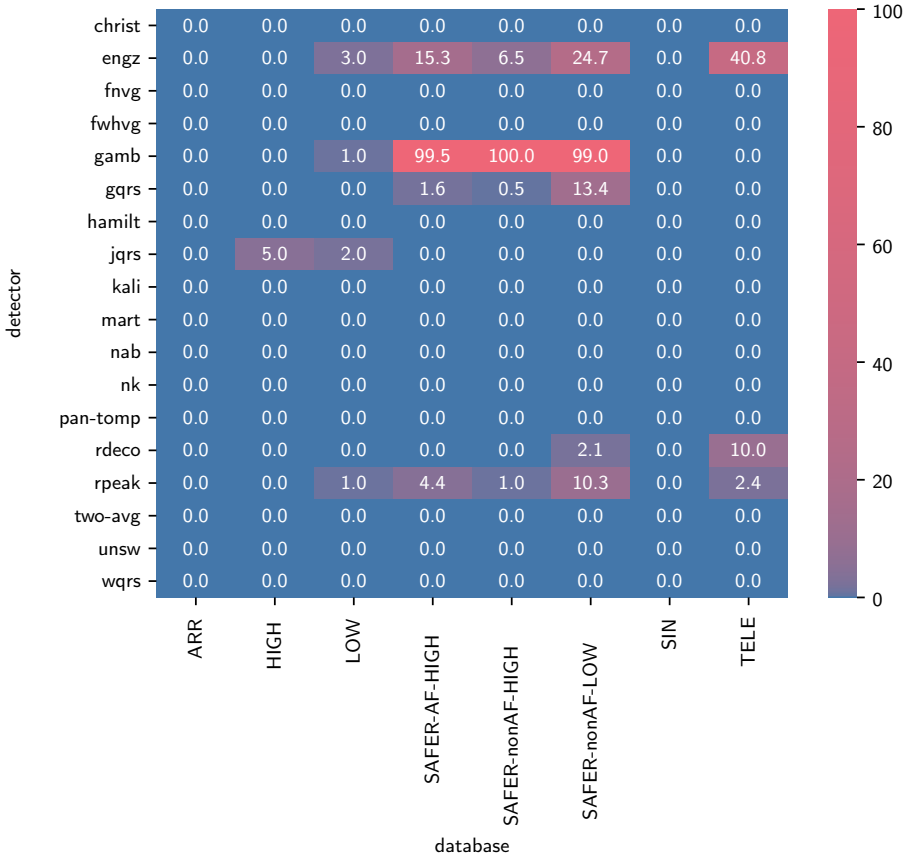


Figure 4: The error rates for each QRS detector (expressed as percentages). These indicate the proportion of ECG windows for which the QRS detector algorithms failed to execute (*i.e.* encountered an error).

Sex had little impact on performance when using unsupervised ECGs as demonstrated by there being no significant differences in performance between female and male subjects on high-quality, non-AF SAFER signals (see Figure 6a in Appendix A.6), and significant differences for only two QRS detectors on high-quality, AF SAFER signals (see Figure 6b). There were no significant differences in performance between sexes on the SIN and ARR datasets (see Figures 10a and 10b in Appendix A.6), although we note the small numbers of subjects in each group in the SIN dataset (13 female and 5 male).

p-values for all statistical comparisons are provided in Tables 6 and 7 in Appendix A.7.

4. Discussion

4.1. Summary of findings

This study assessed the performance of open-source QRS detectors on single-lead, telehealth ECGs. The neurokit (*nk*) and UNSW (*unsw*) QRS detectors were identified as the best-performing out of 18 QRS detectors. They performed well on telehealth ECGs recorded without clinical supervision, and also on ECGs recorded in clinical settings. They achieved F_1 scores of ≥0.98 on high-quality telehealth ECGs and ≥0.97 on ECGs recorded in clinical settings. Performance was lower at ≥0.78 when analysing low-quality telehealth ECGs. Performance was not substantially affected by heart rhythm or gender. *nk* had one of the fastest execution times (at 0.009% of the signal duration), whereas *unsw* was over ten times slower (0.124%).

4.2. Comparison with literature

Several studies have compared the performance of multiple QRS detection algorithms across databases of different quality (34; 19; 6; 8). Previous studies assessed 6-12 algorithms, compared to 18 in the current study. Several of the high-performing algorithms included in the current study were not widely assessed in previous comparison studies: *nk* and *two-avg* were only included in (8); *unsw* was only included in (34); and *rdeco* was not included in these studies. In addition, previous studies had mostly focused on assessing performance on supervised ECG recordings rather than the telehealth setting. Telehealth data was only included in (34): the current study included analyses of both this dataset and also data from the SAFER AF screening study, containing the additional challenge of QRS detection during AF.

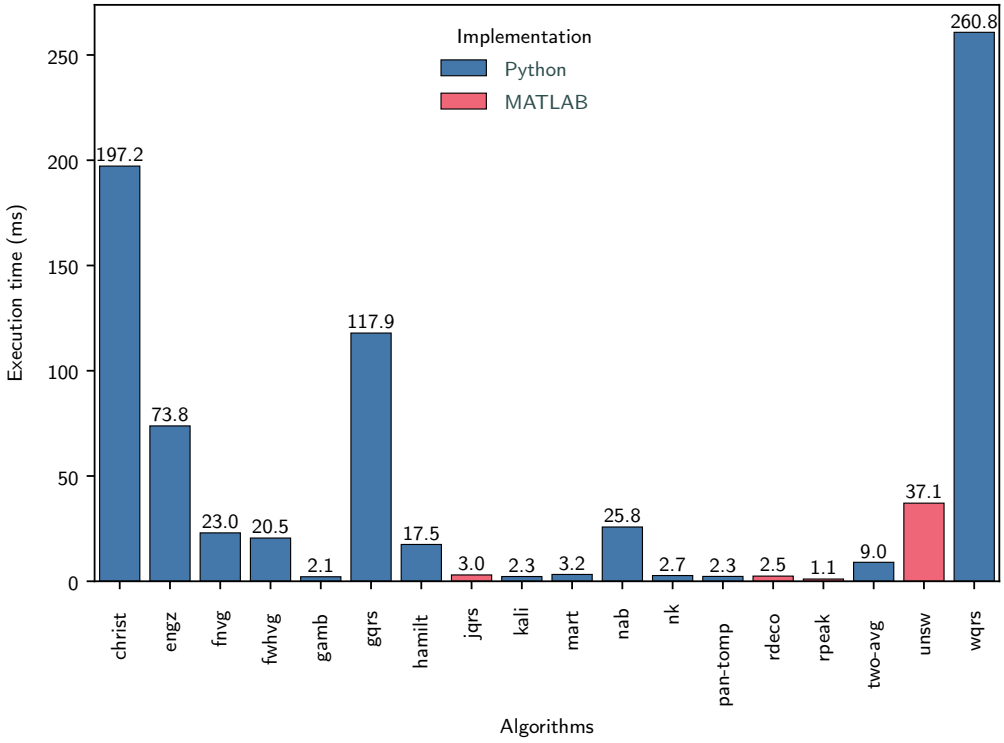


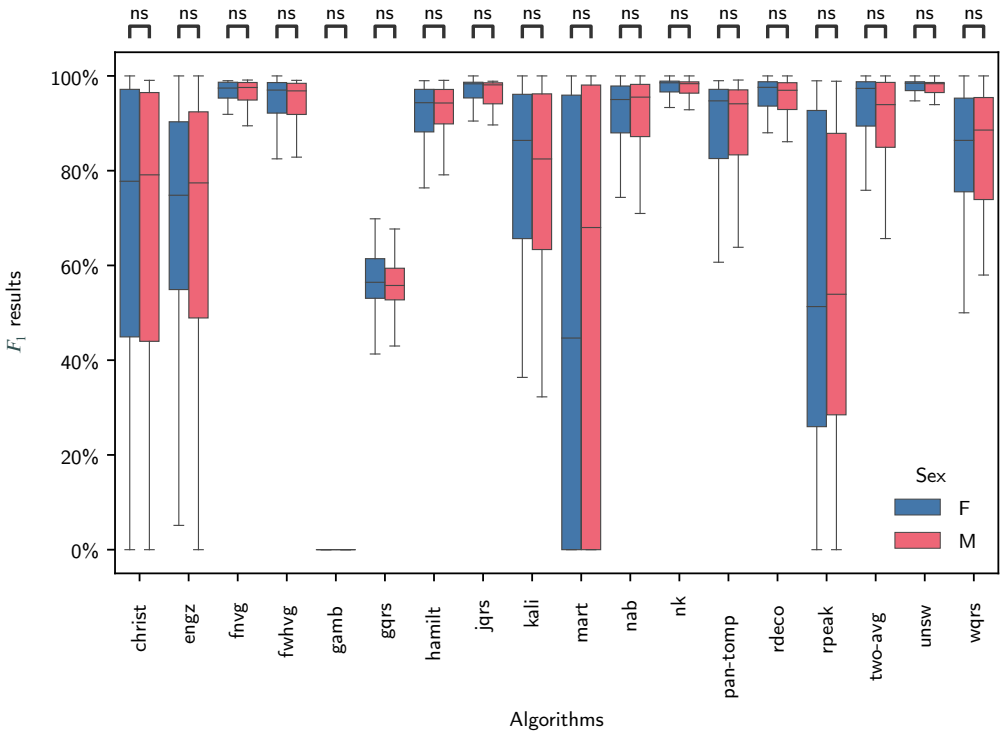
Figure 5: QRS detector execution times: The median execution time of each QRS detector was calculated across all datasets, where QRS detectors were implemented in either Python (blue) or Matlab (red).

The current study adds to our understanding of how best to detect QRS complexes in telehealth ECGs, and demonstrates the need to develop techniques to handle low-quality ECGs appropriately. Previously, QRS detectors had been found to perform worse on telehealth data, and in particular the TELE dataset (34). We also observed worse performance on telehealth data, although we found that the best QRS detectors performed adequately well on high-quality telehealth data, and that performance was only substantially worse on low-quality telehealth data. This provides two complementary directions for future work: (i) QRS detectors could be developed to perform well even in the presence of noise (*e.g.* through denoising (38) or improved algorithm design (4)); and (ii) ECG signal quality algorithms could be developed to identify low-quality recordings in which QRS complexes cannot be accurately identified (39; 19).

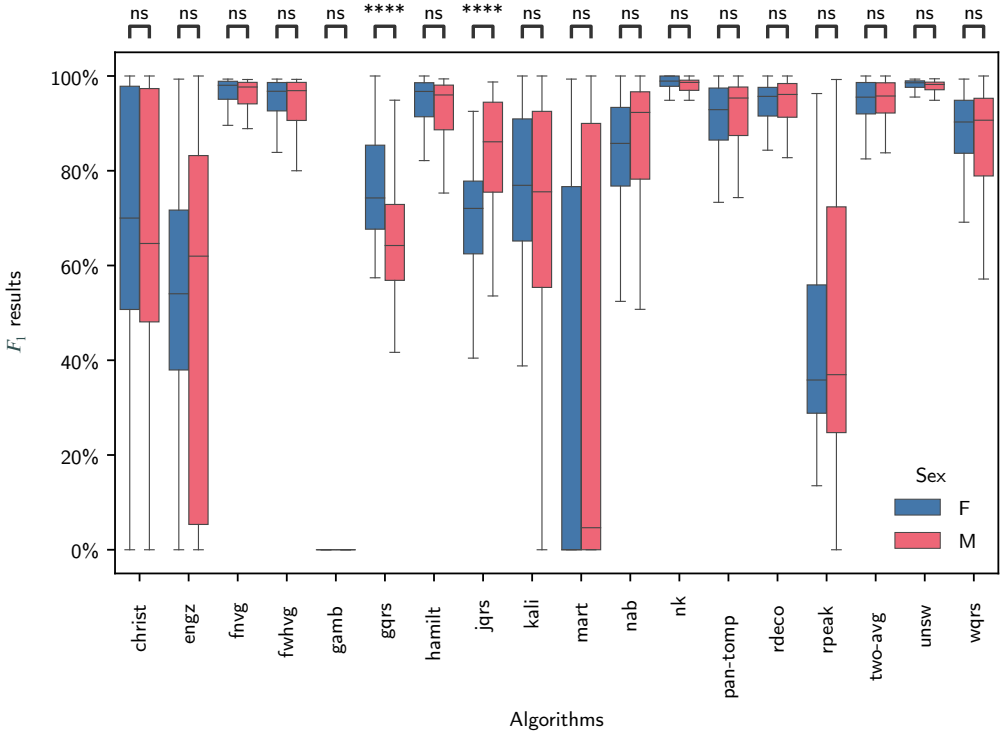
The current study also has implications for future research. We observed that the performance of QRS detectors on supervised or high-quality ECG recordings is not necessarily indicative of their performance on unsupervised recordings, in keeping with (6). This highlights the importance of assessing performance in the target setting, such as in AF screening as performed in this study. We also observed quite different performances on the TELE dataset to those reported previously: whereas the highest performing algorithm achieved an F_1 score of 0.80 on TELE in (5), six of the algorithms included in the present study achieved F_1 scores of 0.90-1.00. Whilst in many cases this may be explained by including additional algorithms in this study, it is notable that the *jqrs* algorithm's performance was substantially higher on this dataset in the present study (0.93) than the previous study (0.79). This may also be explained by the use of different tolerance windows. Nonetheless, this demonstrates the need to share open-source algorithm implementations and the code used to perform algorithm assessments. To address this, we have provided a repository of open-source algorithms and assessment code to accompany this article: `https://github.com/floriankri/ecg_detector_assessment`.

4.3. Strengths and limitations

The key strengths of this study are the assessment of QRS detectors in a real-world AF screening setting, and the inclusion of recently developed, high-performance QRS detectors. The key limitation is that algorithms were run retrospectively on a computer, rather than in real-time on a telehealth device. Some algorithms were implemented in Python, and others in Matlab. Therefore, the comparison of algorithm execution times reported in this study may not be truly representative of the relative execution times which would be observed on devices: the comparison of Python and Matlab execution times may not be fair; different algorithms may have been optimised to different extents; and some algorithms may be more amenable to further optimisation for use on devices than others (such as through implementation in C, as is already the case for parts of *unsw*). We note that in this study we did not investigate the potential benefit of



(a) SAFER-nonAF-HIGH: High-quality, non-AF ECGs (including 100 female and 99 male subjects).



(b) SAFER-AF-HIGH: High-quality, AF ECGs (including 92 female and 91 male subjects).

Figure 6: Comparison of the performance of QRS detectors between female (F) and male (M) SAFER participants.
Definitions: ns - no significant difference

additional ECG filtering beyond that already incorporated into each of the QRS detector algorithms: potentially performance could be improved further by including additional linear or non-linear filtering steps (40; 41). Furthermore, we did not investigate the accuracy of RR-intervals derived from QRS detections, nor their suitability for heart rate variability analysis or arrhythmia detection. We note that additional processing steps may be required to accurately derive RR-intervals, such as locating the R-wave on each detected QRS complex.

4.4. Implications

This study identified leading QRS detector algorithms for use with telehealth ECGs. The best-performing algorithms were able to detect QRS complexes with a very high degree of accuracy on high-quality telehealth ECG data, demonstrating the potential utility of telehealth devices for assessments based on RR-intervals (such as arrhythmia detection). Furthermore, the study demonstrates the importance of selecting a high-performance QRS detector, since performance can vary greatly on telehealth ECGs, between even well-established algorithms. The study also demonstrates the difficulty in analysing low-quality telehealth ECGs, which appear to be of particularly low quality, perhaps due to increased artifact, the use of dry electrodes, being self-recorded without clinical supervision, and acquisition at the hands rather than the chest (4).

The findings are particularly relevant to telehealth settings where ECG signals are recorded without clinical supervision. Several such settings arise in the detection and management of atrial fibrillation at home, including: (i) virtual wards to reduce hospitalisation for atrial fibrillation (42); (ii) screening for paroxysmal atrial fibrillation (43); and (iii) detecting recurrent atrial fibrillation after ablation or cardioversion (44). In each of these examples an accurate QRS detector is a key step in processing the intermittent ECGs acquired by patients at home, where signal quality may be lower than in the clinical setting.

5. Conclusion

This study identified two leading QRS detectors for use with single-lead, telehealth ECGs: the *nk* and *unsw* algorithms. These algorithms provided accurate QRS detection with fast execution times. Whilst most other algorithms performed well on data collected under clinical supervision, many did not perform as well on telehealth data, demonstrating the importance of selecting a high-performance algorithm for use in clinical analysis. The performance of even the leading algorithms was substantially lower on low-quality telehealth ECGs, highlighting the need to handle low-quality ECGs appropriately in an analysis pipeline. All the QRS detection algorithms used in this study are openly available, ensuring that they can be quickly used in future research. Furthermore, the code used to assess algorithm performance is also available to facilitate future research, at: `https://github.com/floriankri/ecg_detector_assessment`.

Declaration of competing interest

MRC is employed by Astrazeneca PLC. HCL is employed by Zenicor Medical Systems AB. The remaining authors declare no competing interests.

Acknowledgments

This study is funded by the British Heart Foundation (grant number FS/20/20/34626), the National Institute for Health and Care Research (NIHR) Programme Grants for Applied Research Programme (Reference Number RP-PG0217-20007), and the NIHR School for Primary Care Research (SPCR-2014-10043, project 410). The views expressed are those of the author(s) and not necessarily those of the NIHR or the Department of Health and Social Care.

(5) provided the foundations for the selection of datasets and their presentation in Table 2.

A. Appendix

A.1. Algorithm sources

Table 4 provides source links for the QRS detection algorithms used in this study.

A.2. Excluded algorithms

Table 5 summarises the QRS detectors which were excluded from this study. They were excluded for the following reasons:

- ***aristot***: Excluded because only a C implementation was found, and the algorithm did not perform competitively compared to *gqrs* in (6).
- ***match***: Excluded because the available implementation only accepted signals with a sampling frequency of either 250 Hz or 360 Hz.
- ***rodr***: Excluded because the available implementation was found to predominantly lead to errors, as reported in (8). Additionally, the results in the cases where no error occurred were not competitive.
- ***rsslope***: Excluded because the available implementation was found to predominantly lead to errors.
- ***visgraph***: Excluded due to a substantially longer execution time than other algorithms, which was deemed undesirable when evaluating signals in a mobile setting.
- ***zong***: Excluded because the available implementation consistently led to errors during evaluation.

A.3. Dataset sources

Table 3 provides links to the publicly available datasets used in this study. Note that the SAFER dataset is private, and further information on this dataset is available in (31).

A.4. Threshold selection

The results of the performance of QRS detectors for thresholds ranging from ± 40ms to ± 150ms are shown in Figure 7. Figure 7(a) and (b) show results for the TELE and SAFER datasets respectively. These were used to inform the choice of threshold to determine whether QRS complexes were correctly detected (see Section 2.4).

A.5. The positive predictive value and sensitivity of QRS detectors

Additional results for the positive predictive value (PPV) and sensitivity (SEN) of QRS detectors are provided in Figures 8 and 9.

A.6. Comparison between males and females

Additional results comparing performance between males and females in supervised ECGs are provided in Figure 10.

A.7. p-values for statistical tests

The p-values for comparisons between datasets are provided in Table 6. The p-values for comparisons between male and female subjects are provided in Table 7

References

[1] C. R. Lopez Perales *et al.*, "Mobile health applications for the detection of atrial fibrillation: a systematic review," *EP Europace*, vol. 23, no. 1, pp. 11–28, 2021.

[2] L. Lu *et al.*, "Wearable health devices in health care: narrative systematic review," *JMIR mHealth and uHealth*, vol. 8, no. 11, p. e18907, 2020.

[3] S. M. Lee and D. Lee, "Opportunities and challenges for contactless healthcare services in the post-covid-19 era," *Technological Forecasting and Social Change*, vol. 167, p. 120712, 2021.

[4] H. Khamis *et al.*, "QRS detection algorithm for telehealth electrocardiogram recordings," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 7, pp. 1377–1388, 2016.

[5] F. Liu *et al.*, "Performance analysis of ten common QRS detectors on different ECG application cases," *Journal of Healthcare Engineering*, vol. 2018, 2018.

[6] M. Llamedo and J. P. Martínez, "QRS detectors performance comparison in public databases," in *Computing in Cardiology*. IEEE, 2014, pp. 357–360.

[7] M. Elgendi, "Fast QRS Detection with an Optimized Knowledge-Based Method: Evaluation on 11 Standard ECG Databases," *PLoS ONE*, vol. 8, no. 9, 2013.

[8] D. Makowski *et al.*, "NeuroKit2: A python toolbox for neurophysiological signal processing," *Behavior Research Methods*, vol. 53, no. 4, pp. 1689–1696, 2021.

[9] B. Porr and L. Howell, "py-ecg-detectors: Seven ECG heartbeat detection algorithms and heartrate variability analysis," 2022, retrieved on: 2022-08-22. [Online]. Available: https://github.com/berndporr/py-ecg-qrs-detectors

[10] I. I. Christov, "Real time electrocardiogram QRS detection using combined adaptive threshold," *BioMedical Engineering OnLine*, vol. 3, no. 1, p. 28, 2004.

[11] W. A. Engelse and C. Zeelenberg, "A single scan algorithm for QRS-detection and feature extraction," *Computers in Cardiology*, vol. 6, pp. 37–42, 1979.

[12] A. Lourenco *et al.*, "Real time electrocardiogram segmentation for finger based ecg biometrics," in *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing*. SciTePress, 2012, pp. 49–54.

[13] J. Emrich *et al.*, "Accelerated sample-accurate r-peak detectors based on visibility graphs," in *European Signal Processing Conference*, 2023, pp. 1090–1094.

[14] T. Koka and M. Muma, "Fast and Sample Accurate R-Peak Detection for Noisy ECG Using Visibility Graphs," in *44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society*, 2022, pp. 121–126.

[15] H. Gamboa, "Multi-modal behavioral biometrics based on HCI and electrophysiology," 2008, phD Thesis, Universidade Técnica de Lisboa.

[16] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. E215–220, 2000.

[17] P. Hamilton, "Open source ECG analysis," in *Computers in Cardiology*. IEEE, 2002, pp. 101–104.

[18] P. S. Hamilton and W. J. Tompkins, "Quantitative investigation of QRS detection rules using the MIT/BIH arrhythmia database," *IEEE Transactions on Biomedical Engineering*, vol. BME-33, no. 12, pp. 1157–1165, 1986.

[19] A. E. W. Johnson *et al.*, "Multimodal heart beat detection using signal quality indices," *Physiological Measurement*, vol. 36, no. 8, pp. 1665–1677, 2015.

[20] J. Behar, J. Oster, and G. D. Clifford, "Non-invasive FECG extraction from a set of abdominal sensors," in *Computing in Cardiology*, 2013, pp. 297–300.

[21] J. Behar, J. Oster, and G. D. Clifford, "Combining and benchmarking methods of foetal ECG extraction without maternal or scalp electrode data," *Physiological Measurement*, vol. 35, no. 8, pp. 1569–1589, 2014.

[22] V. Kalidas and L. Tamil, "Real-time QRS detector using stationary wavelet transform for automated ECG analysis," in *2017 IEEE 17th International Conference on Bioinformatics and Bioengineering*, 2017, pp. 457–461.

[23] J. Martinez *et al.*, "A wavelet-based ECG delineator: evaluation on standard databases," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 4, pp. 570–581, 2004.

[24] M. Nabian *et al.*, "An open-source feature extraction tool for the analysis of peripheral physiological data," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 6, p. 2800711, 2018.

[25] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Transactions on Biomedical Engineering*, vol. BME-32, no. 3, pp. 230–236, 1985.

[26] J. Moeyersons *et al.*, "R-DECO: an open-source matlab based graphical user interface for the detection and correction of r-peaks," *PeerJ Computer Science*, vol. 5, p. e226, 2019.

[27] M. Elgendi, M. Jonkman, and F. DeBoer, "Frequency bands effects on qrs detection," in *International Conference on Bio-inspired Systems and Signal Processing*. SciTePress, 2010, pp. 428–431.

[28] W. Zong, G. Moody, and D. Jiang, "A robust open-source algorithm to detect onset and duration of QRS complexes," in *Computers in Cardiology*, 2003, pp. 737–740.

[29] 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.

[30] G. Moody, B. Moody, and I. Silva, "Robust detection of heart beats in multimodal data: The PhysioNet/computing in cardiology challenge 2014," in *Computing in Cardiology 2014*, 2014, pp. 549–552.

[31] M. Pandiaraja *et al.*, "Screening for atrial fibrillation: Improving efficiency of manual review of handheld electrocardiograms," *Engineering Proceedings*, vol. 2, no. 1, p. 78, 2020.

[32] S. J. Redmond *et al.*, "Electrocardiogram signal quality measures for unsupervised telehealth environments," *Physiological Measurement*, vol. 33, no. 9, pp. 1517–1533, 2012.

[33] M. Adeniji *et al.*, "Prioritising electrocardiograms for manual review to improve the efficiency of atrial fibrillation screening," in *Proc IEEE EMBS*. IEEE, 2022, pp. 3239–3242.

[34] F. Liu *et al.*, "Performance analysis of ten common QRS detectors on different ECG application cases," *Journal of Healthcare Engineering*, vol. 2018, p. e9050812, 2018.

[35] P. Laguna, R. Jané, and P. Caminal, "Automatic detection of wave boundaries in multilead ECG signals: Validation with the CSE database," *Computers and Biomedical Research*, vol. 27, no. 1, pp. 45–60, 1994.

[36] K. Hnatkova *et al.*, "Sex and race differences in QRS duration," *EP Europace*, vol. 18, no. 12, pp. 1842–1849, 2016. [Online]. Available: https://doi.org/10.1093/europace/euw065

[37] N. C. Wang *et al.*, "Clinical implications of QRS duration in patients hospitalized with worsening heart failure and reduced left ventricular ejection fraction," *JAMA*, vol. 299, no. 22, pp. 2656–2666, 2008. [Online]. Available: https://doi.org/10.1001/jama.299.22.2656

[38] N. H. Beni and N. Jiang, "Heartbeat detection from single-lead ECG contaminated with simulated EMG at different intensity levels: A comparative study," *Biomedical Signal Processing and Control*, vol. 83, p. 104612, 2023.

[39] F. Liu *et al.*, "Dynamic ECG Signal Quality Evaluation Based on the Generalized bSQI Index," *IEEE Access*, vol. 6, pp. 41 892–41 902, 2018.

[40] G. D. Clifford, "Linear filtering methods," in *Advanced Methods and Tools For ECG Data Analysis*. Artech, 2006, pp. 135–170.

[41] P. E. McSharry and G. D. Clifford, "Nonlinear filtering methods," in *Advanced Methods and Tools For ECG Data Analysis*. Artech, 2006, pp. 171–196.

[42] A. Kotb *et al.*, "Digitally enabled acute care for atrial fibrillation: conception, feasibility and early outcomes of an AF virtual ward," *Open Heart*, vol. 10, no. 1, p. e002272, 2023.

[43] E. Svennberg *et al.*, "Mass screening for untreated atrial fibrillation: the STROKESTOP study," *Circulation*, vol. 131, no. 25, pp. 2176–2184, 2015.

[44] I. L. Goldenthal *et al.*, "Recurrent atrial fibrillation/flutter detection after ablation or cardioversion using the AliveCor KardiaMobile device: iHEART results," *Journal of Cardiovascular Electrophysiology*, vol. 30, no. 11, pp. 2220–2228, 2019.

[45] G. B. Moody and R. G. Mark, "Development and evaluation of a 2-lead ECG analysis program," *Computers in cardiology*, vol. 9, pp. 39–44, 1982.

[46] T. Rodrigues *et al.*, "A low-complexity r-peak detection algorithm with adaptive thresholding for wearable devices," in *25th International Conference on Pattern Recognition*, 2021, pp. 1–8.

[47] R. Gutierrez-Rivas *et al.*, "Novel real-time low-complexity QRS complex detector based on adaptive thresholding," *IEEE Sensors Journal*, vol. 15, no. 10, pp. 6036–6043, 2015.

[48] P. Podziemski and J. Gieraltowski, "Fetal heart rate discovery: Algorithm for detection of fetal heart rate from noisy, noninvasive fetal ECG recordings," in *Computing in Cardiology*, 2013, pp. 333–336.

[49] W. Zong *et al.*, "An open-source algorithm to detect onset of arterial blood pressure pulses," in *Computers in Cardiology, 2003*, 2003, pp. 259–262.

Table 3
Links to the publicly available datasets used in this study.

Database	Full Source Link
ARR	https://www.physionet.org/physiobank/database/mitdb/
HIGH	https://physionet.org/content/challenge-2014/1.0.0/
LOW	https://physionet.org/content/challenge-2014/1.0.0/
SIN	https://physionet.org/physiobank/database/nsrdb/
TELE	https://doi.org/10.7910/DVN/QTG0EP

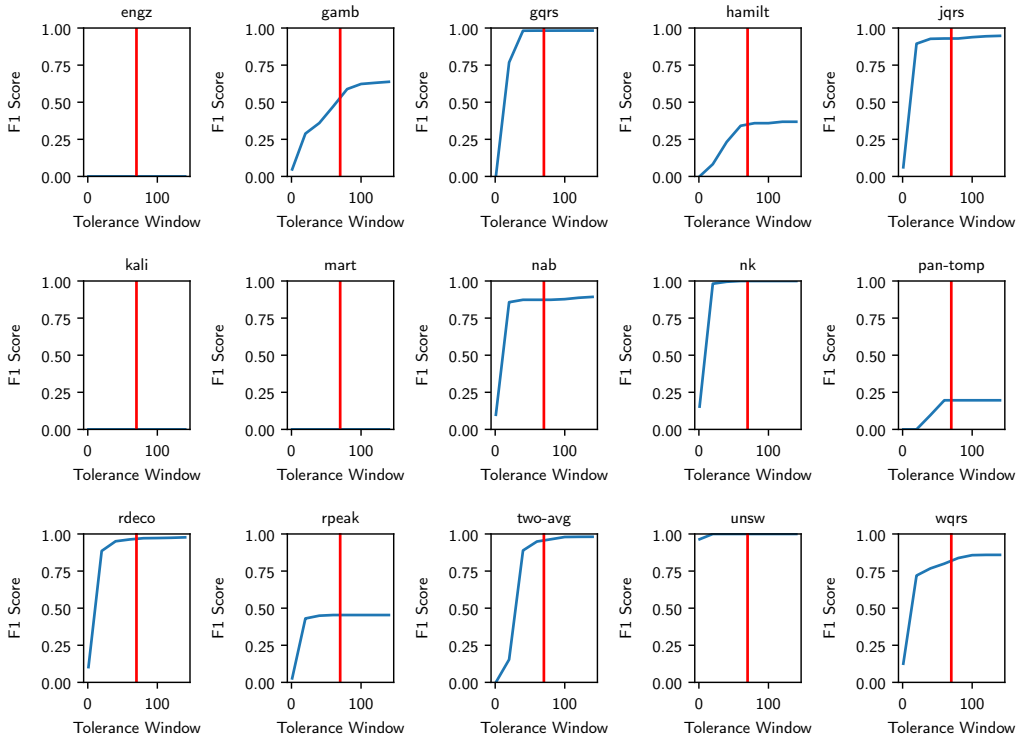
Table 4
Source links for the QRS detection algorithms.

Shortname	Full Source Link
aristot	https://archive.physionet.org/physiotools/activity/cic92/node9.html
christ	https://github.com/berndporr/py-ecg-detectors
engz	https://github.com/berndporr/py-ecg-detectors/blob/master/ecgdetectors.py
fnvg	https://github.com/JonasEmrich/vg-beat-detectors
fwhvg	https://github.com/JonasEmrich/vg-beat-detectors
gamb	https://github.com/PIA-Group/BioSPPy/blob/master/biosppy/signals/ecg.py#L923
gqrs	https://www.physionet.org/content/wfdb-python/3.3.0/#files-panel
hamilt	https://github.com/berndporr/py-ecg-detectors
jqrs	https://github.com/alistairewj/peak-detector/blob/master/sources/qrs_detect2.m
kali	https://github.com/neuropsychology/NeuroKit/blob/master/neurokit2/ecg/ecg_peaks.py
mart	https://github.com/neuropsychology/NeuroKit/blob/master/neurokit2/ecg/ecg_peaks.py
match	https://github.com/berndporr/py-ecg-detectors/blob/master/ecgdetectors.py
nab	https://github.com/neuropsychology/NeuroKit/blob/master/neurokit2/ecg/ecg_peaks.py
nk	https://github.com/neuropsychology/NeuroKit/blob/master/neurokit2/ecg/ecg_peaks.py
pan-tomp	https://github.com/neuropsychology/NeuroKit/blob/master/neurokit2/ecg/ecg_peaks.py
rdeco	https://physionet.org/content/r-deco/1.0.0/R_peak_detection/Algorithm/peak_detection.m
rodr	https://github.com/neuropsychology/NeuroKit/blob/master/neurokit2/ecg/ecg_peaks.py
rpeak	http://www.mit.edu/~gari/CODE/ECGtools/ecgBag/rpeakdetect.m
rsslope	https://archive.physionet.org/challenge/2013/sources/
two-avg	https://github.com/berndporr/py-ecg-detectors/blob/master/ecgdetectors.py
unsw	https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/QTG0EP
visgraph	https://github.com/taulokoka/visgraphdetector/blob/main/visgraphdetector.py
wqrs	https://github.com/berndporr/py-ecg-detectors/blob/master/ecgdetectors.py
zong	https://github.com/PIA-Group/BioSPPy/blob/master/biosppy/signals/ecg.py#L513

Table 5
Excluded QRS detection algorithms

Abbreviation	Name	Reference(s)
aristot	aristotle	(45; 16)
match	matched filter	(25; 9)
rodr	rodrigues	(46; 47; 8)
rsslope	rs-slope	(48)
visgraph	visgraphdetector	(14)
zong	zong, Slope Sum Function (ssf)	(49; 8)

(a) TELE dataset



(b) SAFER dataset

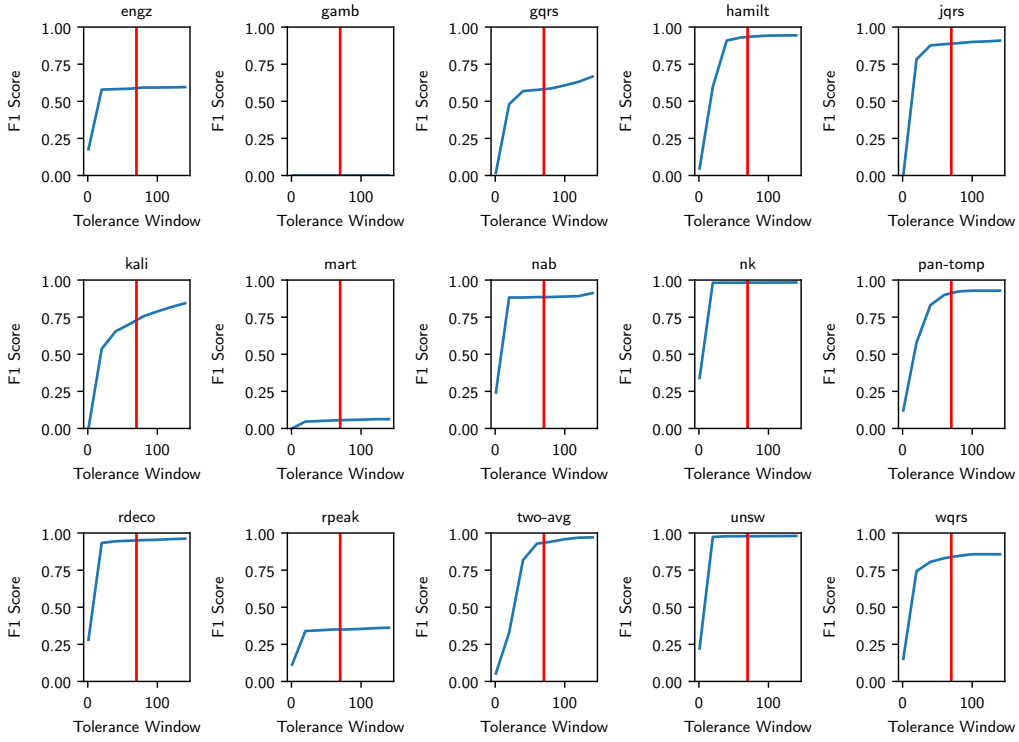


Figure 7: The performance of QRS detectors when using different thresholds to identify correct QRS detections. Each plot shows the performance of a QRS detector when assessed using thresholds varying from 1 to 140ms.

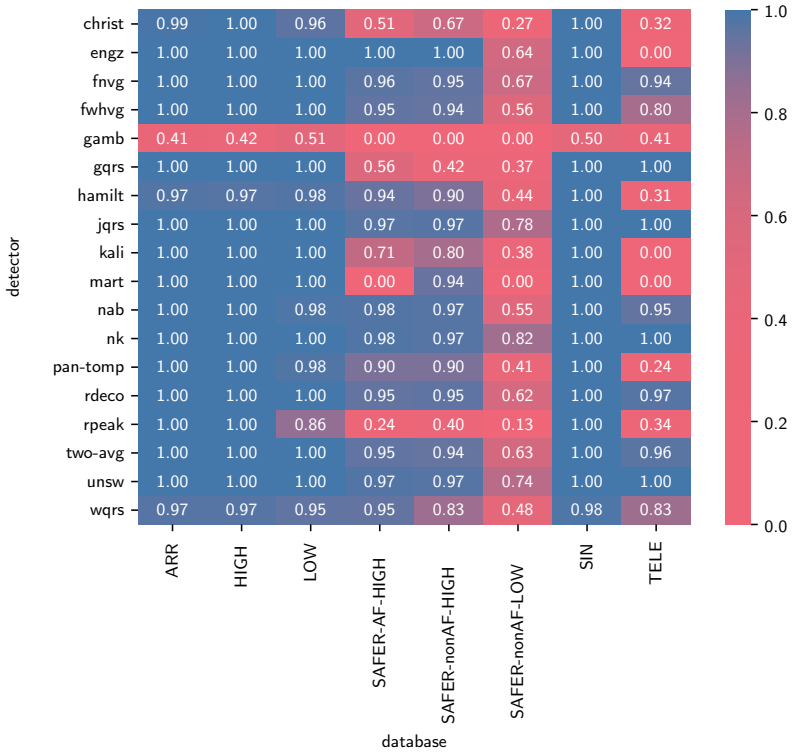


Figure 8: The positive predictive value (PPV) of QRS detectors.

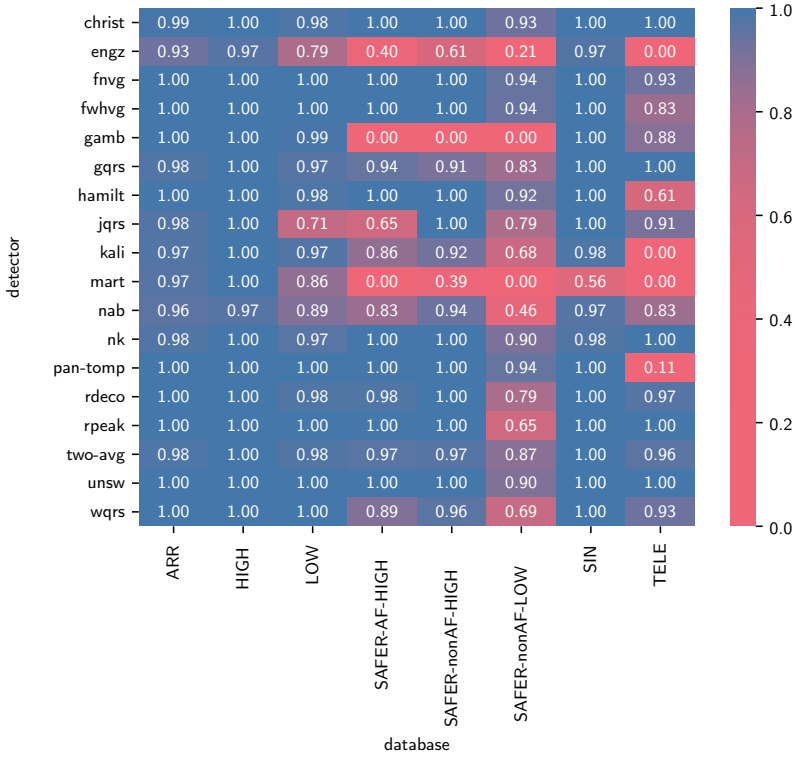
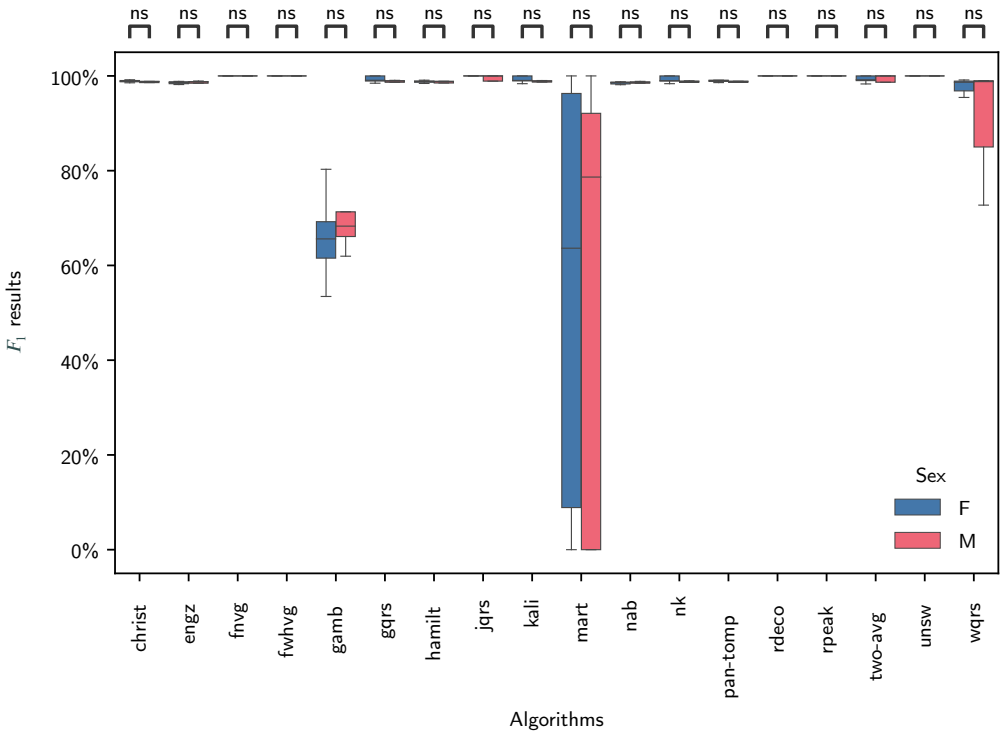
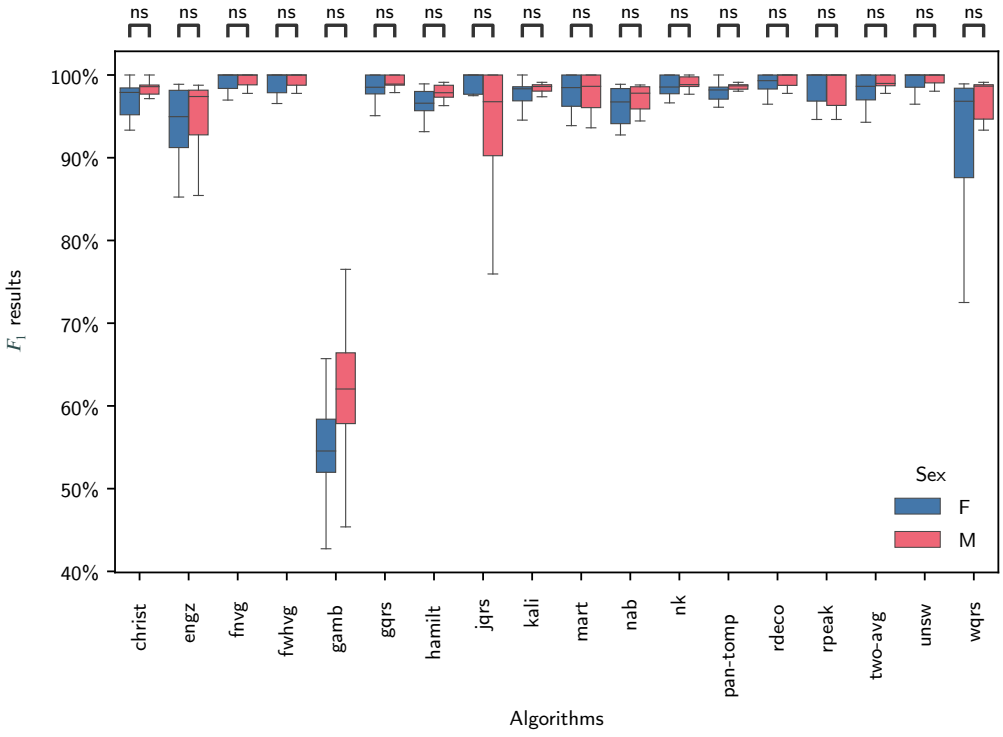


Figure 9: The sensitivity (SEN) of QRS detectors.



(a) SIN database (including 13 female subjects and 5 male subjects).



(b) ARR database (including 22 female and 26 male subjects).

Figure 10: Comparison of the performance of QRS detectors between female (F) and male (M) subjects on supervised ECGs. *Definitions: * - significant difference; ns - no significant difference.*

Table 6
p-values for comparisons between datasets. * *indicates a significant p-value after applying a Bonferoni correction for multiple comparisons.*

QRS Detector	Comparison					
	ARR vs. **SIN**	**SAFER-AF-HIGH** vs. **SAFER-nonAF-HIGH**	**SIN** vs. **SAFER-nonAF-HIGH**	**ARR** vs. **SAFER-AF-HIGH**	**SAFER-nonAF-HIGH** vs. **SAFER-nonAF-LOW**	**HIGH** vs. **LOW**
christ	0.004	0.218	0.000*	0.000*	0.000*	0.000*
engz	0.000*	0.000*	0.000*	0.000*	0.000*	0.000*
fnvg	0.027	0.157	0.000*	0.000*	0.000*	0.000*
fwhvg	0.070	0.276	0.000*	0.000*	0.000*	0.000*
gamb	0.001*	0.299	0.000*	0.000*	1.000	0.000*
gqrs	0.513	0.000*	0.000*	0.000*	0.000*	0.000*
hamilt	0.001*	0.002*	0.000*	0.016	0.000*	0.021
jqrs	0.058	0.000*	0.000*	0.000*	0.000*	0.000*
kali	0.000*	0.010	0.000*	0.000*	0.000*	0.000*
mart	0.000*	0.000*	0.535	0.000*	0.000*	0.000*
nab	0.004	0.000*	0.000*	0.000*	0.000*	0.000*
nk	0.474	0.000*	0.003*	0.397	0.000*	0.000*
pan-tomp	0.002*	0.232	0.000*	0.000*	0.000*	0.000*
rdeco	0.110	0.001*	0.000*	0.000*	0.000*	0.000*
rpeak	0.041	0.018	0.000*	0.000*	0.000*	0.000*
two-avg	0.140	0.344	0.000*	0.000*	0.000*	0.000*
unsw	0.300	0.055	0.000*	0.000*	0.000*	0.000*
wqrs	0.162	0.015	0.000*	0.000*	0.000*	0.202

Table 7
p-values for comparisons between males and females. * *indicates a significant p-value after applying a Bonferoni correction for multiple comparisons.*

QRS Detector	Dataset			
	SIN	**ARR**	**SAFER-nonAF-HIGH**	**SAFER-AF-HIGH**
christ	1.000	1.000	1.000	1.000
engz	1.000	1.000	1.000	1.000
fnvg	1.000	1.000	1.000	1.000
fwhvg	1.000	1.000	1.000	1.000
gamb	1.000	0.180	1.000	1.000
gqrs	1.000	1.000	1.000	0.000*
hamilt	1.000	0.457	1.000	1.000
jqrs	1.000	1.000	1.000	0.000*
kali	1.000	1.000	1.000	1.000
mart	1.000	1.000	1.000	1.000
nab	1.000	1.000	1.000	0.534
nk	0.671	1.000	1.000	1.000
pan-tomp	1.000	0.173	1.000	1.000
rdeco	1.000	1.000	1.000	1.000
rpeak	1.000	1.000	1.000	1.000
two-avg	1.000	1.000	1.000	1.000
unsw	1.000	1.000	1.000	0.317
wqrs	1.000	0.610	1.000	1.000