RecogNoise: Machine-Learning-Based Recognition of Noisy Segments in Electrocardiogram Signals

Amin Aminifar[1], Soheil Khooyooz[1], Anice Jahanjoo[2], Salar Shakibhamedan[2], Nima TaheriNejad[1,2]
[1]Heidelberg University, Germany [2]Technische Universität Wien (TU Wien), Austria

***Abstract*—Today, wearable technology is frequently used for continuous monitoring of physiological indicators in the healthcare domain. However, mobile-health and wearable devices are generally used in ambulatory settings, hence vulnerable to noise. This interferes with the accuracy of Machine Learning (ML) models running on such systems and their decision-making procedures. To address this issue, we first need to identify the presence of noise. In this paper, we propose RecogNoise to detect noisy segments in Electrocardiography (ECG) recordings using heartbeat detection algorithms and ML. We evaluate our approach based on the MIT-BIH arrhythmia database and three types of noise, i.e., Electrode Motion (EM) , Baseline Wander (BW), and Muscle Artifact (MA), with different Signal to Noise Ratios (SNRs). We show that RecogNoise can detect noisy segments with an F1-score of 86.9% and an accuracy of 88.3%.**

I. Introduction

Wearable devices provide the opportunity to monitor the physiological parameters of individuals on a continuous basis, often in ambulatory settings. This provides several advantages, such as early detection of health problems, providing personalized solutions, and remote monitoring of patients [1], [2], [3], [4], [5], [6]. However, the recordings acquired by wearable technologies are far more prone to noise compared to those acquired in hospital environment and using hospital equipment [7], [8], [9], [10], [11]. This is a significant problem in the healthcare domain. If the wearable system has a high false positive rate, the patients and healthcare providers may not pay enough attention to the system's alarms or recommendations. In certain applications, e.g., detection of myocardial infarction or seizure detection [12], [13], high false negative rates could lead to catastrophic events.

To address the problem of noise in biosignals, the first step is to identify the presence of noise within the signal. In [14], [11], [15], the authors address the noise detection problem in Photoplethysmogram (PPG) signals. For instance, in [14], they use variable frequency complex demodulation to detect noise artifacts in pulsatile signals obtained from fingertip videos, with the aim of improving atrial fibrillation detection accuracy, or in [15], the authors employ an autoencoder to detect and remove motion and noise artifacts in PPG signals. Similarly, several studies address noise detection in Electroencephalogram (EEG) signals [16], [17], [18]. For instance, in [18], the authors embed an autoregressive model into the Kalman filter for state noise detection.

For ECG signals, there have also been several studies that aim to detect noise [19], [20], [21], [22]. In previous noise detection algorithms for ECG, one difficulty is that several hyperparameters need to be taken into account for different settings. This is not always possible as, in the test phase, we lack accurate prior knowledge of what patterns the signal (test sample) might include or the type and intensity of noise that contaminates the signal. For instance, in [21], the authors propose clustering the signal segments to identify noise in them. In this method, selecting the features, the parameters of the feature generation phase, and the number of clusters are among the parameters that have to be set. Another complication in previous methods, e.g., in [19], [20], is their dependence on additional physiological signals, such as signals collected from accelerometers.

In this paper, we propose RecogNoise, which is a machine-learning-based technique for the detection of noisy segments in ECG. Our technique employs several heartbeat detection algorithms to identify R-peaks in the ECG segments. Based on the R-peak information, we generate the feature set for an ML algorithm. Then, we train a model to classify segments into noisy and non-noisy classes. RecogNoise does not require setting parameters to achieve high classification performance and is merely dependent on ECG recording, and not other signals collected from accelerometer sensors, to detect noise. We evaluate our technique based on the popular and publicly available MIT-BIH arrhythmia database [23] and three types of noise, i.e., Electrode Motion (EM), Baseline Wander (BW), and Muscle Artifact (MA) [24]. Although, in this study, we focus on R-peaks and ECG recordings, RecogNoise is not limited to them and can be extended to other fiducial points and other physiological signals, for instance, PPG signal.

The rest of this paper is organized as follows. In Section II, we present the details of our technique. In Section III, we provide the experimental results that support our findings in this paper and discuss the results. Finally, Section IV serves as the conclusion of this paper.

II. RecogNoise: ML-Based Recognition of Noisy Segments in ECG Signals

In this section, we discuss the underlying idea behind RecogNoise and explain the details of our technique afterward. While this approach is not limited in terms of the type of signal, here, we focus on ECG signal for the simplicity of presentation. Noise disturbs the ECG signal, and the extent of

This work has been partially funded by the Vienna Science and Technology Fund (WWTF) [10.47379/ICT20034].

The authors gratefully acknowledge funding from European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreement No. 956090 (APROPOS: Approximate Computing for Power and Energy Optimisation, http://www.apropos.eu/).

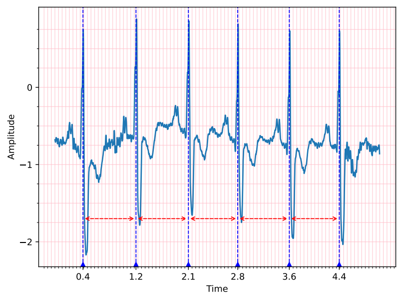


Fig. 1: Clean Signal and the Ground Truth R-Peaks

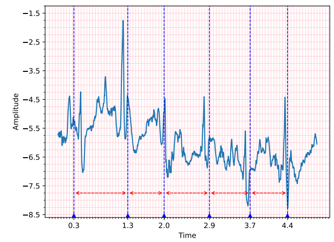
disturbance depends on the type and intensity of the noise. On the other hand, we have several heartbeat detection algorithms for identifying the R-peak in ECG that follow different procedures. The underlying idea is that these algorithms are usually accurate, and their results are similar to each other when the signal is not contaminated with noise. However, when the noise disturbs the signal, these algorithms are not able to detect the R-peaks properly, and their results are different from each other. The presence of a difference between the results of the heartbeat detection algorithms gives us a clue about whether the ECG signal is contaminated with noise.

Now, we discuss the procedure in detail. Our goal is to train an ML model that detects noisy segments. Therefore, to train and test such a model, we should prepare our feature sets. To this end, we first split the ECG recording into 20-second (non-overlapping) segments. We consider a binary classification problem, where the classes are noisy and non-noisy ECG segments. Then, we detect the R-peaks in each segment based on several heartbeat detection algorithms. Here, we focus on R-peaks, but other fiducial points in ECG may also be suitable candidates to be considered. In our work we employ seven popular algorithms, namely Hamilton [25], Christov [26], Pan and Tompkins [27], Stationary Wavelet Transform [28], Two Moving Average [29], Matched Filter [30], and WQRS [31] algorithms.

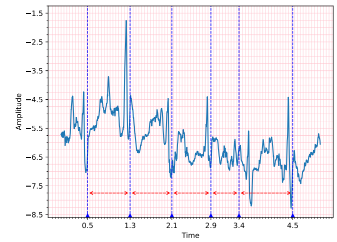
After detecting the R-peaks, we calculate RR intervals for all algorithms. Then, we sort the vector of RR intervals for each algorithm in descending order. The length of this vector is different for each segment and each algorithm. Therefore, we calculate the mean (μ) and standard deviation (σ) of the length of this vector based on the training set and consider a fixed size for the sorted RR interval vector for each segment and algorithm, which is $l = \lfloor \mu \rfloor + \lfloor \sigma \rfloor$. If the length of a vector is less than l, then we extend it with zero values.

Now, as we consider seven heartbeat detection algorithms in RecogNoise, we have seven sorted RR-interval vectors with length l for each segment. For building the feature set for each ECG segment, we merely concatenate these seven vectors. We also have a label, noisy or non-noisy, for each segment. At this point, we can train a binary classification model using an ML algorithm to predict if the ECG segment is noisy or not.

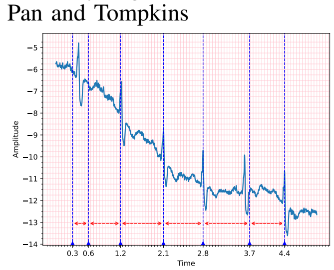
To provide a better understanding, we explain the types of noise that contaminate ECG recordings and provide a few examples to illustrate our technique. In [24], the authors categorize the noise for ECG recording into four classes based on their source. The four classes are presented as follows:



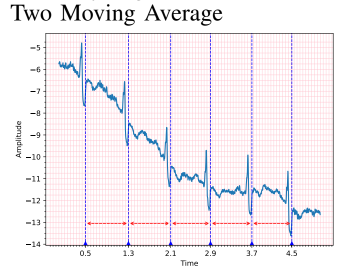
(a) Noisy Segment (EM, SNR 6), Pan and Tompkins



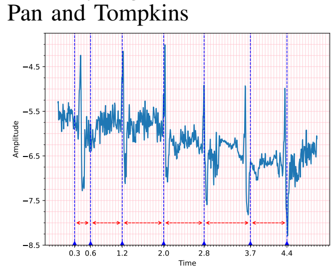
(b) Noisy Segment (EM, SNR 6), Two Moving Average



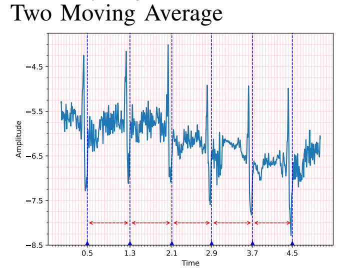
(c) Noisy Segment (BW, SNR 6), Pan and Tompkins



(d) Noisy Segment (BW, SNR 6), Two Moving Average



(e) Noisy Segment (MA, SNR 18), Pan and Tompkins



(f) Noisy Segment (MA, SNR 18), Two Moving Average

Fig. 2: RR Interval Computation Based on Pan and Tompkins and Two Moving Average Algorithms for Noisy Signals (5 Sec. Segments)

i) Electrode Motion (EM): This noise is the result of intermittent mechanical forces acting on the electrode. ii) Baseline Wander (BW): This noise is caused by the motion of the subject or electrode. iii) Muscle Artifact (MA): This noise originates from muscle activity, e.g., when muscles contract or move. iv) Power Line Interference (PLI): This noise occurs due to the electromagnetic interference of the alternating supply [32]. As the authors in [24] explain, PLI can be easily removed using a digital filter and can be ignored.

Here, we provide an example to explain our technique. Fig. 1 shows the clean signal (Subject 118, MLII channel, $05{:}00 - 05{:}05$ interval) and its ground truth R-peaks (blue dashed vertical lines) and RR intervals (red dashed horizontal arrows). Let us consider Pan and Tompkins [27] and Two Moving Average [29] heartbeat detection algorithms in our example. These algorithms obtain the same RR intervals for this clean recording. Fig. 2 shows noisy ECG segments and the R-peak detection results based on discussed algorithms and a publicly available Python implementation in [33]. All the signals are from the same recording and the same 5-second window of time in the recording.

Signals in Fig. 2 are contaminated with different types of noise, i.e., EM, BW, and MA, and different intensities (SNRs). The RR intervals in Figs. 2a, 2c, and 2e are calculated based on

Pan and Tompkins algorithms, while the RR intervals in Figs. 2b, 2d, and 2f are calculated based on Two Moving Average algorithm. When we compare the R-peaks and RR interval results in Figs. 2a and 2b, we see a difference in RR intervals results. This is the same when we compare the results in Figs. 2c and 2d or in Figs. 2e and 2f. This dissimilarity between the RR intervals detected based on different algorithms in noisy segments is a pattern that ML algorithms learn to identify noisy segments.

As discussed, RecogNoise can be extended to other fiducial points in ECG, and other intervals can be considered to detect noisy segments. Moreover, this technique can be employed to detect noise for other physiological signals, for instance, PPG signal.

III. Evaluation

In this section, we first discuss the experimental setup, including the dataset, classification performance metrics, and heartbeat detection algorithms. Then, we provide the experimental results based on different ML algorithms, different types of noise and different SNRs, and various numbers of heartbeat detection algorithms employed in RecogNoise.[1]

A. Experimental Setup

1) Dataset: We use MIT-BIH Arrhythmia Dataset [34], [23] for the evaluation of RecogNoise and performing the experiments. This dataset contains 48 half-hour two-channel recordings of ECG signals, obtained from 47 subjects. Similar to [35], we consider the recordings of this dataset as clean signals and add noise to them using WFDB software package [36] according to [24], [35]. By using the default settings of WFDB software and specifying the type and intensity of noise, starting after the initial 5 minutes of each recording, two-minute intervals of noise are introduced, alternating with two-minute clean segments.

2) Classification performance metrics: For the evaluation of trained ML models, we adopt several popular classification performance metrics, namely, F1-score, accuracy, precision, and recall.

$$F1\text{-}score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}, \quad (1)$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \quad (2)$$

$$Precision = \frac{TP}{TP + FP}, \quad (3)$$

$$Recall = \frac{TP}{TP + FN}, \quad (4)$$

where FP, TN, TP, and FN are the number of false-positives, true-negatives, true-positives, and false-negatives, respectively.

3) Heartbeat detection algorithms: We employed seven popular heartbeat detection algorithms for calculating the RR intervals in RecogNoise. The algorithms are Hamilton [25], Christov [26], Pan and Tompkins [27], Stationary Wavelet Transform [28], Two Moving Average [29], Matched Filter [30], and WQRS [31] algorithms. We used the publicly available Python implementation of these algorithms from [33].

[1] Our source code is available at https://github.com/AminAminifar/RecogNoise

TABLE I: Noise Detection Results for All Noise Types and SNRs Based on Different ML Algorithms

ML Alg.	Metric			
	F1 (%)	Acc. (%)	Perc. (%)	Rec. (%)
Lin. SVM	74.9	78.9	76.7	73.7
RBF SVM	85.4	87.4	84.9	86.1
DT	78.9	81.1	75.7	82.6
ERT	87.4	88.7	84.3	91.0
RF	86.9	88.3	83.6	90.7
GBDT	85.8	87.8	86.3	85.6
XGB	87.2	88.7	84.8	90.0

B. Experimental Results

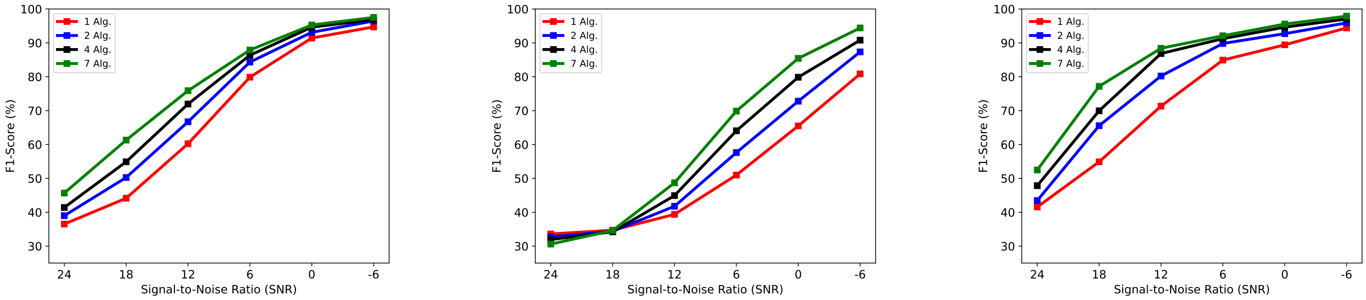
Now, we explain the procedures for the experimental results. We performed three types of experiments. In the first part, we change the ML algorithm in RecogNoise to see how this can affect the performance of our technique. In the second part, we change the type and intensity of noise to see how it can change the performance of our technique. In the third part, we employ various numbers and types of heartbeat detection algorithms in RecogNoise to see how they can influence the results.

In all three parts, recordings are split into 20-second (non-overlapping) segments, and each segment has a noisy or non-noisy label. We repeat each experiment in Sections III-B1, III-B2, and III-B3 for 100 rounds and report the averaged results. In each round, 30% of 48 recordings are selected at random for the test set and the rest for the train set. For the implementations, we use Python, Scikit-learn [37], NumPy [38], and pandas [39].

1) Evaluation based on different ML algorithms: In this experiment, we change the ML algorithm in RecogNoise to evaluate how this changes the prediction results and classification performance. To this end, we employ linear and Radial Basis Function (RBF) Support Vector Machine (SVM) [40], Decision Tree (DT) [41], Extremely Randomized Trees (ERT) [42], Random Forest (RF) [43], Gradient Boosted Decision Trees (GBDT) [44], and XGBoost (XGB) [45] algorithms to train our predictive model. We use the Scikit-learn [37] implementations with default parameters. In this experiment, the noisy signals are from all discussed types, i.e., EM, BW, and MA, and the SNR is one of the following values $\{24, 18, 12, 6, 0, -6\}$. The proportion of types and intensities are equal. Moreover, in this experiment, all seven heartbeat detection algorithms are employed in RecogNoise.

We report the results in Tabel I. We see that the results for ERT algorithm are slightly higher than others with respect to F1-score and accuracy. The results show that RecogNoise can detect noise with an F1-score of 87.4%, an accuracy of 88.7%, a precision of 84.3%, and a recall of 91% when employing ERT algorithm.

2) Evaluation based on different noise types and intensities: In this experiment, we inspect how changing the type and intensity of noise affect the prediction performance of RecogNoise. First, we have three sets of experiments for three types of noise that we change the intensity of noise in them. Then, we have three experiments in which we have various intensities of noise but only one type of noise. Finally, we



(a) Electrode Motion (EM) (b) Baseline Wander (BW) (c) Muscle Artifact (MA)

Fig. 3: Noise Detection Results for Three Types of Noise Considering Various Number of Heartbeat Detection Algorithms

TABLE II: Noise Detection Results for Different Noise Types and SNRs

Type of Noise	SNR	Metric			
		F1 (%)	Acc. (%)	Perc. (%)	Rec. (%)
EM	24	45.6	59.6	53.7	39.9
	18	61.3	68.1	64.1	59.2
	12	75.8	79.0	75.1	77.0
	6	87.8	89.4	86.6	89.3
	0	95.2	95.8	94.1	96.5
	-6	97.5	97.8	96.5	98.5
BW	24	30.6	52.8	41.2	24.5
	18	34.5	54.2	44.3	28.6
	12	48.6	60.5	54.8	44.1
	6	69.8	74.6	71.2	68.8
	0	85.4	87.6	86.3	84.6
	-6	94.4	95.1	93.2	95.6
MA	24	52.4	64.1	60.5	46.8
	18	77.1	80.8	78.7	76.0
	12	88.4	90.0	88.3	88.6
	6	92.1	93.2	92.3	91.9
	0	95.5	96.2	96.2	94.8
	-6	97.8	98.1	97.5	98.2
EM	all	89.5	90.7	87.1	92.2
BW	all	73.7	77.9	75.5	72.2
MA	all	89.1	90.3	86.5	91.9
all	all	86.9	88.3	83.6	90.7

have one experiment in which we have mixed noise types and mixed intensities. In all experiments, we employ all seven heartbeat detection algorithms and RF, which is a popular ML algorithm, for building our predictive model.

Table II shows the results of our experiments. In the first six experiments, the type of noise is EM but the SNR varies from 24 to -6. Similarly, in the next 12 experiments, the noise types, BW and MA, are the same, and only SNR is changed. In the next three experiments, the type of noise is the same, but the SNR is mixed (with an equal ratio) from 24 to -6. In the last experiment, the noise types (EM, BW, and MA) and SNR are mixed (with an equal ratio) from 24 to -6.

The results show that the type of noise and SNR influence the performance of RecogNoise. By considering the first 18 experiments, we see that MA is easier to detect and detecting BW is more challenging, and in general, detecting segments with low SNR is easier. The next three experiments show that RecogNoise can still detect noise patterns in segments when the SNR is variant. For EM, it detects noisy segments with an F1-score of 89.5% and an accuracy of 90.7%. For BW, it detects noise with an F1-score of 73.7% and an accuracy of 77.9%. For MA, it detects noise with an F1-score of 89.1% and an accuracy of 90.3%. The last experiments show that RecogNoise is still able to detect noise when the signal is contaminated with various types of noise and with different SNRs, i.e., it can detect noise with an F1-score of 86.9% and an accuracy of 88.3%.

3) Evaluation based on different numbers of heartbeat detection algorithms: In this part, we investigate how changing the number of heartbeat detection algorithms in RecogNoise can influence prediction performance. We design three sets of experiments in which the type of noise is the same, but we change SNR, from 24 to -6, and the number of heartbeat detection algorithms from 1 to 7. When employing 1, 2, or 4 heartbeat detection algorithms, the algorithms are selected at random, with similar probability, in each iteration. Here, we try to identify if increasing the complexity of our technique improves the prediction performance of RecogNoise. In all experiments, similar to the previous section, we employ RF for building our predictive model.

Fig. 3 shows the F1-score results for our experiments. In Fig. 3a, the signals are contaminated with EM noise, while in Fig. 3b and Fig. 3c, the signals are contaminated with BW and MA noise, respectively. We can see that by decreasing SNR, all the predictions improve but with different patterns. Moreover, increasing the number of heartbeat detection algorithms in the majority of experiments improves the prediction performance. As shown in Fig. 3a, when dealing with EM noise, employing more heartbeat detection algorithms is particularly helpful in higher SNRs. This is while for BW noise, as shown in Fig. 3b, employing more heartbeat detection algorithms is more helpful in low SNR values. For MA noise, as shown in Fig. 3c, employing more heartbeat detection algorithms is more helpful when SNR values are larger.

IV. Conclusion

Wearable devices are prone to collect noisy data due to their inherent limitations, and noise degrades the precision of these systems' decision-making procedures. In this research, we propose RecogNoise to detect noisy segments within ECG recordings, employing heartbeat detection algorithms and ML. We evaluate our technique using the MIT-BIH arrhythmia database, considering three primary types of noise, namely EM, BW, and MA, and various SNRs. Our results show that RecogNoise detects noisy segments with an F1-score of 86.9% and an accuracy of 88.3%.

References

[1] D. Sopic, A. Aminifar, A. Aminifar, and D. Atienza, "Real-time classification technique for early detection and prevention of myocardial infarction on wearable devices," in *2017 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2017, pp. 1–4.

[2] E. De Giovanni, A. A. Valdes, M. Peon-Quiros, A. Aminifar, and D. Atienza, "Real-time personalized atrial fibrillation prediction on multi-core wearable sensors," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 1654–1666, 2020.

[3] F. Forooghifar, A. Aminifar, T. Teijeiro, A. Aminifar, J. Jeppesen, S. Beniczky, and D. Atienza, "Self-aware anomaly-detection for epilepsy monitoring on low-power wearable electrocardiographic devices," in *2021 IEEE 3rd International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, 2021, pp. 1–4.

[4] A. Aminifar, F. Rabbi, V. K. I. Pun, and Y. Lamo, "Monitoring motor activity data for detecting patients' depression using data augmentation and privacy-preserving distributed learning," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2021, pp. 2163–2169.

[5] X. Erickson, S. Bastani, and A. Aminifar, "Personalized seizure detection using spiking neural networks," in *2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS)*. IEEE, 2023, pp. 1–6.

[6] A. Aminifar, M. Shokri, F. Rabbi, V. K. I. Pun, and Y. Lamo, "Extremely randomized trees with privacy preservation for distributed structured health data," vol. 10, pp. 6010–6027, 2022, conference Name: IEEE Access.

[7] M. Götzinger, N. Taherinejad, A. M. Rahmani, P. Liljeberg, A. Jantsch, and H. Tenhunen, "Enhancing the early warning score system using data confidence," in *Wireless Mobile Communication and Healthcare*. Springer International Publishing, 2017.

[8] F. Tabei, R. Kumar, T. N. Phan, D. D. McManus, and J. W. Chong, "A novel personalized motion and noise artifact (mna) detection method for smartphone photoplethysmograph (ppg) signals," *IEEE Access*, vol. 6, pp. 60 498–60 512, 2018.

[9] T. Athaya and S. Choi, "An efficient fingertip photoplethysmographic signal artifact detection method: A machine learning approach," *Journal of Sensors*, vol. 2021, pp. 1–18, 2021.

[10] M. Götzinger, A. Anzanpour, I. Azimi, N. TaheriNejad, A. Jantsch, A. M. Rahmani, and P. Liljeberg, "Confidence-enhanced early warning score based on fuzzy logic," vol. 27, no. 2, pp. 691–708, 2022.

[11] D. Pollreisz and N. TaheriNejad, "Detection and removal of motion artifacts in PPG signals," vol. 27, no. 2, pp. 728–738, 2022.

[12] D. Sopic, A. Aminifar, A. Aminifar, and D. Atienza, "Real-time event-driven classification technique for early detection and prevention of myocardial infarction on wearable systems," vol. 12, no. 5, pp. 982–992, 2018, publisher: IEEE.

[13] S. Baghersalimi, T. Teijeiro, D. Atienza, and A. Aminifar, "Personalized real-time federated learning for epileptic seizure detection," *IEEE journal of biomedical and health informatics*, vol. 26, no. 2, pp. 898–909, 2021.

[14] S. K. Bashar, D. Han, A. Soni, D. D. McManus, and K. H. Chon, "Developing a novel noise artifact detection algorithm for smartphone ppg signals: Preliminary results," in *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 2018, pp. 79–82.

[15] F. Mohagheghian, H. Dong, O. Ghetia, D. Chen, A. Peitzsch, N. Nishita, E. Y. Ding, E. M. Otabil, K. Noorishirazi, A. Hamel *et al.*, "Atrial fibrillation detection on reconstructed photoplethysmography signals collected from a smartwatch using a denoising autoencoder," *Expert Systems With Applications*, p. 121611, 2023.

[16] S. Sadiya, T. Alhanai, and M. M. Ghassemi, "Artifact detection and correction in eeg data: a review," in *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 2021, pp. 495–498.

[17] S. Mahmud, M. S. Hossain, M. E. Chowdhury, and M. B. I. Reaz, "Mlmrs-net: Electroencephalography (eeg) motion artifacts removal using a multi-layer multi-resolution spatially pooled 1d signal reconstruction network," *Neural Computing and Applications*, vol. 35, no. 11, pp. 8371–8388, 2023.

[18] M. Rohál'ová, P. Sykacek, M. Koskaand, and G. Dorffner, "Detection of the eeg artifacts by the means of the (extended) kalman filter," *Meas. Sci. Rev*, vol. 1, no. 1, pp. 59–62, 2001.

[19] Y. Kishimoto, Y. Kutsuna, and K. Oguri, "Detecting motion artifact ecg noise during sleeping by means of a tri-axis accelerometer," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2007, pp. 2669–2672.

[20] J. Ottenbacher, M. Kirst, L. Jatobá, U. Großmann, and W. Stork, "An approach to reliable motion artifact detection for mobile long-term ecg monitoring systems using dry electrodes," in *IV Latin American Congress on Biomedical Engineering*. Springer, 2008, pp. 440–443.

[21] J. Rodrigues, D. Belo, and H. Gamboa, "Noise detection on ecg based on agglomerative clustering of morphological features," *Computers in biology and medicine*, vol. 87, pp. 322–334, 2017.

[22] U. Satija, B. Ramkumar, and M. S. Manikandan, "Automated ecg noise detection and classification system for unsupervised healthcare monitoring," *IEEE Journal of biomedical and health informatics*, vol. 22, no. 3, pp. 722–732, 2017.

[23] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE engineering in medicine and biology magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[24] G. B. Moody, W. Muldrow, and R. G. Mark, "A noise stress test for arrhythmia detectors," *Computers in cardiology*, vol. 11, no. 3, pp. 381–384, 1984.

[25] P. Hamilton, "Open source ecg analysis," in *Computers in cardiology*. IEEE, 2002, pp. 101–104.

[26] I. I. Christov, "Real time electrocardiogram qrs detection using combined adaptive threshold," *Biomedical engineering online*, vol. 3, no. 1, pp. 1–9, 2004.

[27] J. Pan and W. J. Tompkins, "A real-time qrs detection algorithm," *IEEE transactions on biomedical engineering*, no. 3, pp. 230–236, 1985.

[28] 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 (BIBE)*. IEEE, 2017, pp. 457–461.

[29] M. Elgendi, M. Jonkman, and F. De Boer, "Frequency bands effects on qrs detection." *Biosignals*, vol. 2003, p. 2002, 2010.

[30] G. Turin, "An introduction to matched filters," *IRE Transactions on Information Theory*, vol. 6, no. 3, pp. 311–329, 1960.

[31] 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*. IEEE, 2003, pp. 737–740.

[32] S. Mian Qaisar, "Baseline wander and power-line interference elimination of ecg signals using efficient signal-piloted filtering," *Healthcare technology letters*, vol. 7, no. 4, pp. 114–118, 2020.

[33] B. Porr, L. Howell, I. Stournaras, and Y. Nir, "Popular ECG r peak detectors written in python," 2023. [Online]. Available: https://zenodo.org/record/7652725

[34] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[35] G. B. Moody, W. Muldrow, and R. G. Mark, "The MIT-BIH noise stress test database," 1992. [Online]. Available: https://physionet.org/content/nstdb/

[36] Waveform database. [Online]. Available: https://wfdb.io/

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[38] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020.

[39] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[40] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, 1995.

[41] J. Quinlan, "Induction of decision trees," *Machine learning*, 1986.

[42] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine learning*, 2006.

[43] T. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*. IEEE, 1995.

[44] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[45] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016.