

Detecting life-threatening patterns in Point-of-care ECG using efficient memory and processor power[★]

Francisco Bischoff^{1,2}[0000–0002–5301–8672]
Pedro Pereira Rodrigues^{1,2}[0000–0001–7867–6682]

¹ Department of Community Medicine, Information and Health Decision Sciences (MEDCIDS), Faculty of Medicine, University of Porto, Porto, Portugal

² Center for Health Technology and Services Research (CINTESIS), Faculty of Medicine, University of Porto, Porto, Portugal

Abstract. Currently, Point-of-Care (POC) ECG monitoring works either as plot devices or alarms for abnormal cardiac rhythms using predefined normal trigger ranges and some rhythm analysis, which raises the problem of false alarms. In comparison, complex 12-derivation ECG machines are not suitable to use as simple monitors and are used with strict techniques for formal diagnostics. Thinking outside the ICU setting, where high-end devices are available for patient monitoring, we aim to identify, on streaming data, life-threatening hearth electric patterns using low CPU and memory, enabling ward monitors, home devices and even wearable devices to be able to identify such events. The study design is comparable to a diagnostic study, where high accuracy is essential. Physionet's 2015 challenge yielded very good algorithms for reducing false alarms. However, none of the authors reported benchmarks, memory usage, robustness test, or context invariance that could assure its implementation on small devices. We expect to identify the obstacles of detecting life-threatening ECG changes within memory, space, and CPU constraints and using the proposed methods, assess the feasibility of implementing the algorithm in the real world and other settings than ICU monitors.

Keywords: anomaly detection · ECG · fading factors · matrix profile · time series · point-of-care

Papers should be submitted in PDF format through the [EPIA 2022 Easy-Chair submission page](#), and they should be prepared according to the [Springer LNCS](#) format. Submissions to the Doctoral Symposium have a maximum of 3 pages + references, and will be compiled in a separate informal PDF volume. Submissions must be authored by the PhD candidate, and should include information regarding the advisor(s) as a footnote in the first page, anchored in

[★] This work has been done under the scope of - and funded by - the Ph.D. Program in Health Data Science of the Faculty of Medicine of the University of Porto, Portugal - heads.med.up.pt

the author’s name. Submissions should also include a short abstract, keywords, introduction to the topic being addressed in the project and relevant research questions, a brief state of the art in the field, and the methodology to be pursued.

1 Introduction (introduction topic being addressed in the project)

Currently, Point-of-Care (POC) ECG monitoring works either as plot devices or alarms for abnormal cardiac rhythms using predefined normal trigger ranges. Modern devices also incorporate algorithms to analyze arrhythmias improving their specificity. On the other hand, full 12-derivation ECG machines are complex, are not suited to use as simple monitors, and are used with strict techniques for formal diagnostics of hearth electric conduction pathologies. The automatic diagnostics are derived from a complete analysis of the 12-dimension data after it is fully and well collected. Both systems do not handle disconnected leads and patient’s motions, being strictly necessary to have a good and stable signal to allow proper diagnosis. These interferences with the data collection frequently originate false alarms increasing both patient and staff’s stress; depending on how it is measured, the rate of false alarms (overall) in ICU is estimated at 65 to 95% [15].

Alarm fatigue is a well-known problem that consists of a sensory overload of nurses and clinicians, resulting in desensitization to alarms and missed alarms (the “crying wolf” situation). Patient deaths have been attributed to alarm fatigue [34]. In 1982, the increase in alarms with “no end in sight”; studies have demonstrated that most alarm signals have no clinical relevance and lead to clinical personnel’s delayed response. Ultimately patient deaths were reported related to inappropriate responses to alarms [34].

In April of 2013, The Joint Commission [4] issued the Sentinel Event Alert [23], establishing alarm system safety as a top hospital priority in the National Patient Safety Goal. Nowadays (2021), the subject is still on their list, in fourth place of importance [5].

In February of 2015, the CinC/Physionet Challenge 2015 was about “Reducing False Arrhythmia Alarms in the ICU [12]. The introduction article stated that it had been reported that up to 86% resulting of the alarms are false, and this can lead to decreased staff attention and an increase in patients’ delirium [11, 26, 30].

This subject draws attention to the importance of correctly identify abnormal hearth electric patterns in order to avoid the overload of clinical staff. Meanwhile, this opens the opportunity of thinking outside the ICU setting, where we still monitoring patients (and ourselves) using devices with low processing power, as for example ward monitors, home devices and wearable devices.

2 Objectives and the research question (relevant research questions)

While this research was inspired on the CinC/Physionet Challenge 2015, its purpose is not to beat the state of the art on that challenge, but to identify, on streaming data, abnormal hearth electric patterns, specifically those which are life-threatening, using low CPU and low memory requirements in order to be able to generalize the use of such information on lower-end devices, outside the ICU, as ward devices, home devices, and wearable devices.

The main questions is: can we accomplish this objective using a minimalist approach (low CPU, low memory) while maintaining robustness?

3 Related Works (a brief state of the art in the field)

Their algorithm did a pretty good job on the Physionet test set. However, independently of their approach to this problem, none of the authors reported benchmarks, memory usage, robustness test, or context invariance that could assure its implementation on real monitors to reduce alarm fatigue indeed.

There are other arrhythmias that this challenge did not assess, like atrial standstill (hyperkalemia), third-degree atrioventricular block, and others that may be life-threatening in some settings. Pulseless electrical activity is a frequent condition in cardiac arrest but cannot be identified without blood pressure information. This information is usually present in ICU settings but not in other locations.

4 The planned approach and methods for solving the problem (the methodology to be pursued.)

4.1 Research plan and methods

Matrix Profile Matrix Profile (MP) [37], is a state-of-the-art [14, 17] time series analysis technique that once computed, allows us to derive frameworks to all sorts of tasks, as motif discovery, anomaly detection, regime change detection and others [37].

Before MP, time series analysis relied on what is called *distance matrix* (DM), a matrix that stores all the distances between two time series (or itself, in case of a Self-Join). This was very power consuming, and several methods of pruning and dimensionality reduction were researched [27].

For brevity, let's just understand that the MP and the companion Profile Index (PI) are two vectors that hold one floating point value and one integer value, respectively, regarding the original time series: (1) the similarity distance between that point on time (let's call these points "indexes") and its first nearest-neighbor (1-NN), (2) The index where this this 1-NN is located. The original paper has more detailed information [37]. It is computed using a rolling window but instead of creating a whole DM, only the minimum values and the index of

these minimum are stored (in the MP and PI respectively). We can have an idea of the relationship of both on Fig. 1.

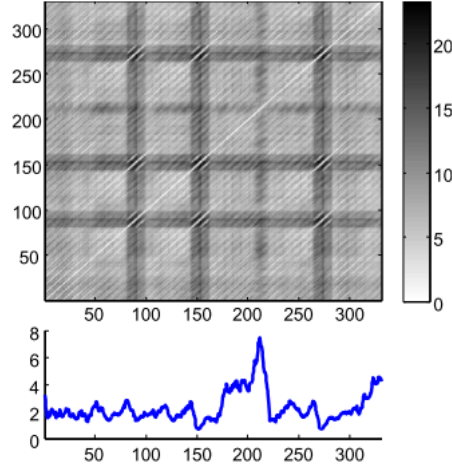


Fig. 1. A distance matrix (top), and a matrix profile (bottom). The matrix profile stores only the minimum values of the distance matrix.

This research has already yielded two R packages concerning the MP algorithms from UCR [6]. The first package is called `tsmp`, and a paper has also been published in the R Journal [10] (Journal Impact FactorTM, 2020 of 3.984). The second package is called `matrixprofiler` and enhances the first one, using low-level language to improve computational speed. The author has also joined the Matrix Profile Foundation as co-founder together with contributors from Python and Go languages [1, 35].

This implementation in R is being used for computing the MP and MP-based algorithms of this thesis.

Type of study This thesis will be a diagnostic study as the algorithm must classify the change in pattern as positive or negative for life-threatening.

The data The current dataset used is the CinC/Physionet Challenge 2015 public dataset, modified to include only the actual data and the header files in order to be read by the pipeline and is hosted by Zenodo [3] under the same license as Physionet.

The dataset is composed of 750 patients with at least five minutes records. All signals have been resampled (using anti-alias filters) to 12 bit, 250 Hz and have had FIR band-pass (0.05 to 40Hz) and mains notch filters applied to remove noise. Pacemaker and other artifacts are still present on the ECG [12].

Furthermore, this dataset contains at least two ECG derivations and one or more variables like arterial blood pressure, photoplethysmograph readings, and respiration movements.

The *events* we seek to identify are the life-threatening arrhythmias as defined by Physionet in Table 1.

Table 1. Definition of the five alarm types used in CinC/Physionet Challenge 2015.

Alarm	Definition
Asystole	No QRS for at least 4 seconds
Extreme Bradycardia	Heart rate lower than 40 bpm for 5 consecutive beats
Extreme Tachycardia	Heart rate higher than 140 bpm for 17 consecutive beats
Ventricular Tachycardia	5 or more ventricular beats with heart rate higher than 100 bpm
Ventricular Flutter/Fibrillation	Fibrillatory, flutter, or oscillatory waveform for at least 4 seconds

The fifth minute is precisely where the alarm has been triggered on the original recording set. To meet the ANSI/AAMI EC13 Cardiac Monitor Standards [9], the onset of the event is within 10 seconds of the alarm (i.e., between 4:50 and 5:00 of the record). That doesn’t mean that there are no other arrhythmias before.

For comparison, on Table 2 we collected the score of the five best participants of the challenge [13, 16, 21, 24, 31].

Table 2. Challenge Results on real-time data. The scores were multiplied by 100.

Score	Authors
81.39	Filip Plesinger, Petr Klimes, Josef Halamek, Pavel Jurak
79.44	Vignesh Kalidas
79.02	Paula Couto, Ruben Ramalho, Rui Rodrigues
76.11	Sibylle Fallet, Sasan Yazdani, Jean-Marc Vesin
75.55	Christoph Hoog Antink, Steffen Leonhardt

The equation used on this challenge to compute the score of the algorithms is in the Equation (1). This equation is the accuracy formula, with penalization of the false negatives. The reasoning pointed out by the authors [12] is the

clinical impact of existing a genuine life-threatening event that was considered unimportant. Accuracy is known to be misleading when there is a high class imbalance [7].

$$Score = \frac{TP + TN}{TP + TN + FP + 5 * FN} \quad (1)$$

Assuming that this is a finite dataset, the pathologic cases (1) $\lim_{TP \rightarrow \infty}$ (whenever there is an event, it is positive) or (2) $\lim_{TN \rightarrow \infty}$ (whenever there is an event, it is false), cannot happen. This dataset has 292 True alarms and 458 False alarms. Experimentally, this equation yields:

- 0.24 if all guesses are on False class
- 0.28 if random guesses
- 0.39 if all guesses are on True class
- 0.45 if no false positives plus random on True class
- 0.69 if no false negatives plus random on False class

This small experiment (knowing the data in advance) shows that “a single line of code and a few minutes of effort” [36] algorithm could achieve at most a score of 0.39 in this challenge (the last two lines, the algorithm must to be very good on one class).

Nevertheless, this equation will only be useful to allow us to compare the results of this thesis with other algorithms.

Proposed approach The proposed approach is depicted in Fig. 2. That is only a draft of the final workflow. The algorithm for the classification of the regime changes is still to be defined. However, the main innovation resides in the correct regime detection. Also, to achieve the goal of low CPU and memory usage, the strategy will be to combine fading factors[18, 33] to reduce computation in online settings like in this research.

Detecting regime changes The regime change approach will be using the *Arc Counts* concept, used on the FLUSS (Fast Low-cost Unipotent Semantic Segmentation) algorithm, as explained by Gharghabi, *et al.*, [19].

The FLUSS (and FLOSS, the on-line version) algorithm is built on top of the Matrix Profile (MP)[37], described on section 4.1. Recalling that the MP and the companion Profile Index (PI) are two vectors holding information about the 1-NN. One can imagine several “arcs” starting from one “index” to another. This algorithm is based on the assumption that between two regimes, the most similar shape (its nearest neighbor) is located on “the same side”, so the number of “arcs” decreases when there is a change on the regime, and increases again.

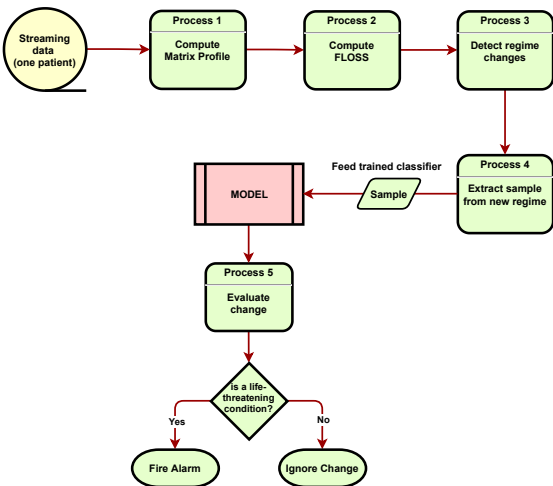


Fig. 2. Pipeline of the final process. The streaming data, coming from one patient, is processed to create its Matrix Profile. Then, the FLOSS algorithm is computed for detecting a regime change. When a new regime is detected, a sample of this new regime is analysed by the model and a decision is made. If the new regime is life-threatening, the alarm will be fired.

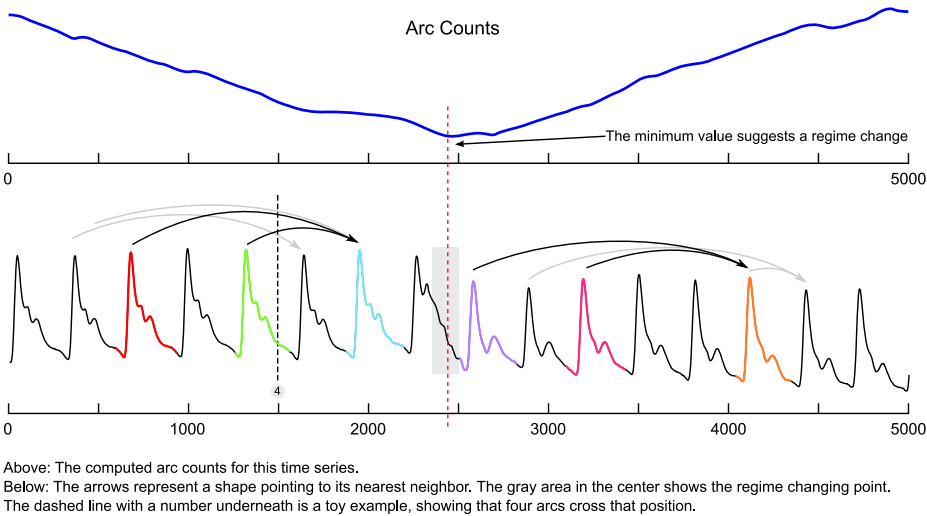


Fig. 3. FLUSS algorithm, using arc counts.

As show on Fig. 3. This drop on the *Arc Counts* is a signal that a change on the shape of the signal has happened.

The choice of the FLOSS algorithm (on-line version of FLUSS) is founded on the following arguments:

- **Domain Agnosticism:** the algorithm makes no assumptions about the data as opposed to most available algorithms to date.
- **Streaming:** the algorithm can provide real-time information.
- **Real-World Data Suitability:** the objective is not to *explain* all the data. Therefore, areas marked as “don’t know” areas are acceptable.
- **FLOSS is not:** a change point detection algorithm [8]. The interest here is changes in the shapes of a sequence of measurements.

Other algorithms we can cite are based on Hidden Markov Models (HMM) that require at least two parameters to be set by domain experts: cardinality and dimensionality reduction. The most attractive alternative could be the Autoplait [28], which is also domain agnostic and parameter-free. It segments the time series using Minimum Description Length (MDL) and recursively tests if the region is best modeled by one or two HMM. However, Autoplait is designed for batch operation, not streaming, and also requires discrete data. FLOSS was demonstrated to be superior in several datasets in its original paper. In addition, FLOSS is robust to several changes in data like downsampling, bit depth reduction, baseline wandering, noise, smoothing, and even deleting 3% of the data and filling with simple interpolation. Finally, the most important, the algorithm is light and suitable for low-power devices.

In the MP domain, it is worth also mentioning other possible algorithm: the Time Series Snippets [22], based on MPdist [20]. The latter measures the distance between two sequences considering how many similar sub-sequences they share, no matter the order of matching. It proved to be a useful measure (not a metric) for meaningfully clustering similar sequences. Time Series Snippets exploits MPdist properties to summarize a dataset extracting the k sequences that represent most of the data. The final result seems to be an alternative for detecting regime changes, but it is not. The purpose of this algorithm is to find which pattern(s) explains most of the dataset. Also, it is not suitable for streaming data. Lastly, MPdist is quite expensive compared to the trivial Euclidean distance.

The regime change detection will be evaluated following the criterias explained on section ??.

Classification of the new regime The next step towards the objective of this work is to verify if the new regime detected by the previous step is indeed a life-threatening pattern that we should trigger the alarm.

First let’s dismiss some apparent solutions: (1) Clustering. It is well understood that we cannot cluster time series subsequences meaningfully with any distance measure, or with any algorithm [25]. The main argument is that in a meaningful algorithm, the output depends on the input, and this has been

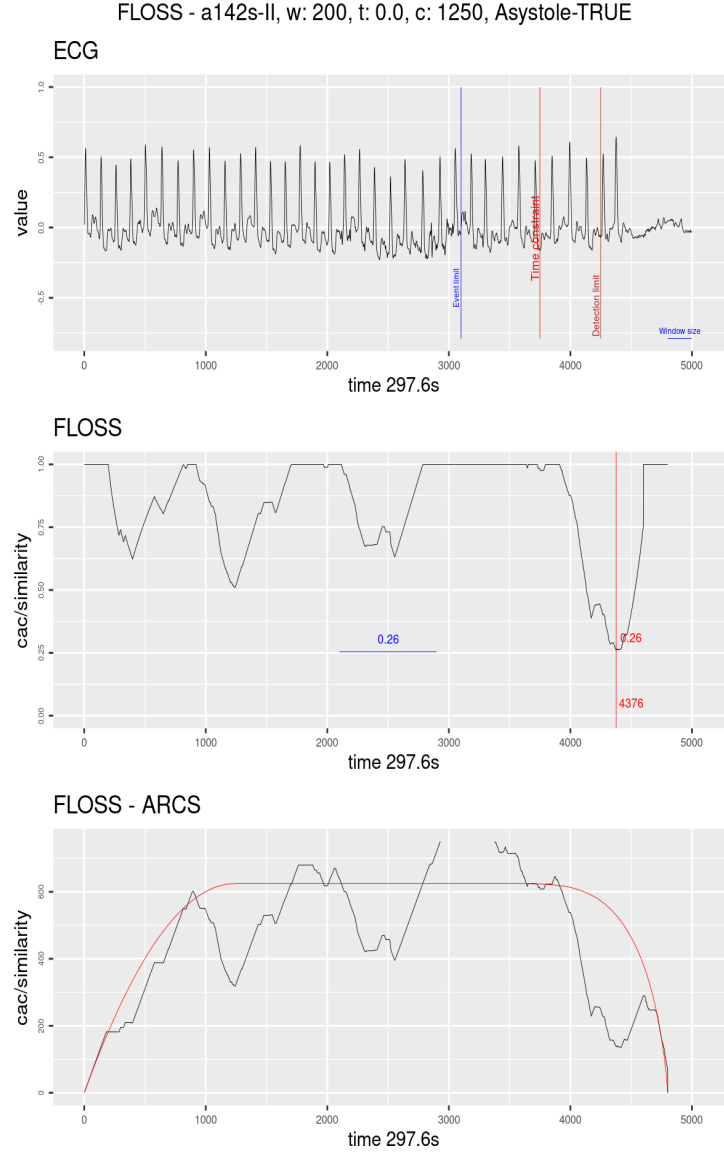


Fig. 4. Regime change detection example. The graph on top shows the ECG streaming; the blue line marks the ten seconds before the original alarm was fired; the red line marks the time constraint of 1250; the dark red line marks the limit for taking a decision in this case of Asystole the blue horizontal line represents the size of the sliding window. The graph on the middle shows the Arc counts as seen by the algorithm (with the corrected distribution); the red line marks the current minimum value and its index; the blue horizontal line shows the minimum value seen until then. The graph on the bottom shows the computed Arc counts (raw) and the red line is the theoretical distribution used for correction.

proven to not happen in time series subsequence clustering [25]. (2) Anomaly detection. In this work we are not looking for surprises, but for patterns that are known to be life-threatening. (3) Forecasting. We may be tempted to make predictions, but clearly this is not the idea here.

The method of choice is classification. The simplest algorithm could be a TRUE/FALSE binary classification. Nevertheless, the five life-threatening patterns have well defined characteristics that may seem more plausible to classify the new regime using some kind of ensemble of binary classifiers or a “six-class” classifier (being the sixth class the FALSE class).

Since the model doesn’t know which life-threatening pattern will be present in the regime (or if it will be a FALSE case), the model will need to check for all five TRUE cases and if none of these cases are identified, it will classify the regime as FALSE.

In order to avoid exceeding processor capacity, an initial set of shapelets [32] can be sufficient to build the TRUE/FALSE classifier. And to build such set of shapelets, leveraging on the MP, we will use the Contrast Profile [29].

The Contrast Profile (CP) looks for patterns that are at the same time very *similar* to its neighbors in class *A* while is very *different* from the nearest neighbor from class *B*. In other words, this means that such pattern represents well class *A* and may be taken as a “signature” of that class.

In this case we need to compute two MP, one self-join MP using the *positive* class $MP^{(++)}$ (the class that has the signature we want to find) and one AB-join MP using the *positive* and *negative* classes $MP^{(+-)}$. Then we subtract the first $MP^{(++)}$ from the last $MP^{(+-)}$, resulting in the *CP*. The high values on *CP* are the locations for the signature candidates we look for (the author of CP calls these segments *Plato’s*).

Due to the nature of this approach, the MP’s (containing values in Euclidean Distance) are truncated for values above $\sqrt{2w}$, where w is the window size. This because values above this threshold are negatively correlated in the Pearson Correlation space. Finally, we normalize the values by $\sqrt{2w}$. The formula (2) synthesizes this computation.

$$CP_w = \frac{MP_w^{(+-)} - MP_w^{(++)}}{\sqrt{2w}} \quad \text{where } w \text{ is the window size} \quad (2)$$

For a more complete understanding of the process, Fig. 5 shows a practical example from the original article [29].

After extracting candidates for each class signature, a classification algorithm will be fitted and evaluated using the criterias explained on section ??.

An example of candidates for ventricular tachycardia is presented on Fig. 6.

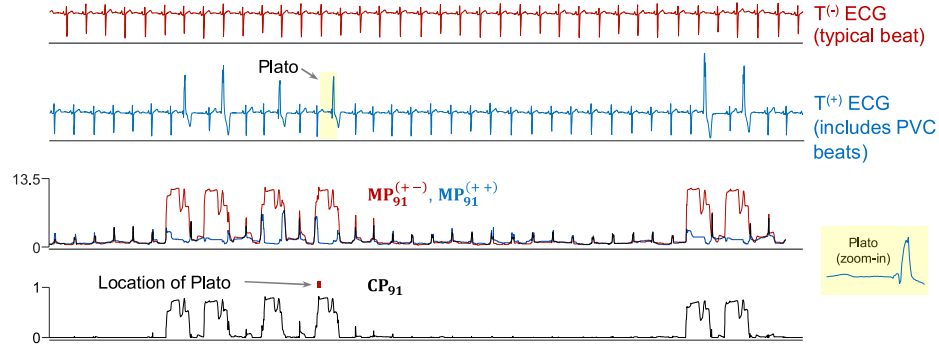


Fig. 5. Top to bottom: two weakly-labeled snippets of a larger time series. $T(-)$ contains only normal beats. $T(+)$ also contains PVC (premature ventricular contractions). Next, two Matrix Profiles with window size 91; AB-join is in red and self-join in blue. Bottom, the Contrast Profile showing the highest location.

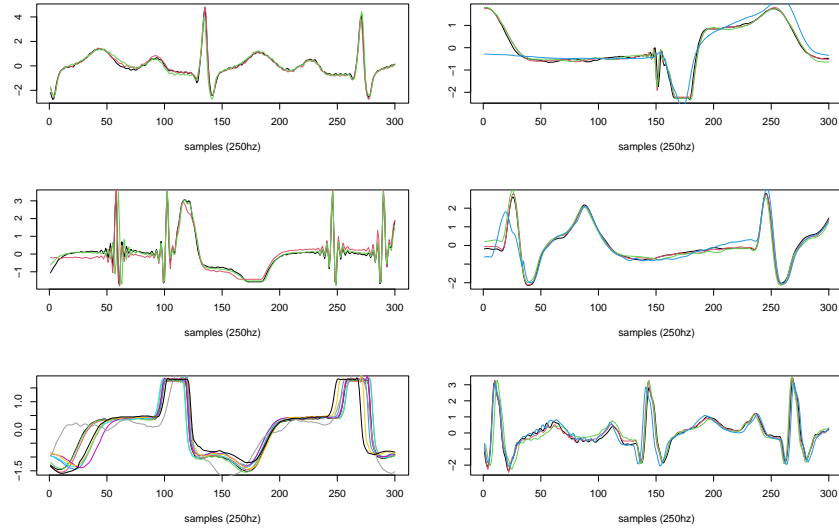


Fig. 6. Shapelet candidates for Ventricular Tachycardia.

4.2 Feasibility trial

A side-project called “false.alarm.io” has been derived from this work (an unfortunate mix of “false.alarm” and “PlatformIO” [2], the IDE chosen to interface the panoply of embedded systems we can experiment with). The current results of this side-project are very enlightening and show that the final algorithm can indeed be used in small hardware. Further data will be available in the future.

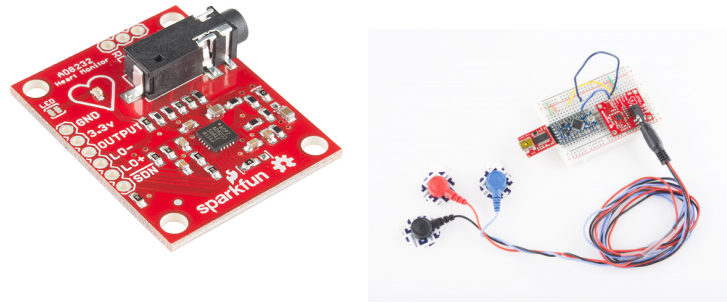


Fig. 7. Single Lead Heart Rate Monitor

A brief mentioning, linking back to the objectives of this work, an initial trial was done using an ESP32 MCU (Fig. 8) in order to be sure if such small device can handle the task.

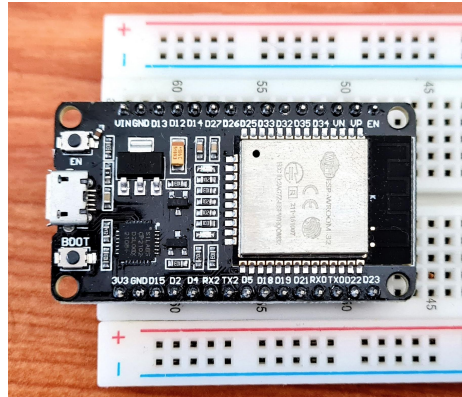


Fig. 8. ESP32 MCU

Research Team

- Thesis Author: Francisco Bischoff
- Supervisor: Professor Pedro Pereira Rodrigues
- Co-supervisor: Professor Eamonn Keogh (UCR, Riverside)

4.3 Expected results and outcomes

At the end, this thesis will provide a framework for identify life-threatening conditions using biological streaming data on devices with low CPU and low memory specifications. We expect to achieve a high quality model on identifying these pathological conditions, maintaining its robustness in presence of noise and artifacts seen on real-world applications.

References

1. Matrix Profile Foundation, <https://matrixprofile.org/>, last accessed 2022/01/27.
2. PlatformIO, a professional collaborative platform for embedded development, <https://platformio.org/>, last accessed 2022/01/05.
3. Reducing False Arrhythmia Alarms in the ICU - The PhysioNet Computing in Cardiology Challenge 2015, <https://zenodo.org/record/4634013>, (2021). <https://doi.org/10.5281/zenodo.4634013>.
4. The joint commission, <https://www.jointcommission.org>, last accessed 2021/04/08.
5. The joint commission - national patient safety goals, <https://www.jointcommission.org/standards/national-patient-safety-goals/hospital-national-patient-safety-goals/>, last accessed 2021/04/08.
6. UCR Matrix Profile Page, <https://www.cs.ucr.edu/~eamonn/MatrixProfile.html>, last accessed 2022/01/27.
7. Akosa, J.S.: Predictive accuracy: A misleading performance measure for highly imbalanced data. SAS Global Forum. 942, 1–12 (2017).
8. Aminikhanghahi, S., Cook, D.J.: A survey of methods for time series change point detection. Knowledge and Information Systems. 51, 2, 339–367 (2016). <https://doi.org/10.1007/s10115-016-0987-z>.
9. Association for the Advancement of Medical Instrumentation: Cardiac monitors, heart rate meters, and alarms. Association for the Advancement of Medical Instrumentation, Arlington, Va. (2002).
10. Bischoff, F., Rodrigues, P.P.: tsmpp: An R Package for Time Series with Matrix Profile. The R Journal. 12, 1, 76–86 (2020). <https://doi.org/10.32614/RJ-2020-021>.

14. Bischoff F., Rodrigues PP.
11. Chambrin, M.C.: Alarms in the intensive care unit: How can the number of false alarms be reduced? *Critical care* (London, England). 5, 4, 184–8 (2001). <https://doi.org/10.1186/cc1021>.
12. Clifford, G.D. et al.: The PhysioNet/computing in cardiology challenge 2015: Reducing false arrhythmia alarms in the ICU. In: *Computing in cardiology*. (2015). <https://doi.org/10.1109/cic.2015.7408639>.
13. Couto, P. et al.: Suppression of false arrhythmia alarms using ECG and pulsatile waveforms. Presented at the September (2015). <https://doi.org/10.1109/cic.2015.7411019>.
14. De Paepe, D. et al.: A generalized matrix profile framework with support for contextual series analysis. *Engineering Applications of Artificial Intelligence*. 90, January, 103487 (2020). <https://doi.org/10.1016/j.engappai.2020.103487>.
15. Donchin, Y., Seagull, F.J.: The hostile environment of the intensive care unit. *Current Opinion in Critical Care*. 8, 4, 316–320 (2002). <https://doi.org/10.1097/00075198-200208000-00008>.
16. Fallet, S. et al.: A multimodal approach to reduce false arrhythmia alarms in the intensive care unit. Presented at the September (2015). <https://doi.org/10.1109/cic.2015.7408640>.
17. Feremans, L. et al.: Pattern-Based Anomaly Detection in Mixed-Type Time Series. In: *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)*. pp. 240–256 (2020). https://doi.org/10.1007/978-3-030-46150-8_15.
18. Gama, J. et al.: On evaluating stream learning algorithms. *Machine Learning*. 90, 3, 317–346 (2013). <https://doi.org/10.1007/s10994-012-5320-9>.
19. Gharghabi, S. et al.: Domain agnostic online semantic segmentation for multi-dimensional time series. *Data Mining and Knowledge Discovery*. 33, 1, 96–130 (2018). <https://doi.org/10.1007/s10618-018-0589-3>.
20. Gharghabi, S. et al.: Matrix profile XII: MPdist: A novel time series distance measure to allow data mining in more challenging scenarios. Presented at the November (2018). <https://doi.org/10.1109/ICDM.2018.00119>.
21. Hoog Antink, C., Leonhardt, S.: Reducing false arrhythmia alarms using robust interval estimation and machine learning. Presented at the September (2015). <https://doi.org/10.1109/cic.2015.7408642>.
22. Imani, S. et al.: Matrix profile XIII : Time series snippets : A new primitive for time series data mining. In: *2018 IEEE international conference on data mining (ICDM)*. (2018).
23. Joint Commission: Sentinel event alert - Medical device alarm safety in hospitals. 50, 1–3 (2013). <https://www.ncbi.nlm.nih.gov/pubmed/23767076>.

24. Kalidas, V., Tamil, L.S.: Enhancing accuracy of arrhythmia classification by combining logical and machine learning techniques. Presented at the September (2015). <https://doi.org/10.1109/cic.2015.7411015>.
25. Keogh, E., Lin, J.: Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems*. 8, 2, 154–177 (2005). <https://doi.org/10.1007/s10115-004-0172-7>. <https://www.ncbi.nlm.nih.gov/pubmed/1622>.
26. Lawless, S.T.: Crying wolf: False alarms in a pediatric intensive care unit. *Critical care medicine*. 22, 6, 981–5 (1994). <https://www.ncbi.nlm.nih.gov/pubmed/8205831>.
27. Lin, J. et al.: Experiencing SAX: A novel symbolic representation of time series. *Data Mining and Knowledge Discovery*. 15, 2, 107–144 (2007). <https://doi.org/10.1007/s10618-007-0064-z>.
28. Matsubara, Y. et al.: AutoPlait: Automatic mining of co-evolving time sequences. *Proceedings of the ACM SIGMOD International Conference on Management of Data*. 193–204 (2014). <https://doi.org/10.1145/2588555.2588556>.
29. Mercer, R. et al.: Matrix profile XXIII: Contrast profile: A novel time series primitive that allows real world classification. In: *ICDM 2021*. p. 10 IEEE, Auckland (2021).
30. Parthasarathy, S., Tobin, M.J.: Sleep in the intensive care unit. *Intensive Care Medicine*. 30, 2, 197–206 (2004). <https://doi.org/10.1007/s00134-003-2030-6>.
31. Plesinger, F. et al.: False alarms in intensive care unit monitors: Detection of life-threatening arrhythmias using elementary algebra, descriptive statistics and fuzzy logic. Presented at the September (2015). <https://doi.org/10.1109/cic.2015.7408641>.
32. Rakthanmanon, T., Keogh, E.: Fast shapelets: A scalable algorithm for discovering time series shapelets. In: *Proceedings of the 2013 SIAM international conference on data mining*. pp. 668–676 Society for Industrial; Applied Mathematics, Philadelphia, PA (2013). <https://doi.org/10.1137/1.9781611972832.74>.
33. Rodrigues, P.P. et al.: Memoryless Fading Windows in Ubiquitous Settings. In: *Proceedings of the first ubiquitous data mining workshop*. pp. 23–27 (2010).
34. Sendelbach, S., Funk, M.: Alarm Fatigue. *AACN Advanced Critical Care*. 24, 4, 378–386 (2013). <https://doi.org/10.4037/nci.0b013e3182a903f9>.
35. Van Benschoten, A. et al.: MPA: A novel cross-language API for time series analysis. *Journal of Open Source Software*. 5, 49, 2179 (2020). <https://doi.org/10.21105/joss.02179>.

16 Bischoff F., Rodrigues PP.

- 36. Wu, R., Keogh, E.: Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress. *IEEE Transactions on Knowledge and Data Engineering*. (2021). <https://doi.org/10.1109/TKDE.2021.3112126>.
- 37. Yeh, C.-C.M. et al.: Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In: 2016 IEEE 16th international conference on data mining (ICDM). pp. 1317–1322 IEEE (2016). <https://doi.org/10.1109/ICDM.2016.0179>.