ECG-Based Biometric System using TinyML: Implementation and Performance Evaluation on ESP32

HATEM ZEHIR1,*, TOUFIK HAFS1, SARA DAAS1

1. LERICA Laboratory, Faculty of Technology, Badji Mokhtar-Annaba University, P.O. BOX 12, Annaba, 23000, Algeria

zehir.hatem@univ-annaba.org, hafstoufik@gmail.com, sara.daas@yahoo.com

ABSTRACT

This paper presents a novel biometric system based on electrocardiogram (ECG) signals and tiny machine learning (TinyML) techniques for personal identification. A subset of 10 individuals from the MIT-BIH Arrhythmia Database was used to develop and evaluate the proposed system. In the preprocessing stage, the ECG signals were denoised using a 4th-order Butterworth bandpass filter. Subsequently, the signals were segmented into 100ms windows centred around each r-peak, which were detected using the Pan-Tompkins++ algorithm. To ensure uniformity, the amplitudes of the segmented windows were normalized within the range of 0 to 1.

For the classification, a deep learning model consisting of an input layer and three fully connected layers was employed. To enable deployment on resource-constrained devices, the trained model was converted to TensorFlow Lite format and further transformed into an Arduino library.

The implemented system was successfully deployed on an ESP32 microcontroller, thereby demonstrating the feasibility of real-time biometric authentication using TinyML. The identification accuracy achieved was 96.71%.

KEY WORDS

Biometrics; Electrocardiogram; Deep Learning; Tiny Machine Learning; TensorFlow Lite; ESP32

I. INTRODUCTION

The use of electrocardiogram (ECG) signals has emerged in recent years as a secure and reliable biometric modality. ECG signals are proven to be unique and vary from individual to individual, making them an interesting subject of study in the biometrics field. This growing interest is due to the fact that ECG signals are robust to different attacks, such as spoofing and forgery. Thanks to technological advancements, they can also be easily acquired at a low cost. ECG signals can now be acquired using a simple smartwatch or wearable wristband.

ECG signals refer to the graphical representation of the electrical signals caused by the depolarization of the heart. The electrocardiogram is composed of three prominent waves: the P wave, the QRS complex, and the T wave. The QRS complex is the most visible wave because of its large amplitude and duration of 80 to 100 milliseconds. These waves are shown in Fig. 1.

Several studies have been conducted on ECG biometrics using deep learning. Belo et al. [1] proposed two neural network architectures, one based on a Temporal Convolutional Neural Network (TCN) and the other on a Recurrent Neural Network (RNN) for both identification and authentication. The authors tested their approach on three public databases: Fantasia [2], MIT-BIH Arrhythmia Database [3], and CYBHi [4]. The experiments showed that the TCN model outperformed the RNN model by achieving a maximum test accuracy of 100% on the Fantasia database, and a lowest equal error rate (EER) of 0.0% on the same database.

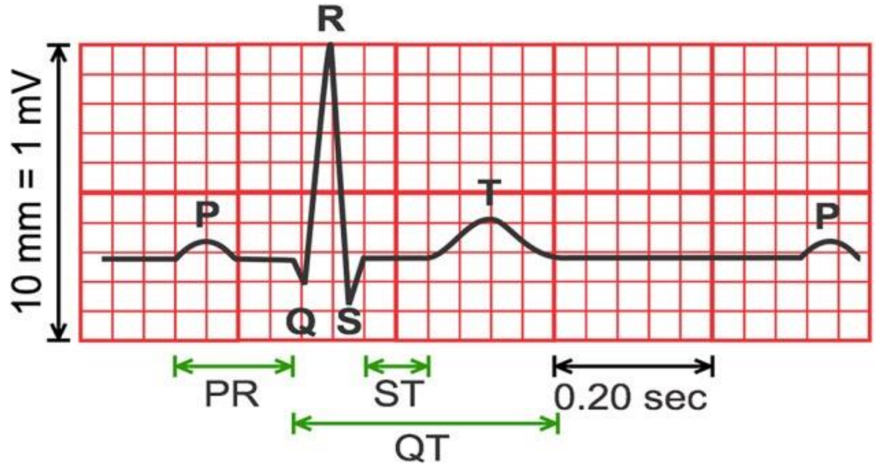

Fig. 1. A schematic representation showing the most important points of an ECG signal [5]

In another paper, Li et al. **[6]** proposed an approach based on a cascaded convolutional neural network (CNN). Two CNN models were trained: the first for feature extraction and the second for biometric identification. To preprocess the signals from any noise, a 6-order Butterworth bandpass filter with cutoff frequencies of 2 Hz and 50 Hz was used. The amplitudes were then normalized between 0 and 1, and the signals were segmented into heartbeats around each R-peak. The model was tested on five different databases, and an average identification accuracy of 94.3% was achieved.

In a third paper, Zehir et al. **[7]** proposed a GRU-based deep learning model to classify the instantaneous frequencies of the first two intrinsic mode functions IMFs extracted using the Hilbert–Huang transform (HHT) **[8]** from ECG signals decomposed using the empirical mode decomposition (EMD) **[9]**. Before extracting those features the signals were denoised using a 4th-order Butterworth bandpass filter with cutoff frequencies of 5 Hz and 15 Hz. The authors were able to achieve identification accuracies of 95.31% and 96.42% on the MIT-BIH and PTB databases respectively.

By reviewing the existing works, we notice the lack of using Tiny Machine Learning (TinyML) [10]–[12] and the deployment of deep learning-based ECG biometric systems on low-constraint devices, despite the popularity of deep learning on ECG biometrics. TinyML is an emerging subfield of machine learning that creates and deploys machine and deep learning models on low-power devices such as microcontrollers. TinyML models are much simpler and smaller in size than the original models, which allows them to be executed on embedded systems and wearable devices.

In this work, we aim to fill this gap by exploiting the recent advances in TinyML techniques to deploy and evaluate the performance of an ECG biometric system on the ESP32 board. TinyML is an emerging field that focuses on deploying machine learning models on embedded systems with low-computational power and limited memory resources.

The rest of the paper is organized as follows: Section 2 provides an overview of our proposed method for ECG biometrics. Section 3 presents and discusses the results. Section 4 summarizes the approach and the prospects of this paper.

II. PROPOSED METHOD

This study as described by Fig. 2 utilized a subset of 10 individuals from the MIT-BIH Arrhythmia Database. In the preprocessing stage, signals were denoised, segmented into 100ms windows around r-peaks, and their amplitudes were normalized. The deep learning model consisted of an input layer and three fully connected layers. The model was converted to TFLite format, transformed into an Arduino library, and deployed on an ESP32.

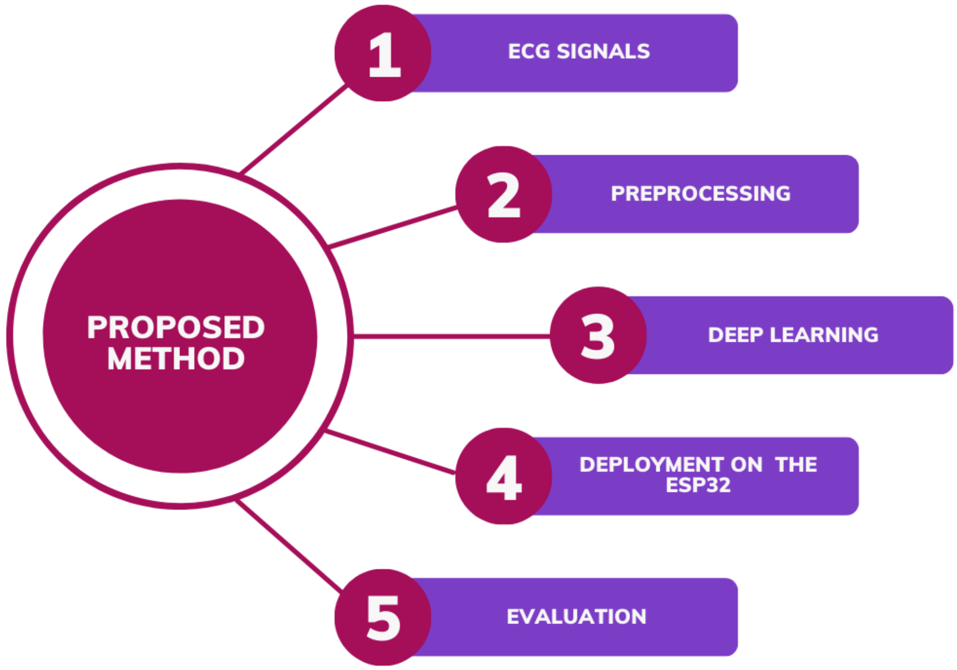

Fig. 2. A diagram showing the different stages of our proposed system

A. ECG Data

To evaluate our system we have used the MIT-BIH Arrhythmia Database [3], [13]. It consists of 48 different recordings collected at a sampling frequency of 360 Hz from 47 individuals. The data was gathered from a diverse group of subjects receiving medical care, including those who are inpatients at the Beth Israel Hospital in Boston and those who are attending as outpatients. Due to the memory constraints of the ESP32, we have chosen to only evaluate our model using the first 10 subjects of the database, and we worked with single lead data which has been validated as satisfactory for ECG biometric identification [14].

B. Filtering

When recording, ECG signals get affected by many types of noises that will reduce the relevant information carried by the signals. Those noises can be low or high frequencies and may include: baseline wander, which is a noise which falls within the range of [0.15 – 0.30] Hz and can be caused by breathing or muscles movement; power line interference, which has a frequency of 50 or 60 Hz; and instrumentation noise caused by the electrode and devices used to acquire the signals.

To eliminate the noise, we have opted to use a 4th-order Butterworth bandpass filter with cutoff frequencies of 1 Hz and 40 Hz before further preprocessing the signals.

C. Segmentation

In ECG, The peak with the highest prominence is known as the R peak as seen in **Fig. *1***. In our study, the R-peaks are chosen as fiducial marks to segment the signals. First, the Pan-Tomkins++ algorithm [15] is applied to detect the R-peaks. After that, the signals were segmented around each detected peak; taking into consideration that the QRS complex has a maximum duration of 100 ms and the sampling frequency of the signals, the segmentation window was 18 samples before and 18 samples after each R peak. Making sure that each segment represents a full QRS complex. Segmenting the signal into fixed-size windows around each R-peak will reduce the variability, which will eventually lead to the improvement of the performance of the proposed biometric system. After the segmentation, we obtained 26006 QRS complexes, those segments will be used to train the deep learning model.

D. Normalization

Once the signals are segmented, the amplitude of each segment was normalized in the range of [0 - 1]. The normalization process ensures that all the extracted QRS complexes are comparable between all the individuals, which has a positive impact on the identification accuracy. The segments were normalized according to this equation:

$$x_{rescaled} = \frac{x - min_x}{max_x - min_x} \tag{1}$$

E. Architecture of the Proposed Deep Learning Model

The model is implemented using Python, TensorFlow, and Keras. It is structured as shown by Fig. 3 with three dense layers stacked sequentially. The first layer is a dense layer with 10 units and utilizes the rectified linear unit (ReLU) activation function. It serves as the initial layer for processing the input data, which has a shape of (36x1). The second layer is also a dense layer that utilizes ReLU activation and consists of 20 hidden units. Lastly, the model incorporates a dense layer with 10 hidden units, where 10 represents the total number of output classes. This third and final dense layer employs the softmax activation function, which generates probability distributions over the classes.

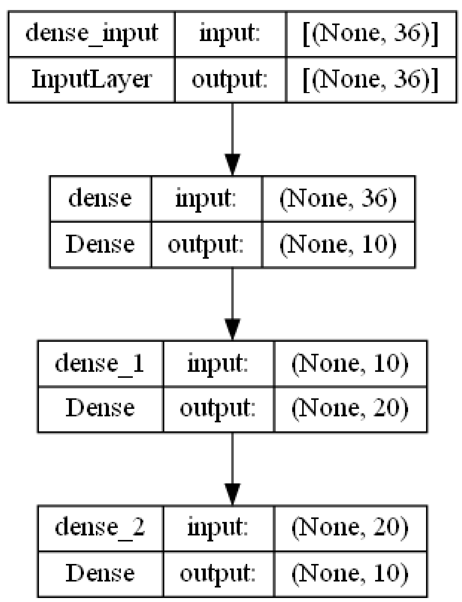

Fig. 3. The Proposed Architecture for the ECG-based Biometric System

F. Deployment on the ESP32 Board

Once the deep learning model is trained, it is converted to TensorFlow Lite (TFLite) format, which is a lighter version of the TensorFlow library designed for devices with low computational power. Despite the smaller size of the TFLite format, it maintains the same accuracy and performance as the original model.

To further prepare the model for deployment, the Edge Impulse Framework [16] is used. Edge Impulse is a framework that supports the developments of TinyML and deep learning models optimized for edge and constrained devices such as microcontrollers. This framework is used to convert the TFLite model into an

Arduino library compatible with the ESP32 microcontroller. When the Arduino library is uploaded to the ESP32, The inference tasks can be run directly on the device.

III. RESULTS AND DISCUSSION

A subset of 10 subjects from the MIT-BIH Arrhythmia Database is used to evaluate the performance of the proposed approach. Before the deployment on the ESP32, the deep learning model described in section 2.E is trained on a machine with an I3-10100F processor, a GTX1650 GPU and 8GB of RAM and compiled with appropriate parameters. Sparse categorical cross-entropy is chosen as the loss function, which is suitable and more adapted for multi-class classification tasks. The Adam optimizer was utilized. A learning rate of 0.001 is set for the optimizer.

The data was split into 70% for training, 10% for validation, and the rest 20% is used to test the performance of the model. The data is divided into mini-batches of 150 samples. The model is trained for 100 epochs. After each epoch, the model's performance is validated using the validation data. The training plots are shown in Fig. 4.

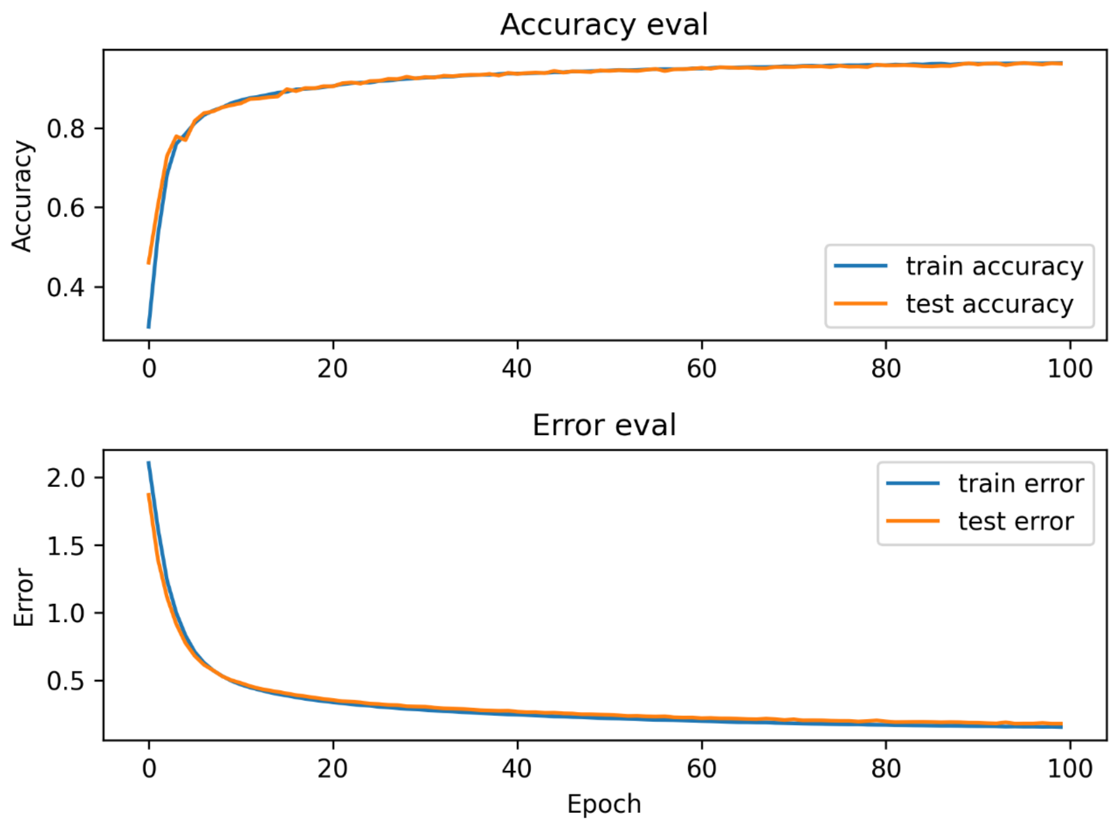

Fig. 4. Plots of accuracy and loss for training and validation sets

The model was evaluated using the accuracy metric and was able to achieve an accuracy of 96.71%. The accuracy can be calculated according to the formula:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

As can be seen on the confusion matrix shown by Fig. 5, our model is able to identify individuals with high accuracy and efficiency. We notice that only 172 QRS complexes are misclassified from a total of 5227 test segments.

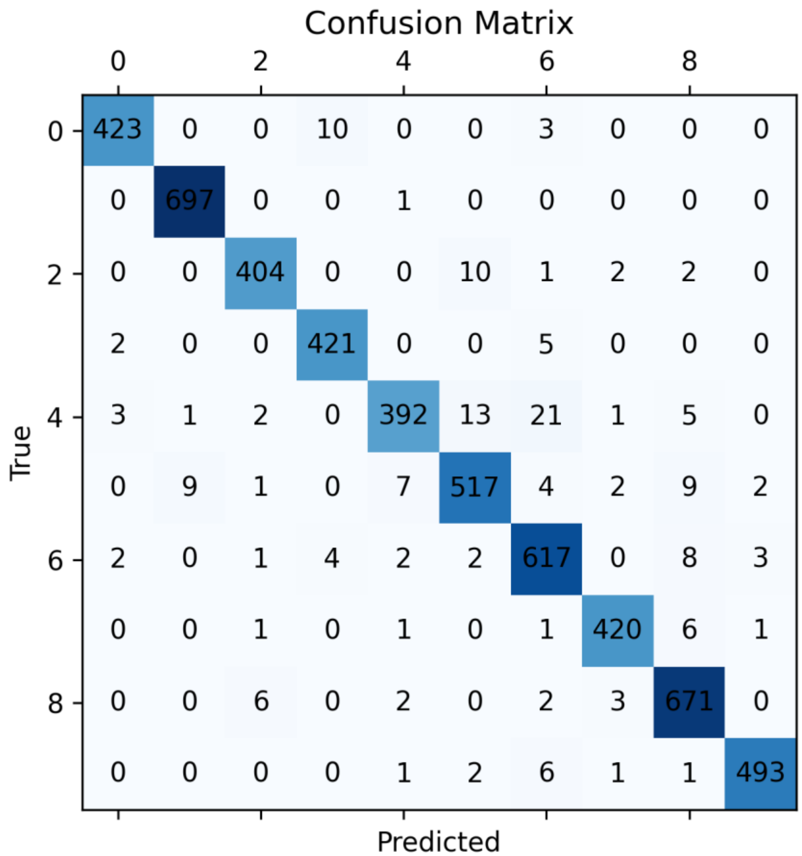

Fig. 5. Confusion matrix under different loading conditions of the proposed method

To evaluate the model on the ESP32 board, many on-device performances were taken into consideration such as the inference time and RAM and ROM usage. Regarding the inference, which represents the time the ESP32 takes to process and classify an input sample, the embedded system was very fast and only required 3ms to predict the corresponding output. One of the other important factors to consider when deploying a deep learning model on a microcontroller is memory usage, the proposed model is compact in size with only 12 Kb of ROM usage and 1.3 Kb of RAM requirements. A fast inference time of just 3 ms is achieved thanks to the Dual-core Tensilica Xtensa LX6 microprocessor running at a maximum frequency of 240 MHz, which the ESP32 is equipped with. The model size does not cause any problems and can be easily deployed on this board as it is equipped with 4 MB of flash memory and 512 KiB SRAM. Fig. 6 shows the classification result of the prediction of a QRS complex belonging to the 10th class (subject 109) on the serial monitor of the Arduino IDE with a level of confidence of 98.407%. As shown in Table 1 and Table 2, the achieved results demonstrate the feasibility of deploying ECG-based biometric systems on an embedded system such as the ESP32 which enables the data processing and reduces the transmission of data to external servers, which will reduce latency and provides the microcontroller with an autonomous work flow. Additionally, the possibility to run a deep learning model on a microcontroller without any external hardware will hugely reduce costs.

Table 1: On-device performance of the Proposed Model on the ESP32

Processing time	RAM usage	Flash usage
3 ms	1.3 k	12 k

Although our model was trained only using a subset of 10 persons from the MIT-BIH Arrhythmia Database, it outperforms other state-of-the-art methods that were evaluated on the same database as shown in Table 2, these results highlight the effectiveness of our model in accurately classifying QRS complexes for biometric identifications. Despite the smaller training data size, our proposed model managed to generalize well on new unseen test data and produce high results. Additionally, our model benefits from the different preprocessing techniques deployed such as filtering, normalization and segmentation applied to the signals of the database.

Fig. 6. Classification Results on the ESP32 displayed on the Serial Monitor of the Arduino IDE

One of the limitations that face our approach is the trade-off between the resource constraints of the ESP32 and the complexity of the deep learning model and the size of the data. Due to the limited hardware resources provided by microcontrollers, the deployed model will be limited in size and number of parameters.

Table 2: Performance comparison of our model with other state-of-the-art approaches

Method	Accuracy
Zhang et al. [17]	91.31%
Zihlmann [18]	91.15%
Zehir et al. [7]	95.31%
Proposed Method	96.71%

IV. CONCLUSION

In conclusion, this paper proposed a biometric system that combines ECG and TinyML for used identification. The model was tested on a subset of 10 individuals from the MIT-BIH Arrhythmia Database. The preprocessing stage involved denoising the ECG signals using a 4th-order Butterworth bandpass filter. The signals were then segmented into 100ms windows centred around each r-peak, detected using the pan-tompkins++ algorithm. To ensure uniformity, the amplitudes of the segmented windows were normalized within the range of 0 to 1. To classify the signals, a deep learning model was trained. The model was successfully deployed on the ESP32 board after it was converted to the TensorFlow Lite format and further transformed into an Arduino library. And, it achieved an identification accuracy of 96.71%.

To overcome the limitation that faces our proposed method, we will focus on our future works on techniques such as model compression and quantization to further reduce the memory usage and inference time while maintaining state-of-the-art accuracy levels.

REFERENCES

[1] D. Belo, N. Bento, H. Silva, A. Fred, and H. Gamboa, 'ECG Biometrics Using Deep Learning and Relative Score Threshold Classification', *Sensors*, vol. 20, no. 15, p. 4078, Jul. 2020, doi: 10.3390/s20154078.

[2] N. Iyengar, C. K. Peng, R. Morin, A. L. Goldberger, and L. A. Lipsitz, 'Age-related alterations in the fractal scaling of cardiac interbeat interval dynamics', *Am. J. Physiol.*, vol. 271, no. 4 Pt 2, pp. R1078-1084, Oct. 1996, doi: 10.1152/ajpregu.1996.271.4.R1078.

[3] G. B. Moody and R. G. Mark, 'The impact of the MIT-BIH Arrhythmia Database', *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, Jun. 2001, doi: 10.1109/51.932724.

[4] H. P. da Silva, A. Lourenço, A. Fred, N. Raposo, and M. Aires-de-Sousa, 'Check Your Biosignals Here: A new dataset for off-the-person ECG biometrics', *Comput. Methods Programs Biomed.*, vol. 113, no. 2, pp. 503–514, Feb. 2014, doi: 10.1016/j.cmpb.2013.11.017.

[5] N. Musa *et al.*, 'A systematic review and Meta-data analysis on the applications of Deep Learning in Electrocardiogram', *J. Ambient Intell. Humaniz. Comput.*, Jul. 2022, doi: 10.1007/s12652-022-03868-z.

[6] Y. Li, Y. Pang, K. Wang, and X. Li, 'Toward improving ECG biometric identification using cascaded convolutional neural networks', *Neurocomputing*, vol. 391, pp. 83–95, May 2020, doi: 10.1016/j.neucom.2020.01.019.

[7] H. Zehir, T. Hafs, S. Daas, and A. Nait-Ali, 'An ECG Biometric System Based on Empirical Mode Decomposition and Hilbert-Huang Transform for Improved Feature Extraction', in *2023 5th International Conference on Bio-engineering for Smart Technologies (BioSMART)*, Jun. 2023, pp. 1–4. doi: 10.1109/BioSMART58455.2023.10162055.

[8] N. E. Huang, Z. Wu, S. R. Long, K. C. Arnold, X. Chen, and K. Blank, 'ON INSTANTANEOUS FREQUENCY', *Adv. Adapt. Data Anal.*, vol. 01, no. 02, pp. 177–229, Apr. 2009, doi: 10.1142/S1793536909000096.

[9] N. E. Huang *et al.*, 'The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis', *Proc. Math. Phys. Eng. Sci.*, vol. 454, no. 1971, pp. 903–995, 1998.

[10] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, 'Efficient Processing of Deep Neural Networks: A Tutorial and Survey', *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017, doi: 10.1109/JPROC.2017.2761740.

[11] C. R. Banbury *et al.*, 'Benchmarking TinyML Systems: Challenges and Direction'. arXiv, Jan. 29, 2021. Accessed: Jul. 16, 2023. [Online]. Available: http://arxiv.org/abs/2003.04821

[12] P. Warden and D. Situnayake, *TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. O'Reilly Media, Inc., 2019.

[13] 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, Jun. 2000, doi: 10.1161/01.cir.101.23.e215.

[14] L. Biel, O. Pettersson, L. Philipson, and P. Wide, 'ECG analysis: a new approach in human identification', *IEEE Trans. Instrum. Meas.*, vol. 50, no. 3, pp. 808–812, Jun. 2001, doi: 10.1109/19.930458.

[15] M. N. Imtiaz and N. Khan, 'Pan-Tompkins++: A Robust Approach to Detect R-peaks in ECG Signals', in *2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Dec. 2022, pp. 2905–2912. doi: 10.1109/BIBM55620.2022.9995552.

[16] S. Hymel *et al.*, 'Edge Impulse: An MLOps Platform for Tiny Machine Learning'. arXiv, Apr. 28, 2023. doi: 10.48550/arXiv.2212.03332.

[17] Q. Zhang, D. Zhou, and X. Zeng, 'HeartID: A Multiresolution Convolutional Neural Network for ECG-Based Biometric Human Identification in Smart Health Applications', *IEEE Access*, vol. 5, pp. 11805–11816, 2017, doi: 10.1109/ACCESS.2017.2707460.

[18] M. Zihlmann, D. Perekrestenko, and M. Tschannen, 'Convolutional recurrent neural networks for electrocardiogram classification', in *2017 Computing in Cardiology (CinC)*, IEEE, 2017, pp. 1–4.