Robust deep learning pipeline for PVC beats localization

Bohdan Petryshak[a,*], Illia Kachko[a,*], Mykola Maksymenko[b] and Oles Dobosevych[a]
[a]*The Machine Learning Laboratory, Ukrainian Catholic University Lviv, Ukraine*
[b]*SoftServe Inc., Austin, TX, USA*

Abstract.
BACKGROUND: Premature ventricular contraction (PVC) is among the most frequently occurring types of arrhythmias. Existing approaches for automated PVC identification suffer from a range of disadvantages related to hand-crafted features and benchmarking on datasets with a tiny sample of PVC beats.
OBJECTIVE: The main objective is to address the drawbacks described above in the proposed framework, which takes a raw ECG signal as an input and localizes R peaks of the PVC beats.
METHODS: Our method consists of two neural networks. First, an encoder-decoder architecture trained on PVC-rich dataset localizes the R peak of both Normal and anomalous heartbeats. Provided R peaks positions, our CardioIncNet model does the delineation of healthy versus PVC beats.
RESULTS: We have performed an extensive evaluation of our pipeline with both single- and cross-dataset paradigms on three public datasets. Our approach results in over 0.99 and 0.979 F1-measure on both single- and cross-dataset paradigms for R peaks localization task and above 0.96 and 0.85 F1 score for the PVC beats classification task.
CONCLUSIONS: We have shown a method that provides robust performance beyond the beats of Normal nature and clearly outperforms classical algorithms both in the case of a single and cross-dataset evaluation. We provide a Github[1] repository for the reproduction of the results.

Keywords: Electrocardiography, PVC identification, ECG segmentation, ECG classification

1. Introduction

Premature ventricular contraction (PVC) is the common arrhythmia beat, often occurring in many patients, including those in good health [1]. Automated detection of the PVCs in ECG signals would allow doctors to catch the long-term frequency of abnormal beats providing valuable insights about the patient's cardiac health and early notifications about events requiring medical attention.

The primary aim of this study is to develop the pipeline, which will automatically localize the PVC beats on the noisy ECG signal and do not need any preprocessing steps with medical domain knowledge or manually crafted features and generalizes well to new patients and signal sources. For this, we use a two-step approach. The first step is the encoder-decoder architecture. Trained on a PVC-rich dataset, it localizes R peaks both for Normal and anomalous heartbeats. The second step is the adopted for ECG signal data 1D convolutional InceptionTime neural network which we dubbed CardioIncNet. This network

---

*Corresponding authors: Bohdan Petryshak and Illia Kachko, The Machine Learning Lab, Ukrainian Catholic University Lviv, Ukraine. E-mails: petryshak@ucu.edu.ua; mmaks@softserveinc.com.
[1]https://github.com/ucuapps/Robust-DL-pipeline-for-PVC-localization.

takes the heartbeat as an input and performs the healthy versus PVC bits classification. To our knowledge, it is the first application of InceptionTime architecture to the ECG signal and our results showcase comparable state-of-the-art performance of such an approach. We perform a thorough evaluation of our approach on the PVC-rich dataset on both patient and beat levels. For this purpose, we tested it on three available datasets, which include MIT-BIH [2], MIT-SVDB [3], and the 3rd China Physiological Signal Challenge 2020 dataset [4].

The paper is organized as follows: Section 2 provides background and related works. Section 3 describes all datasets we used in this work; Section 4 presents all our method details. In Section 5 we describe the preprocessing steps, training details and evaluation approaches of the framework. In Section 6, we present the obtained results. Finally, we conclude the paper in Section 7.

2. Related works

2.1. R peak detection

There are many approaches to R peaks detection – from classical algorithms to modern methods using deep learning. The Pan-Tompkins algorithm [5] is one of the classical signal processing pipelines for peak detection that is broadly used in research and industry. This algorithm is useful for finding peaks on a healthy signal, but it loses accuracy on a sick person's signal. For example, in PVC beats R peaks are located in specific locations. The filters cannot delineate them, and subsequently, the algorithm often misses pathological beats. Some handcrafted modifications to the algorithms allow, to some extent, tackle this issue.

Recent developments use the Deep Neural Networks approach to detect the QRS complex. Neural Network approaches can perform very well on a raw signal while avoiding handcrafted features or decision rules. In [6], a simple 1-D CNN was used that was trained on [3] and achieved very high results (sensitivity of 99.81%) on detecting QRS complexes. In [7] the U-Net architecture was utilized to automate beat-wise arrhythmia diagnosis on extended electrocardiographic recordings with heterogeneous arrhythmia types. Although the paper shows high accuracy results, the authors show them only on the MIT-BIH dataset subset with the intra-patient splitting (used the same patients for training and testing). In [8] the authors pushed the idea of using Convolutional Neural Networks even further and used deep fully convolutional network architecture called U-Net [9] that not only achieved a high F1 score on detecting QRS complex but was capable of detecting P and T offset. Although producing high results, the method cannot be considered reliable for R-peak detection as it was trained on the dataset with a very low occurrence of arrhythmia peaks (only eight records out of 200) [10]. In this study, we extend this approach to the PVC-rich datasets.

2.2. Classification

The arrhythmia classification is generally a long-standing problem. For the last 20 years, dozens of classical machine learning [11–14] and template matching [15,16] approaches have been applied to it and are often used in industrial applications. However, these methods extremely dependent on specific ECG devices, signal noise, and handcrafted features.

Deep learning has shown promising results in numerous AI applications [17–19]. There is a vast amount of arrhythmia detection approaches that use deep learning strategies. We will do a short overview of the central ideas on the main neural network types used in the classification task.

2.2.1. Convolutional neural networks

Kiranyaz et al. [20,21] adopted a 1D convolutional neural network (CNN) to detect ventricular ectopic beats (VEBs) and supraventricular ectopic beats (SVEBs) with an accuracy of 99% and 97.6%, respectively. They used 44 patients from MIT-BIH arrhythmia dataset [2] for training and testing their models. In [22] study, used as an input to CNN converted 1D ECG signals from the MIT-BIH arrhythmia database into 2D images via one-hot encoding. The authors demonstrated that the classification accuracy of six types of arrhythmia on 2D input was significantly higher than the 1D signal. In [23] authors proposed another way of converting the ECG signal into 2D images using the Short-time Fourier transform. They proposed specific data augmentation strategies for such ECG representation and confirmed a hight performance of such a method.

2.2.2. Recurrent neural networks

The second idea, which is widely presented in the literature, is that the ECG signal may be treated as time series [24] showed improved performance of detecting VEBs and SVEBs by implementing a patient-specific ECG model based upon recurrent neural networks and density-based clustering algorithm [25]. In [26] it is found that combination of lead convolutional neural network (LCNN), long short-term memory (LSTM) [27] and rules inference shows superior results in Premature ventricular contraction (PVC) beats classification. The authors used MIT- BIH arrhythmia and CCDD [28] databases for training and validating their methods. Our approach is based on the CNN type. We have adopted the classification architecture specially designed for signals into the PVC classification domain. We provide all details about it in the Methodology section.

3. Datasets

For our experiments, we took three publicly available datasets: MIT-BIH Arrhythmia Database [2] (recorded from 48 subjects with the labels for R peaks and PVCs locations), MIT-BIH Supraventricular Arrhythmia Database [3] (contains labels for R peaks and PVCs locations), and dataset from The 3rd China Physiological Signal Challenge 2020 (CSPC) [4] (42075 PVC beats from a total of more than a million beats).

We took the ECG1 and MLII (modified limb II) leads for MIT-SVDB and MIT-BIH datasets, respectively. For CSPC 2020 no data of the lead type is provided. Considering the morphology of the normal and PVC beats, we assume that it is most likely ECG1 or lead II. Although these leads are different, we can perform the cross-dataset evaluation on all three datasets, as all the leads measure the electrical potential difference in the same direction, and the morphology of the normal and PVC beats is similar enough.

4. Method

Our pipeline consists of two parts: the segmentation model based on inception U- Net architecture [29], which finds R waves on the ECG signal and novel classification network CardioIncNet based on In-ceptionTime architecture (refer to the Fig. 1), which does a PVC versus Normal binary classification of each found heartbeat. To the best of our knowledge, this is the first time, InceptionTime architecture was adopted for ECG signal classification.
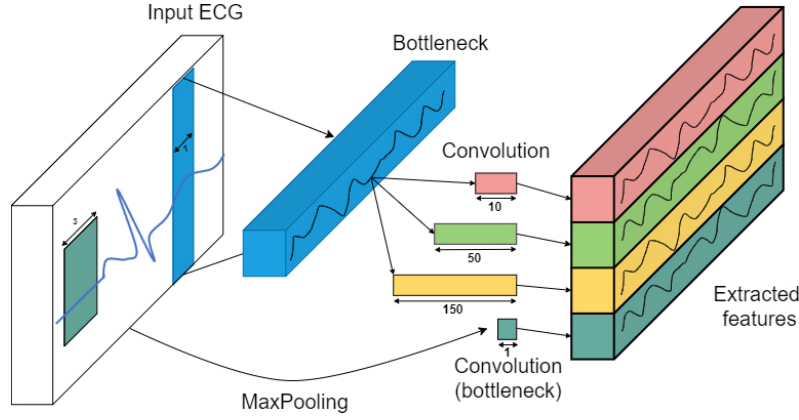
Fig. 1. InceptionTime module, used for PVC beats classification. The bottleneck layer reduces the dimension of the input data and models complexity mitigating the over- fitting. Multi-scale convolutional windows effectively learn patterns at different scales of the abnormal cardiac cycle.

We have done a thorough PVC oriented cross-dataset evaluation of proposed deep learning approaches on the R peak segmentation and PVC beats classification task using a test set that abounds with PVC beats. We also have compared the performance of deep learning models with the existing traditional approaches on the R peaks localization task.

4.1. R-peak segmentation

4.1.1. Problem formulation

The problem is formulated as a regression task of getting smoothed by convolution R peaks locations from the input ECG signal. The dataset $S = (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(n)}, y^{(n)})$, consists of input signal $x^{(i)}$, and corresponding ground truth $y^{(i)}$, where $(x^{(i)}, y^{(i)} \in R^n)$. n is the number of input signals, which are presented in the training set. The $y^{(i)}$ was created from the R peaks locations. The shape of $y^{(i)}$ is the same size as an input $x^{(i)}$.

The proposed segmentation model is the Encoder-Decoder fully convolutional architecture. It consists of two paths. The first one is the contraction path, which takes $x^{(i)}$ as input and produces the compressed feature vector. The vector is then upsampled in the expanding path and produces the *segmentation output*$^{(i)}$, which is the same size as $x^{(i)}$. The following equation reflects the architecture:

$$segmentation\ output^{(i)} = F\left(x^{(i)}; \theta\right) \tag{1}$$

where F is the segmentation neural network function, θ is the weights of the net-work. The weights of the proposed architecture have been optimized by minimizing the *SmoothL$_1$* [30], which is less sensitive to outliers than L2 loss and prevents the exploding gradients [30].

4.1.2. Model architecture

The segmentation part has been adapted from the RPnet network [31]. It is an Encoder-Decoder architecture that takes in a chunk of ECG record with the input size of (1, 5000) and consists of convolutional Inception-Resblocks. The encoder part downsamples the input signal 8 times with the help of 8 layers of strided convolution. Downsampling by a factor of 2 is achieved through the 1D convolution filters with the stride 1 and kernels size 4. On every convolutional block, the number of filters is multiplied

by 2 until 1024, after which it is kept constant. The Batch Normalization, leaky ReLU activation with the slope of 0.2 and Inception-Resblock have been applied after each 1D convolution layer. The output is then upsampled with the help of Transpose convolutions in the decoder part. More details about the model architecture can be found in the original paper [31].

4.2. PVC Classification

4.2.1. Problem formulation

To perform the classification of PVC disease, the softmax regression model is used as a last layer of the ECG signal 1D convolutional neural network, a novel CardioIncNet architecture. The training set $C = (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(n)}, y^{(n)})$, where n is the number of ECG beats, which has class labels. $x^{(i)}$ is an ECG fragment, which represents the entire heartbeat. $y^{(i)} \in 0, 1$ if the class for $x^{(i)}$ (0 and 1 are the labels for normal and PVC beats correspondingly). For an ECG beat $x^{(i)}$, the output for the CardioIncNet network is *classification output*$^{(i)}$, as shown in Eq. (2).

$$classification\ output^{(i)} = L\left(x^{(i)}; \theta\right), \tag{2}$$

where L is the CardioIncNet architecture function, and θ is the corresponding pa- rameters of the model. The *classification output*$^{(i)}$ from the last CardioIncNet layer is the vector, containing the features, which have been extracted with the convolutional part of the network from the raw ECG signal. It is then fed to the fully connected layer, which calculates the output weights for each heartbeat category. We use the softmax function for the raw output of the network to turn them into probabilities Eq. (3):

$$\hat{y} = \frac{eps(z^i)}{\sum_{j=1}^{C} eps\left(z^j\right)}, \tag{3}$$

where C is the number of heartbeat categories, $\hat{y}$ is the probability assigned to each class by the network.

4.2.2. Model architecture

The central ingredient of each module is the "bottleneck" block, which reduces the dimension of the data and model's complexity, decreasing overfitting problems. The bottleneck layer has a dimensionality of 1. The second main component is large sliding multiple convolutional windows simultaneously on the same input signal with length 10, 50, 150. Its main purpose is to effectively learn long-range abnormality features within P-waves, QRS complexes, and cardiac cycles. To make our model invariant to small signal perturbations we added the parallel MaxPooling operation. We utilized the residual connections [32] and batch normalization layers [33] to improve the model's convergence during training.

5. Experiments

5.1. Data pre-processing

To preserve the consistency across the datasets, which we used for training and testing our models, we resampled all datasets to the 250 Hz frequency, which is an average frequency over three databases, we used in this research.

ECG signals may be affected by several types of noise. The first one is the baseline wander, which appears when patients move during the ECG recording procedure. The component of the signal usually has frequencies below 0.5 Hz. The second type of noise is a muscle artifact noise that is a result of an

electrical muscle activity and is challenging to remove due to the overlap in frequency with the QRS complex. The last one is the power line interference which is caused by external electromagnetic devices.

It is crucially to note that the denoise algorithms should clean the signal meantime keeping the key features of beats in terms of R peak and PVC disease. We have pre-processed the signal with different strategies for the segmentation and classification models. To minimize the difference of ECG signals between datasets, we normalized all data between 0 and 1. The normalization was performed beat-wise. We have considered second-order band-pass FIR filter with frequency cut-off at 10 Hz to eliminate the described types of noise.

Both datasets MIT-BIH and MIT-SVDB, which we used for training, are extremely imbalanced. We did not use any resampling techniques for the segmentation task. However, we balanced the amount of PVC and normal beats for classification. We used random undersampling of majority(normal) class [34]. For training on each fold, we used 4000 and 6000 beats of both classes (PVC and normal) in MIT-BIH and MIT-SVDB, respectively. For each fold, we have used a 60/20/20 train/val/test split scheme.

5.2. Implementation details

The models were trained on an NVIDIA 2080 RTX and written in Python with the PyTorch library. Both segmentation models had ADAM optimizer. As a loss, we chose L1SmoothLoss and MSE. Models were trained with different initial learning rates. UNet has 10^{-3}, and Inception-UNet has $0.5 * 10^{-1}$. The learning rate was decreased by a factor of 10 on a moment of loss change plateau. For this, the ReduceRLOnPLateau function was used from the PyTorch library.

5.3. Evaluation metrics

5.3.1. Classification
Four main evaluation metrics were considered to show the performance of the classification techniques: precision, recall, F1-score, and accuracy.

Calculation of these measures are given as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{FP} + \text{FN}} \tag{6}$$

$$\text{F1} = 2 \times \frac{\text{PRECISION} \times \text{RECALL}}{\text{PRECISION} + \text{RECALL}}. \tag{7}$$

Here, TP is the true positive, TN denotes true negative, FN denotes false negative, and finally, FP denotes false positive.

5.3.2. Segmentation
For an R peak segmentation problem, we propose the following strategy. If $\hat{y}_i$ is a location of a predicted peak, y_i is a nearest ground truth peak and s is a constant value(for this constant we used an average duration of the QRS complex times 0.3, as ground truth labels of PVC beats on the test set are within the QRS complex range), then predicted peak considered to be found if $|\hat{y}_i - y_i| < s$. We have resampled MIT-BIH and SVDB to one frequency (250 Hz) and after that used the length between Q and S waves

to get the average size of the QRS complex and used the same average size for all datasets. We have used precision, recall, and F1-score (refer to Eqs (5)–(7) formulas) for measuring the quality of the segmentation model. In this formulation, we can interpret given formulas in such a way:

- TP – number of predicted peaks, that were close to the ground truth (less than s).
- FP – number of predicted peaks, that was far from the ground truth (more than s).
- FN – number of missed ground truth peaks.

5.4. Single- and Cross-dataset evaluation paradigm

5.4.1. Single-dataset evaluation

We used the inter-patient paradigm for single-dataset evaluation and further investigated the evaluation strategies of deep learning algorithms and tested the models' cross-database generalization capabilities. Most of the literature focused on detecting PVC disease showcase the results on the subset of the MIT-BIH dataset, which is up to 4K PVC peaks on the test set only from a few patients. In this study, we train our methods on one or a mixed set of datasets (MIT-BIH, MIT-SVDB) and test on the separate dataset (The 3rd China Physiological Signal Challenge 2020 data) with near 40K of PVC beats (see "Pre-processing" and "Dataset" sections for more details).

6. Results

6.1. Single dataset evaluation

We have employed four-fold cross-validation to report the results of our models on a single datasets paradigm. We have applied this approach separately for both MIT-BIH [2] and MIT-SVDB [3] databases. We have split the training dataset into test and training part. The test set was chosen manually. We selected 30% of all PVC beats in the dataset for the test set. Then we randomly divided the remaining training set into four equal folds. 75% heartbeats were used for training, and 25% was used for choosing the best model on the validation set. This method was repeated four times by rotating the validation set. Then the performance metrics (for more details, see Section 5.3) are computed on the test set on every fold rotation. Eventually, the final score of the model is obtained by averaging all metrics recorded in all four folds.

6.1.1. Segmentation

We present results for the standard U-Net architecture and our Inception-U-Net in Table 1 showcasing better performance of the latter on both MIT- BIH and MIT-SVDB datasets. More than 97% R peaks localized correctly, suggesting our model performs well on one dataset cross-validation task similar to state of the art results on Normal beats reported in [6]. We can compare these results with the performance of the classical algorithms in Table 2. These end-to-end results provided by DNN models clearly outperform out of the box classical algorithms like Pan-Tompkins, Wavelet-based Pan-Tompkins [35], Christov [36], Two Moving Average [37] and Hamilton [38]. For running the algorithms on MIT-BIH and MIT-SVDB, we took the implementation from this GitHub[2] [39].

---

[2]https://github.com/berndporr/py-ecg-detectors.

Table 1
R-peaks segmentation cross-validation results

Dataset	Model	F1	Accuracy	Precision	Recall
MIT-BIH	Net	0.9923	0.9889	0.9957	0.9889
	Inc-UNet	0.9951	0.9938	0.9965	0.9938
MIT-SVDB	UNet	0.9646	0.9645	0.9647	0.9645
	Inc-UNet	0.9694	0.9691	0.9697	0.9691

Table 2
Classical algorithms results comparison for the R peaks detection

Dataset	Algorithm	F1	Accuracy	Precision	Recall
MIT-BIH	Pan-Tompkins	0.917	0.908	0.929	0.908
	Pan-Tompkins SWT	0.958	0.943	0.977	0.943
	Hamilton	0.933	0.943	0.977	0.943
	Christov	0.737	0.966	0.684	0.966
	Two-Average	0.797	0.849	0.759	0.849
MIT-SVDB	Pan-Tompkins	0.184	0.174	0.183	0.174
	Pan-Tompkins SWT	0.178	0.168	0.204	0.168
	Hamilton	0.178	0.177	0.201	0.177
	Christov	0.246	0.327	0.191	0.327
	Two-Average	0.191	0.189	0.192	0.189

Table 3
PVC beats classification cross-validation results

Dataset	Model	F1	Accuracy	Precision	Recall
MIT-BIH	Bi-LSTM	0.927	0.927	0.927	0.927
	Ours	0.964	0.965	0.966	0.964
	ResNet	0.883	0.886	0.889	0.883
MIT-SVDB	Bi-LSTM	0.9033	0.904	0.905	0.903
	Ours	0.962	0.9605	0.961	0.9615
	ResNet	0.814	0.8175	0.82075	0.814

6.1.2. Classification

Table 3 presents the performance metrics on MIT-BIH and MIT- SVDB datasets. We have compared our model with two other architectures, which are widely used for arrhythmia classification. Our model shows excellent performance on both datasets comparable to results reported in the literature [40–42]. As all datasets are extremely imbalanced and they contain a minimum 10 times more normal bits than PVC beats, the class imbalance negatively affects the classification model's performance. We refer to this problem with the random undersampling of the majority class. To avoid losing the majority class information, we took a random undersample from all sets of majority class every epoch. We compare the performance of our CardioIncNet model with Bi-LSTM and ResNet architectures that were discussed in [6,21,43].

6.2. Cross-dataset evaluation

To check the cross-dataset generalizability of our models, we have trained the algorithms on one dataset and tested it on another. For this purpose, we used three separate datasets. We have split the training dataset into the training and validation parts and tested the best model, chosen on the validation set on another dataset as a test set.

Table 4
R-peaks Segmentation cross-dataset results

Train dataset	Test dataset	Model	F1	Accuracy	Precision	Recall
MIT-BIH	MIT-SVD	UNet	0.9795	0.9783	0.9807	0.9783
		Inc-UNet	0.9726	0.9650	0.98	0.965
MIT-SVDB	MIT-BIH	UNet	0.9759	0.9724	0.9795	0.9724
		Inc-UNet	0.9819	0.9845	0.9794	0.9845
MIT-BIH	CSPC2020	UNet	–	–	–	0.9883
		Inc-UNet	–	–	–	0.9901
MIT-SVDB	CSPC2020	UNet	–	–	–	0.9913
		Inc-UNet	–	–	–	0.9965

Table 5
PVC beats classification cross-dataset results

Train dataset	Test dataset	Model	F1	Accuracy	Precision	Recall
MIT-BIH	MIT-SVDB	Bi-LSTM	0.833	0.834	0.833	0.833
		Ours	0.854	0.85	0.845	0.854
		ResNet	0.826	0.823	0.828	0.824
MIT-SVDB	MIT-BIH	Bi-LSTM	0.94	0.94	0.942	0.938
		Ours	0.946	0.946	0.946	0.945
		ResNet	0.911	0.909	0.910	0.913
MIT-BIH	CSPC2020	Bi-LSTM	–	–	–	0.788
		Ours				0.807
		ResNet	–	–	–	0.701
MIT-SVDB	CSPC2020	Bi-LSTM	–	–	–	0.792
		Ours	–	–	–	0.813
		ResNet	–	–	–	0.702

6.2.1. R-peaks segmentation

The R-peak segmentation generalizes well if we use the MIT-SVDB dataset for testing the model trained on MIT-BIH and vice versa (see Table 4). Here the F1 score is 0.972 and 0.982 correspondingly. The models which were trained on MIT-BIH or MIT-SVDB performed worse on CSPC 2020, as data were collected by a unified wearable ECG device in contrast to MIT-BIH and MIT-SVDB. We do not provide the cross dataset results for models trained on CSPC 2020, as this dataset contains only locations for the PVC beats, without Normal class.

6.2.2. Classification

Table 5 demonstrates generalization capabilities of our model in case of cross-dataset evaluation. While such cross dataset evaluation is not directly com- parable with the result reported in the literature, the high F1 score of the order 0.85 and 0.95 is qualitatively comparable to other large scale experiments in the literature [40–42].

7. Conclusions and future work

In this study, we propose an end-to-end framework that automates the process of PVC beats localization. It is based on a two-step approach in which an encoder-decoder U-Net-like neural network trained on PVC-rich datasets is used to segment R-peaks of normal and anomalous beats. Provided R-peak segmentation, an InceptionTime architecture, which we dubbed here CardioIncNet, classifies which of these beats are of PVC type. To our knowledge, this is the first study in which neural architectures are applied for

segmentation of anomalous beats and IncpetionNet-like architecture is used for the ECG classification task. Our results are comparable to the state of the art approaches reported in the literature [40–42]. Also similar kinds of cross dataset experiments were reported by [44] for beats of ventricular origin resulting in sensitivity and precision metrics comparable to our results, i.e. precision 88.8 for Lead II configuration.

We provide extensive experimental results for both single and cross-dataset paradigms. First, our R-peak segmentation pipeline provides a robust performance beyond Normal- like beats and clearly outperforms classical algorithms both in the case of a single and cross-dataset evaluation. Second, our CardioIncNet model demonstrates state of the art results on a single-dataset evaluation and provides high performance on cross-dataset check, where comparison to literature results is less applicable. Somewhat lower performance on a cross dataset evaluation suggests that such method requires fine tuning when applied in practice to a data from a new device or lead position.

Future work may involve minor changes in the framework formulation, signal filtering and training process to make it end-to-end for the cross-dataset evaluation. Furthermore, it is desirable to evaluate and compare the computational complexity of the method to make it suitable for real-time usage.

Acknowledgments

Conflict of interest

None to report.

References

[1] Stanfield CL, Germann WJ. Principles of human physiology. Pearson Benjamin Cummings, 2008.

[2] Moody GB, Mark RG. The impact of the MIT-BIH arrhythmia database. In: IEEE engineering in medicine and biology magazine: the quarterly maga- zine of the Engineering in Medicine Biology Society. 2001; 20: 45-50.

[3] Greenwald SD, Patil RS, Mark RG. Improved detection and classification of arrhythmias in noise-corrupted electrocardiograms using contextual information. In: 1990; pp. 461-464.

[4] The 3rd China Physiological Signal Challenge 2020. http//www.icbeb.org/: CSPC2020.

[5] Pan J, Tompkins WJ. A Real-Time QRS Detection Algorithm. IEEE Transactions on Biomedical Engineering BME. 1985; 32(3): 230-236.

[6] Sărlija M, Jurišć F, Popović S. A convolutional neural nework based approach to QRS detection. In: Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis. 2017, pp. 121-125.

[7] Oh SL, et al. Automated beat-wise arrhythmia diagnosis using modified U-net on extended electrocardiographic recordings with heterogeneous arrhythmia types. Computers in Biology and Medicine. 2018; 105.

[8] Moskalenko V, Zolotykh N, Osipov G. Deep Learning for ECG Segmentation. In: Advances in Neural Computation, Machine Learning, and Cognitive Research III, 2019; pp. 246-254.

[9] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. 2015. arXiv: 150504597. [cs.CV].

[10] Kalyakulina AI, et al. LU electrocardiography database: a new open-access validation tool for delineation algorithms. 2018. arXiv: 180903393. [q-bio.QM].

[11] Ye C, Vijaya Kumar BVK, Coimbra MT. Combining general multiclass and specific two-class classifiers for improved customized ECG heartbeat classification. In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012). 2012, pp. 2428-2431.

[12] Ye C, Vijaya Kumar BVK, Coimbra MT. Heartbeat classification using morphological and dynamic features of ECG signals. IEEE Transactions on Biomedical Engineering. 2012; 59(10): 2930-2941.

[13] Venkatachalam M. ECG arrhythmia classification based on logistic model tree. Journal of Biomedical Science and Engineering. 2009; 2: 405-411.

[14] Exarchos T, et al. A methodology for the automated creation of fuzzy expert systems for ischaemic and arrhythmic beat classification based on a set of rules obtained by a decision tree. Artificial Intelligence in Medicine. 2007; 40: 187-200.

[15] Krasteva V, Jekova I. QRS template matching for recognition of ventricular ectopic beats. Annals of Biomedical Engineering. 2008; 35: 2065-76.

[16] Luker J, et al. Automated template matching correlates with earliest activation during mapping of idiopathic premature ventricular contractions. IJC Heart Vessels. 2014; 4.

[17] Krizhevsky A, Sutskever I, Hinton G. ImageNet Classification with deep convolutional neural networks. Neural Information Processing Systems. 2012; 25.

[18] Vaswani A, et al. Attention Is All You Need. 2017. arXiv: 1706. 03762 [cs.CL].

[19] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional etworks for Biomedical Image Segmentation. 2015. arXiv: 150504597. [cs.CV].

[20] Kiranyaz S, Gabbouj M. Real-time patient-specific ECG classification by 1-D convolutional neural networks. IEEE Transactions on Biomedical Engineering. 2016; 63(3): 664-675.

[21] Cláudia B, Ana M, António S. Electrocardiogram Beat Classification Based on a ResNet Network. In: Studies in Health Technology and Informatics 264.MEDINFO 2019: Health and Wellbeing e-Networks for All (2019).

[22] Li J, et al. Deep convolutional neural network based ecg classification sys- tem using information fusion and one-hot encoding techniques. In: Mathematical Problems in Engineering. 2018, pp. 1-10.

[23] Ullah A, et al. Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation. In: (Apr. 2020).

[24] Zhang C, et al. Patient-specific ECG classification based on recurrent neural networks and clustering technique. In: 2017 13th IASTED International Conference on Biomedical Engineering (BioMed). 2017, pp. 63-67.

[25] Rodriguez A, Laio A. Clustering by fast search and find of ensity peaks. Science. 2014; 344(6191): 1492-1496.

[26] Zhou F, Jin L, Dong J. Premature ventricular contraction detection combining deep neural networks and rules inference. In: Artificial Intelligence in Medicine. 2017; 79.

[27] Hochreiter S, Schmidhuber J. Long Short-term memory. Neural Computation. 1997; 9(8): 1735-1780.

[28] Zhang J, Liu X, Dong J. CCDD: An enhanced standard ecg database with its management and annotation tools. In: International Journal on Artificial Intelligence Tools 21 (Nov. 2012).

[29] Vijayarangan S, et al. RPnet: A Deep Learning approach for robust R eak detection in noisy ECG. 2020. arXiv: 200408103. [eess.SP].

[30] Girshick R. Fast R-CNN. 2015. arXiv: 150408083. [cs.CV].

[31] Vijayarangan S, et al. RPnet: A Deep Learning approach for robust R Peak detection in noisy ECG. 2020. arXiv: 200408103. [eess.SP].

[32] He K, et al. Deep Residual Learning for Image Recognition. 2015. arXiv1512.03385: [csCV].

[33] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015. arXiv: 1502. 03167.

[34] More A. Survey of resampling techniques for improving classification performance in unbalanced datasets. 2016. arXiv: 160806048. [stat.AP].

[35] Kalidas V, Tamil L. Real-time QRS detector using Stationary Wavelet Transform for Automated ECG Analysis. In: Oct. 2017, pp. 457-461.

[36] Christov II. Real time electrocardiogram QRS detection using combined adaptive threshold. In: BioMedical Engineering OnLine. 2004; 3(1).

[37] Elgendi M, et al. Frequency bands effects on qrs detection. In: Proceedings of the Third International Conference on Bio-inspired Systems and Signal Processing. SciTePress – Science, and Technology Publications, 2010.

[38] Hamilton P. Open source ECG analysis'. In: Computers in Cardiology. 2002, pp. 101-104.

[39] Luis H, Porr B. Popular ECG R peak detectors written in python. en. 2019.

[40] Rajpurkar P, et al. Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks. 2017. arXiv: 170701836. [cs.CV].

[41] Jun TJ, et al. Premature Ventricular Contraction Beat Detection with Deep Neural Networks. In: 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA). 2016, pp. 859-864.

[42] Yan G, et al. Fusing Transformer Model with Temporal Features for ECG eartbeat Classification. In: 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). 2019, pp. 898-905.

[43]   Gao J, et al. An effective LSTM recurrent network to detect arrhythmia on imbalanced ECG dataset. Journal of Healthcare Engineering. 2019, pp. 1-10.

[44]   Ortín S, et al. Automated real-time method for ventricular heartbeat classification. In: Computer Methods and Programs in Biomedicine. 2019; 169; 1-8.