



# Simultaneous ECG Heartbeat Segmentation and Classification with Feature Fusion and Long Term Context Dependencies

Xi Qiu<sup>1,2</sup>, Shen Liang<sup>1,2</sup>, and Yanchun Zhang<sup>2,3</sup>(✉)

<sup>1</sup> School of Computer Science, Fudan University, Shanghai, China  
{17210240279,sliang11}@fudan.edu.cn

<sup>2</sup> Cyberspace Institute of Advanced Technology (CIAT), Guangzhou University, Guangzhou, China

<sup>3</sup> Institute for Sustainable Industries and Liveable Cities, Victoria University, Melbourne, Australia  
Yanchun.Zhang@vu.edu.au

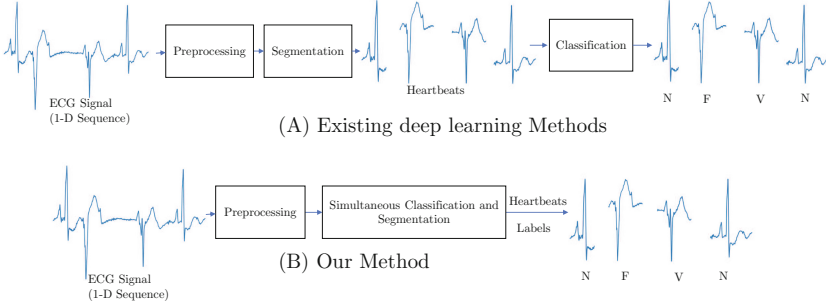
**Abstract.** Arrhythmia detection by classifying ECG heartbeats is an important research topic for healthcare. Recently, deep learning models have been increasingly applied to ECG classification. Among them, most methods work in three steps: preprocessing, heartbeat segmentation and beat-wise classification. However, this methodology has two drawbacks. First, explicit heartbeat segmentation can undermine model simplicity and compactness. Second, beat-wise classification risks losing inter-heartbeat context information that can be useful to achieving high classification performance. Addressing these drawbacks, we propose a novel deep learning model that can simultaneously conduct heartbeat segmentation and classification. Compared to existing methods, our model is more compact as it does not require explicit heartbeat segmentation. Moreover, our model is more context-aware, for it takes into account the relationship between heartbeats. To achieve simultaneous segmentation and classification, we present a Faster R-CNN based model that has been customized to handle ECG data. To characterize inter-heartbeat context information, we exploit inverted residual blocks and a novel feature fusion subroutine that combines average pooling with max-pooling. Extensive experiments on the well-known MIT-BIH database indicate that our method can achieve competitive results for ECG segmentation and classification.

**Keywords:** Arrhythmia detection · ECG classification · End-to-end deep neural network · Heartbeat segmentation

## 1 Introduction

Arrhythmia occurs when the heart rhythms are irregular, which can lead to serious organ damage. Arrhythmias can be caused by high blood pressure, heart

diseases, etc [1]. Electrocardiogram (ECG) is one of the most popular tools for arrhythmia diagnosis. To manually handle long ECG recordings with thousands of heartbeats, clinicians have to determine the class of each heartbeat to detect arrhythmias, which is highly costly. Therefore, great efforts have been made to create computer-aided diagnosis tools that can detect irregular heartbeats automatically.



**Fig. 1.** The difference between our method and existing deep learning based methods

In recent years, deep learning models have been gradually applied to ECG classification. Among them, most methods work in three steps: preprocessing, heartbeat segmentation and beat-wise classification (see Sect. 2). The preprocessing step removes various kinds of noise from raw signals, the heartbeat segmentation step identifies individual heartbeats, and the beat-wise classification step classifies each heartbeat. This methodology has the following drawbacks: First, explicit heartbeat segmentation can undermine model simplicity and compactness. Traditional heartbeat segmentation methods explicitly extract ECG features for QRS detection. Since deep learning methods can produce feature maps from raw data, heartbeat segmentation can be simultaneously conducted with classification with a single neural network. Second, beat-wise classification uses isolated heartbeats, which risks losing inter-heartbeat context information that can be useful to boosting classification performance.

Addressing these drawbacks, we propose a novel deep learning model that can simultaneously conduct heartbeat segmentation and classification. Compared to existing methods, our model is more compact as it does not require explicit heartbeat segmentation. The difference between our model and existing deep learning models is shown in Fig. 1. As is shown, our model takes in a 1-D ECG sequence and outputs both the segmented heartbeats and their corresponding labels. Besides, our model is more context-aware, for it takes into account the relationship between heartbeats.

To achieve simultaneous segmentation and classification, we present a Faster R-CNN [2] based model that has been customized to handle ECG sequences. To capture inter-heartbeat context information, we exploit inverted residual blocks [3] to produce multi-scale feature maps, which are then fused by a novel feature

fusion mechanism to learn inter-heartbeat context information. Moreover, the semantic information and morphological information are explored from the fused features to improve performance.

Our main contributions are as follows:

- We propose a novel deep learning model for simultaneous heartbeat segmentation and classification.
- We present a novel Faster R-CNN based model that has been customized to handle ECG data.
- We use inverted residual blocks and a novel feature fusion subroutine to exploit long term inter-heartbeat dependencies for context awareness.
- We conduct extensive experiments on the well-known MIT-BIH database [4, 5] to demonstrate the effectiveness of our model.

The rest of this paper is organized as follows. Section 2 reviews the related work. Section 3 presents our model. Section 4 reports the experimental results. Section 5 concludes this paper.

## 2 Related Work

Traditional arrhythmia detection methods extract handcrafted features from ECG data, such as R-R intervals [6, 7], ECG morphology [6], frequency [8], etc. Classifiers such as Linear Discriminant Analysis models [6], Support Vector Machines [7] and Random Forests [9] are then built upon these features.

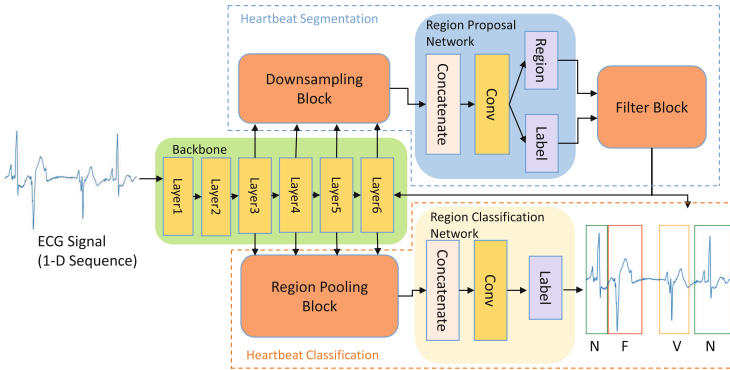
In recent years, many researchers turn to deep neural networks for heartbeat classification. The majority of deep learning models take raw signals as their input, omitting explicit feature extraction and selection steps. In [10], Kiranyaz et al. proposed a patient-specific 1-D CNN for ECG classification. In [11], Yildirim et al. designed a deep LSTM network with wavelet-based layers for heartbeat classification. Some methods [12, 13] combine LSTM with CNN.

These aforementioned deep learning methods work in three steps: preprocessing, heartbeat segmentation and beat-wise classification. They do not explicitly utilize context information among heartbeats. By contrast, Mousavi et al. [14] proposed a sequence-to-sequence LSTM based model which maps a sequence of heartbeats in time order to a sequence of labels, where context dependencies are captured by the cell state of the network. Hannun et al. [15] proposed a 33-layer neural network for arrhythmia detection which maps ECG data to a sequence of labels. However, the precise regions of arrhythmias cannot be obtained. Oh et al. [16] used a modified U-net to identify regions of heartbeats and the background from raw signals, yet this method needs extra steps to detect arrhythmias from the generated annotation. Several researches applies Faster R-CNN to ECG analysis. For example, Ji et al. [17] proposed a heartbeat classification framework based on Faster R-CNN. 1-D heartbeats extracted from original signals are converted to images as the input of the model. Sophisticated preprocessing is required before classification. He et al. [18] and Yu et al. [19] use Faster R-CNN to perform heartbeat segmentation and QRS complex detection. In our method,

we present a modified Faster R-CNN for arrhythmia detection which works in only two steps: preprocessing, and simultaneous heartbeat segmentation and classification.

### 3 The Proposed Model

The architecture of our model is shown in Fig. 2, which takes 1-D ECG sequence as its input and conducts heartbeat segmentation and classification simultaneously. To achieve this, our model consists of 6 modules: a backbone network, a region proposal network (RPN), a region classification network (RCN), a filter block, a down sampling block and a region pooling block.



**Fig. 2.** The architecture of our model

The backbone network produces multi-scale feature maps from the ECG signal. The modules in the upper part of Fig. 2 performs heartbeat segmentation, while ones in the lower part performs heartbeat classification. We now elaborate on the details of our method.

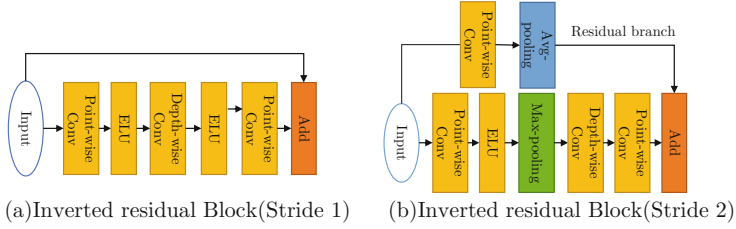
#### 3.1 Preprocessing

The preprocessing step removes noise from raw signals. Here we employ a three-order Butterworth bandpass filter with a frequency range of 0.27 Hz–45 Hz because this range contains the main components of ECG signals [20].

#### 3.2 Backbone Network and Down Sampling Block

The backbone network generates multi-scale semantic and morphological feature maps from raw ECG signals. For efficiency, we choose the inverted residual block [3] as the building block. We customize it for ECG data in the following manner: (1) We increase the kernel size from 3 to 5 to enlarge the receptive field. (2) The

activation function is replaced by ELU for less information loss. (3) The residual connection is added to the building block in stride 2 condition. There are two branches in stride 2 condition (Fig. 3). (4) Stride 2 convolution is replaced by max-pooling to make the model more lightweight. There are 6 layers in our backbone. Each layer is composed of several building blocks (Fig. 3). Besides, each layer downsamples the feature map by a factor of two.



**Fig. 3.** Inverted residual blocks in (a) stride 1 condition and (b) stride 2 condition

Different from most deep learning methods which compute feature maps for a single heartbeat, our backbone model takes a long ECG sequence as its input. The produced feature maps encode not only morphological and semantic information of individual heartbeats, but also context information amongst multiple heartbeats. The bottom layers of the backbone generate feature maps of strong morphological information while the top layers produce feature maps of strong semantic information [21]. Moreover, the receptive field increases from bottom layers to top layers, thus the feature maps encode involve inter-heartbeat context dependencies with varying differences in time.

We fuse multi-scale feature maps to utilize both morphological and semantic information of the heartbeats in segmentation and classification. Besides, the fused feature maps can provide more context information. All feature maps except those in the first two layers are used for efficiency.

In the segmentation task, the feature maps need to be normalized to have equal dimensions by the downsampling block before fusion. Feature maps are downsampled to a fixed length by a novel mechanism shown in Eq. 1, which is a trade-off between performance and complexity. Contrary to convolution based down sampling methods [21], our down sampling block is parameter-free. Average pooling is exploited for less information loss during downsampling, while max-pooling highlights the discriminative features in the feature maps.

$$x = \begin{cases} x & \text{if } \text{len}(x) = L; \\ \text{Max} - \text{Pool}(x) + \text{Avg} - \text{Pool}(x) & \text{else.} \end{cases} \quad (1)$$

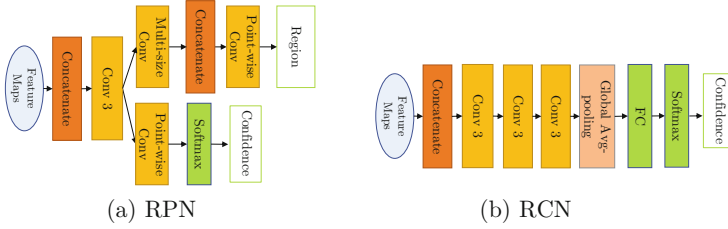
### 3.3 Region Proposal Network (RPN) and Filter Block

The RPN fuses the feature maps from the downsampling block and performs heartbeat segmentation. Here we directly segment the heartbeats without QRS

detection. As is shown in Fig. 4 (a), RPN has two branches performing regression (top) and classification (bottom). The classification branch produces a binary label for each region indicating whether it contains a single heartbeat. The regression branch produces endpoints of each region which encloses a heartbeat. Intuitively, the regression task is far more difficult than binary classification for RPN, thus we use multi-size convolutional layers (3, 5, 7 in this paper) to further extract features before regression.

Following the practice of Faster R-CNN [2], at each position of a feature map, we pre-define three reference regions. These regions have different sizes (128, 192, 256 for ECG heartbeats). To predict a region, we use the center of one of the three regions as the reference point and report its offset to this reference.

However, the regions obtained by RPN can overlap with nearby regions, undermining the efficiency of the model. In response, we use non-maximum suppression (NMS) to filter these regions in the filter block. NMS selects a region with max confidence in each iteration and then compute the overlaps between each remaining region and the selected one. The regions whose overlap exceeds a pre-set threshold (30% in this paper) are discarded, so are the regions containing no heartbeats (confidence below 0.5).



**Fig. 4.** The architecture of (a) RPN and (b) RCN

### 3.4 Region Pooling Block and Region Classification Network

In the heartbeat classification task, the region pooling block generates heartbeat feature maps for the predicted heartbeat regions [2]. Feature maps in the last four layers (with strides 8, 16, 32, 64) are reused to extract heartbeat features. Because these feature maps have different sizes, the predicted regions are mapped as:  $Region = (start/stride, end/stride)$ . Moreover, each region is divided into fixed-size sub-regions. Heartbeat feature maps are produced by average pooling on each sub-region. To keep sufficient morphological information, heartbeat feature maps in the bottom layers have larger sizes (8, 4, 2, 1 for strides 8, 16, 32, 64).

Heartbeat feature maps are then fed into the region classification network (RCN, Fig. 4 (b)) to classify heartbeats inside each region. RCN performs heartbeat classification by fusing the feature maps from the region pooling block.

Note that we do not fine-tune the regions as in Faster R-CNN because it trades efficiency for only minor improvements in accuracy.

### 3.5 Implementation Details

Following common practices in the detection task [2, 21], our backbone network is initialized with a pre-trained network. We extract heartbeats from the experimental database (to be discussed later) and pre-train the backbone network with extra layers on these heartbeats. Then, the last few layers are removed while the remain ones are used as the backbone network.

We coarsely annotate the groundtruth heartbeat region for each heartbeat which ranges from 0.25 s–0.83 s around the R peak so as to most of heartbeat. Since our model can capture inter-heartbeat context information, finer annotation is not necessary. The offsets of reference regions to the groundtruth ones [2] are used to train the RPN regression branch. To train the classification branch, positive labels are assigned a predicted region when the following criteria are met: (1) Its overlap with a groundtruth heartbeat region is over 0.7. (2) It has the highest overlap with a groundtruth heartbeat region. In RCN training, the label of a predict region is assigned to the heartbeat inside it. We use Jaccard distance as the metric for overlap computation.

Similar to [2], our entire training process has two steps: 1) train the RPN with regression loss and binary classification loss. 2) train the RCN with multi-classification loss. For better performance, we choose Smooth L1 loss (Eq. 2) to train the regression branch and Focal loss (Eq. 3) to train RCN and RPN for classification.

$$\text{Smooth } L1(x) = \begin{cases} 0.5x^2 & |x| \leq 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \quad (2)$$

$$\text{Focal Loss}(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t) \quad (3)$$

where  $p_t$  is the estimated probability for the class  $t$ . We set  $\gamma$  to 2, and  $\alpha_t$  to [0.25, 1] for binary classification and [1, 0.5, 1, 0.5, 1] for multi-classification.

## 4 Experiments

### 4.1 Experimental Settings

We implemented our model using PyTorch<sup>1</sup>. Our source code is available<sup>2</sup> for reproducibility. The experiments were run on a Linux server with an Intel Xeon W-2145 CPU @3.7 GHz, 64 GB memory and a single NVIDIA 1080Ti GPU. Adadelata was used as the optimizer with weight decay  $1e^{-5}$ . The learning rate was set to 0.15 for training, decaying every 10 epochs exponentially as follows:  $lr = 0.3 * 0.9^{epoch/50}$ . The batch size was set to 240 for both training and testing.

<sup>1</sup> <https://pytorch.org/>.

<sup>2</sup> [https://github.com/QiXi9409/Simultaneous\\_ECG\\_Heartbeat](https://github.com/QiXi9409/Simultaneous_ECG_Heartbeat).

**Table 1.** Dataset information

Classes	Number of heartbeats
Beats other than S, V, F and Q (N)	82472
Supraventricular ectopic beats (S)	2527
Ventricular ectopic beats (V)	6439
Fusion of ventricular and normal beats (F)	754
Paced beats or beats that cannot be classified (Q)	14
Total	92206

We used data from the well-known MIT-BIH database [4,5], which contains 48 half-hour two-lead ECG recordings. We used the MLII lead in the experiments and excluded 4 recordings with paced beats, following the ANSI/AAMI EC57 standard [22]. Due to limited computational resources, we divided each recording into a series of long sequences with 3600 data points. The first and last 10 s of each recording were discarded. Note that our model can process much longer ECG recordings with abundant computational resources. We run the experiments 5 times, randomly dividing the dataset into training, validation and test sets for each run. The training set contained 70% of all data. The evaluation and test sets included 10% and 20% of all data. The heartbeat labels were mapped into 5 groups by ANSI/AAMI standard, namely N, S, V, F, Q (see Table 1). We did not take the Q class into consideration because of its scarcity. We applied the following metrics for evaluation: positive predictive value (PPV), sensitivity (SEN), specificity (SPE) and accuracy (ACC).

#### 4.2 Performance of ECG Heartbeat Segmentation

To evaluate heartbeat segmentation performance, we define truth positive (TP) as: 1) A predicted region contains only one heartbeat and 2) its non-overlapping area with the groundtruth is less than 150 ms. We define false positive (FP) as: 1) A predict region encloses more than two heartbeats or 2) its non-overlapping area with the groundtruth is exceeds 150 ms. We define false negative (FN) as: A groundtruth heartbeat is not enclosed by any predicted region.

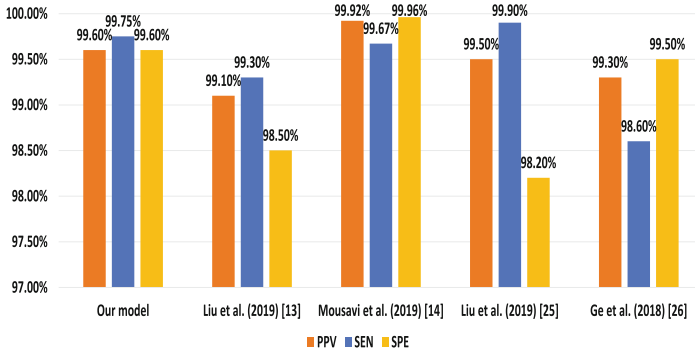
For baselines, we used two QRS detection based heartbeat segmentation methods: Pan-Tompkins [23] and Wavedet [24]. The results are shown in Table 2. As is shown, our method is highly competitive against the baselines. It is worth noting that unlike the baselines, our model does not apply QRS detection for segmentation, thus there may be inconsistencies on the definitions of TP, FP and FN between our model and the baselines. However, it is safe to say that our model performs well enough to be applied in real-world scenarios.



**Table 2.** Heartbeat segmentation performance

Method	PPV (%)	SEN (%)	ACC (%)
Our model	99.63	99.89	99.52
Pan-Tompkins [23]	99.56	99.76	99.32
Wavedet [24]	99.86	99.80	99.66

### 4.3 Performance of ECG Heartbeat Classification


**Fig. 5.** Heartbeat classification performance

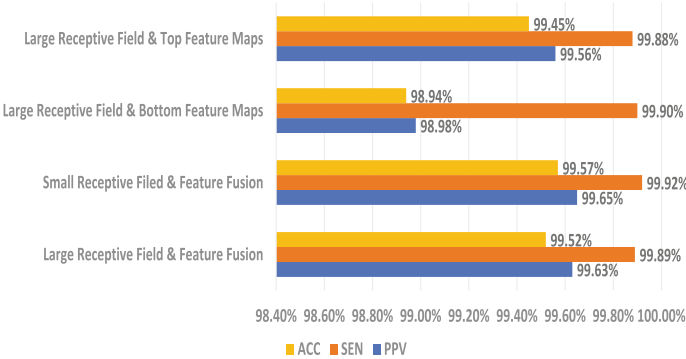
We now evaluate the heartbeat classification performance. The baselines come from [13, 14, 25, 26]. Here we applied SMOTE [27] for data augmentation as it was also used in our baselines. Figure 5 shows the results. Our model achieves an accuracy of 99.6%, a sensitivity of 99.75% and a specificity of 99.6%. These results are similar to those obtained by [14] which used a LSTM-based sequence-to-sequence model to learn context information. The difference between our work and [14] is that we learn context information from raw signals while [14] did so using a sequence of individual heartbeats. Besides, the LSTM-based model has lower efficiency. Compared to other baselines which perform classification on individual heartbeat, our model has a simpler model structure but achieves similar or better results on some metrics, highlighting the power of context-awareness.

### 4.4 Ablation Study

We now investigate the impact of the key design features of our model. For better evaluation, we did not use SMOTE [27] here.

**Long Term Context Information and Feature Fusion.** To better capture long term dependencies, we have enlarged the receptive field to retain context information. Also, our model captures inter-heartbeat dependencies by learning

multi-scale feature maps with RPN and RCN. To demonstrate the effectiveness of these design choices, we conducted the following ablation tests: 1) setting the kernel size to 3 for all convolution filters, 2) using the feature maps in the top layers only, 3) using the feature maps in the bottom layers only, 4) equalize the output sizes of multi-scale feature maps in the region pooling block. Figure 6 and Table 3 presents the results in the segmentation and classification tasks. As is shown, while the effectiveness of these design features in the segmentation task is limited, they are indeed beneficial to the classification task.



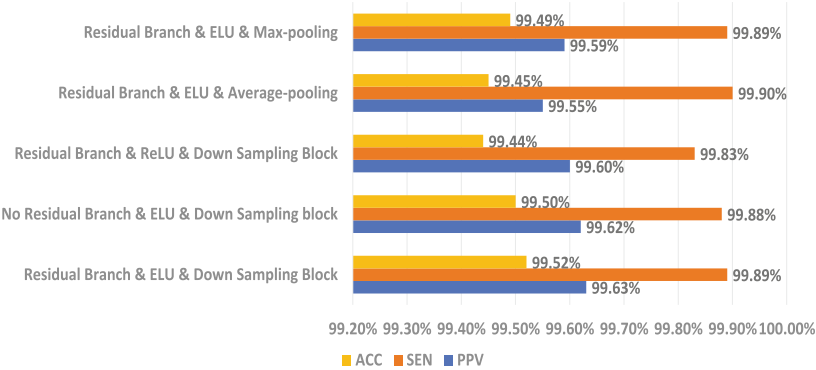
**Fig. 6.** Impact of long term context information and feature fusion on segmentation

**Table 3.** Impact of long term context information and feature fusion on classification

Receptive field	Bottom feature	Top feature	Region pooling size	PPV (%)	SPE (%)
Large	✓	✓	(8, 4, 2, 1)	98.11	99.53
Small	✓	✓	(8, 4, 2, 1)	97.91	99.48
Large	✓		(8, 4, 2, 1)	97.84	99.46
Large		✓	(8, 4, 2, 1)	97.70	99.43
Large	✓	✓	(4, 4, 4, 4)	97.80	99.47

**Inverted Residual Block and Downsampling Block.** In our model, we have modified the inverted residual block and designed a novel downsampling mechanism. To demonstrate their effectiveness, we conducted the following ablation tests: 1) removing the residual branch. 2) changing the activation function to ReLU. 3) only using average pooling in the downsampling block. 4) only using max-pooling in the downsampling block. Figure 7 and Table 4 show the results. By modifying the backbone, our model can learn more strong features to improve performance. Our downsampling method also outperforms using only max-pooling or average pooling.

**Architecture of RPN.** Our model has more parameters in the regression branch of RPN based on the intuition that regression is more difficult than



**Fig. 7.** Influence of backbone structure and downsampling block on segmentation

**Table 4.** Influence of inverted residual block structure on classification

Residual branch	Activation function	PPV (%)	SPE (%)
✓	ELU	98.11	99.53
✓	ReLU	97.89	99.47
	ELU	98.05	99.51

binary classification in RPN. To evaluate this design, we conducted the following ablation tests: 1) enlarging the classification branch. 2) simplifying the regression branch. Table 5 presents the results. As is shown, simplifying the regression branch has negative impact on performance, while enlarging the classification branch brings about no improvement.

**Table 5.** Influence of RPN structure on segmentation

Larger classification branch	Larger regression branch	PPV (%)	SPE (%)	ACC (%)
	✓	99.63	99.89	99.52
✓	✓	99.29	99.86	99.16
		99.61	99.89	99.50

5 Conclusions and Future Work

In this paper, we have propose a novel deep learning model that can simultaneously conduct heartbeat segmentation and classification. Compared to existing methods, our model is more compact as it does not require explicit heartbeat segmentation. Moreover, our model is context-aware by using feature fusion and long term context dependencies. In the future, we plan to extend our model to multi-lead ECG analysis tasks.

**Acknowledgement.** This work is funded by NSFC Grant 61672161 and Dongguan Innovative Research Team Program 2018607201008. We sincerely thank Prof Chun Liang and Dr Zhiqing He from Department of Cardiology, Shanghai Changzheng hospital for their valuable advice.

## References

1. Podrid, P.J., Kowey, P.R.: Cardiac Arrhythmia: Mechanisms, Diagnosis, and Management. Lippincott Williams & Wilkins, Philadelphia (2001)
2. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
3. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNetV2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520 (2018)
4. Moody, G.B., Mark, R.G.: The impact of the MIT-BIH arrhythmia database. IEEE Eng. Med. Biol. Mag. **20**(3), 45–50 (2001)
5. Goldberger, A.L., et al.: Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. Circulation **101**(23), e215–e220 (2000)
6. De Chazal, P., O'Dwyer, M., Reilly, R.B.: Automatic classification of heartbeats using ECG morphology and heartbeat interval features. IEEE Trans. Biomed. Eng. **51**(7), 1196–1206 (2004)
7. Melgani, F., Bazi, Y.: Classification of electrocardiogram signals with support vector machines and particle swarm optimization. IEEE Trans. Inf. Technol. Biomed. **12**(5), 667–677 (2008)
8. Martis, R.J., Acharya, U.R., Min, L.C.: ECG beat classification using PCA, LDA, ICA and discrete wavelet transform. Biomed. Signal Process. Control **8**(5), 437–448 (2013)
9. Alickovic, E., Subasi, A.: Medical decision support system for diagnosis of heart arrhythmia using DWT and random forests classifier. J. Med. Syst. **40**(4), 1–12 (2016). <https://doi.org/10.1007/s10916-016-0467-8>
10. Kiranyaz, S., Ince, T., Gabbouj, M.: Real-time patient-specific ECG classification by 1-D convolutional neural networks. IEEE Trans. Biomed. Eng. **63**(3), 664–675 (2015)
11. Yildirim, Ö.: A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification. Comput. Biol. Med. **96**, 189–202 (2018)
12. Warrick, P., Homsy, M.N.: Cardiac arrhythmia detection from ECG combining convolutional and long short-term memory networks. In: Computing in Cardiology (CinC), pp. 1–4. IEEE (2017)
13. Liu, F., Zhou, X., Cao, J., Wang, Z., Wang, H., Zhang, Y.: A LSTM and CNN based assemble neural network framework for arrhythmias classification. In: ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1303–1307. IEEE (2019)
14. Mousavi, S., Afghah, F.: Inter-and intra-patient ECG heartbeat classification for arrhythmia detection: a sequence to sequence deep learning approach. In: ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1308–1312. IEEE (2019)

15. Hannun, A.Y., et al.: Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nat. Med.* **25**(1), 65 (2019)
16. Oh, S.L., Ng, E.Y., San Tan, R., Acharya, U.R.: Automated beat-wise arrhythmia diagnosis using modified U-net on extended electrocardiographic recordings with heterogeneous arrhythmia types. *Comput. Biol. Med.* **105**, 92–101 (2019)
17. Ji, Y., Zhang, S., Xiao, W.: Electrocardiogram classification based on faster regions with convolutional neural network. *Sensors* **19**(11), 2558 (2019)
18. He, Z., Niu, J., Ren, J., Shi, Y., Zhang, W.: A deep learning method for heartbeat detection in ECG image. In: Deng, Z. (ed.) *CIAAC 2019. LNEE*, vol. 586, pp. 356–363. Springer, Singapore (2020). [https://doi.org/10.1007/978-981-32-9050-1\\_41](https://doi.org/10.1007/978-981-32-9050-1_41)
19. Yu, R., Gao, Y., Duan, X., Zhu, T., Wang, Z., Jiao, B.: QRS detection and measurement method of ECG paper based on convolutional neural networks. In: 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 4636–4639. IEEE (2018)
20. Tereshchenko, L.G., Josephson, M.E.: Frequency content and characteristics of ventricular conduction. *J. Electrocardiol.* **48**(6), 933–937 (2015)
21. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125 (2017)
22. Association for the Advancement of Medical Instrumentation. ANSI/AAMI EC57: 2012–Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms. American National Standard (2013)
23. Pan, J., Tompkins, W.J.: A real-time QRS detection algorithm. *IEEE Trans. Biomed. Eng.* **32**(3), 230–236 (1985)
24. Martínez, J.P., Almeida, R., Olmos, S., Rocha, A.P., Laguna, P.: A wavelet-based ECG delineator: evaluation on standard databases. *IEEE Trans. Biomed. Eng.* **51**(4), 570–581 (2004)
25. Liu, F., Zhou, X., Cao, J., Wang, Z., Wang, H., Zhang, Y.: Arrhythmias classification by integrating stacked bidirectional LSTM and two-dimensional CNN. In: Yang, Q., Zhou, Z.-H., Gong, Z., Zhang, M.-L., Huang, S.-J. (eds.) *PAKDD 2019. LNCS (LNAI)*, vol. 11440, pp. 136–149. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-16145-3\\_11](https://doi.org/10.1007/978-3-030-16145-3_11)
26. Ge, H., Sun, K., Sun, L., Zhao, M., Wu, C.: A selective ensemble learning framework for ECG-based heartbeat classification with imbalanced data. In: *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 2753–2755. IEEE (2018)
27. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002)