RMKnet: A novel and parameter-efficient architecture for robust r-peak detection

Qi Lu
College of Computer science, Sichuan University
Department, University, Chengdu, Sichuan, 610065, China
ramble_on@stu.scu.edu.cn

Abstract. The Electrocardiogram (ECG) has become the most important diagnostic criteria in the non-invasive diagnosis of heart disease. The automatic detection of r-peaks in an electrocardiogram signal is also crucial in all kinds of applications about heart disease. Recently, lots of algorithms have been proposed to automatically detect r-peaks in ECG signals. Traditional detection methods have high detection accuracy on high quality ECG signals which don't have so much noise, but they perform poorly on low quality ECG signal. Most of the deep learning methods are more robust and have higher accuracy than the traditional methods, but they are usually computationally expensive and difficult to implement and customize. In this paper, we proposed a novel and robust model architecture based on Unet[17] to detect r-peaks. After training, we use re-parameterization greatly accelerating the inference process and reducing the inference memory utilization. The proposed network is evaluated on two open-access ECG databases: CPSC-DB, MIT-BIH and achieves high F1 scores which showing good generalization on different databases comparing to all competing methods.

1. Introduction
The Electrocardiogram (ECG) has become the most important diagnostic criteria in the non-invasive diagnosis of heart disease. The accurate detection of r-peak is the most prominent feature in ECG.

In order to process gigantic amount ECGs and reduce doctors’ workload, a lot of algorithms have been proposed throughout the last several decades to automatically detect R-peak. Some of them include digital filtering[1], wavelet transform, Hilbert transform, and machine learning.

Most of the mentioned popular methods showed high detection accuracy on high quality ECG signals which don't have so much noise, but they perform poorly on low quality ECG signal. In real clinical environment, strong noise is unavoidable due to arrhythmias and motion artifacts during long-term ECG monitoring. It is still a challenging problem that detecting the r-peak more accurately and robustly.

Recently deep learning (DL) have been successful used in CV and NLP. Some neural network has been demonstrated in r-peak detection such as one dimension residual neural networks and LSTM network and U-Net combined with blocks based on Inception net and Residual net CNN model[2]. However most of the deep learning models mentioned are difficult to implement and customize. And they take up a lot of memory and the inference speed is slow. When the testing data is from different source with the training data, they usually perform not so well.

Our contributions in this study are summarized as follows.
We propose a novel and robust deep learning approach that achieves good performance in complex r-peak detection and has been well generalized across different ECG databases.
We use re-param methods on the proposed structure after training greatly reduces the amount of parameters and speeds up the inference process.

2. Databases

2.1. CPSC
The 2019CPSCDB database (China Physiological Signal Challenge)[8] is made up of 2,000 single-lead electrocardiogram recordings collected from multiple sources of patients with cardiovascular disease. Electrocardiogram recordings were obtained from multiple sources using a variety of instrumentation. It contains electrocardiogram summaries which have much noise with various arrhythmia patterns. All recordings are sampled at 500 Hz and each recording is 10 s long.

2.2. MIT-BIH
The MIT-BIH is made up of 48 double-lead ECGs from 47 subjects sampled at 360 Hz[9]. The recording in database covers 30 minutes and are collected in a clinical setting environment which are relatively clear and of high quality. This dataset has been widely used as a baseline and is currently the most popular database for evaluating r-peak detection algorithms in clinical electrocardiography.

3. Methodology

3.1. Problem Formulation
We consider r-peaks detection as a one-dimensional segmentation problem to segment r-peaks. The proposed model transforms and maps the input electrocardiogram to an output signal which has the same length with the input and is centered on the r-peak location. For input ECG signals from different datasets, all recordings are resampled to 500 Hz. To achieve better model generalization, the input electrocardiogram signals \( x \) was standardized and label \( y \) was obtained from general r-peak locations as a 1D segmentation map which lies within 75ms duration of the reference location.

3.2. Model Architecture
The proposed model’s architecture is adapted from the U-Net network which is commonly used in 2D medical image segmentation. The architecture of the model RMKnet(Re-para-Multi-size-Kernel-UNet) is shown in Fig. 1. The RMKnet, takes the electrocardiogram record of 10s, 500Hz and a total of 5000 recording points as input. The input electrocardiogram signal is downsampled through 8 layers of
convolutions with a downsampling factor of 2. The number of filters starts from 16 and doubles every two layers until it reaches 256, after which it is kept constant. There is a relu activation layer after each convolution. Then the compressed feature vector is upsampled through nearest padding by a factor of two and concatenated with the correspondingly feature map.

There is a RMK block after each 1d convolution in each layer. The architecture of RMK block is shown in Fig 2(a). In the RMK block, there are four parallel dilated convolutional blocks following the input layer, each convolutional block contains an identity branch and different conv branches with kernel-size of [9, 15, 21] which are all followed by a batch Normalization layer. Then the addition features extracted from the multi-branch convolutional blocks are input into the se module which uses attention mechanisms and are followed by three fully connected layers.

![Diagram of RMK block](image)

**Fig. 2:** Structural re-parameterization of a RMK block.

### 3.3. Re-param for Model
In the RMK block, we use three convolutional branches and one identity branch, the identity can be viewed as a 1 × 1 conv with an identity matrix as the kernel. Note that we use BN in each branch before the addition, so we convert every BN and its preceding conv layer into a conv with a bias vector. The derivation is as follows, substituting the convolution computation into BN computation, and we use $W_{fused}$ replaces $\frac{r \times w}{\sqrt{var}}$, $B_{fused}$ replaces $\frac{r \times (b - mean)}{\sqrt{var}} + \beta$ to get the final weights and bias.

$$Conv(x) = W(x) + b$$
$$BN(x) = \gamma \times \frac{(x - mean)}{\sqrt{var}} + \beta$$
$$BN(Conv(x)) = \frac{\gamma \times W(x)}{\sqrt{var}} + (\gamma \times \frac{(b - mean)}{\sqrt{var}} + \beta)$$

After such transformations, we will have one 1 × 21 kernel, one 1 × 15 kernel, one 1 × 9 kernel, and three bias vectors. Then we obtain the final bias by adding up the three bias vectors, and the final 1 × 21 kernel by adding the 1×9 kernel, 1×15, and 1×1 kernel (represent the identity branch) onto the central point of 1 × 21 kernel, which can be easily implemented by zero-padding. The whole process is shown in Fig 2. After reparameterization, small kernel branch can be merged to the maximum kernel branch. The amount of parameters, the time and space of inference process only depend on the maximum size kernel branch, which is greatly optimized compared to the training phase.

### 4. Experiments and results

#### 4.1. Implementation Details
We use pytorch to implement the proposed model is on an Nvidia GTX TITAN X 12G GPU. The model’s parameters were randomly initialized before training. And we use Adam optimizer to
optimize the cross entropy loss between the output predictions and the ground truth. We trained the network for 250 epochs with a batch size of 64. The inference process is all on an Intel Core i5-4590 CPU.

4.2. Performance evaluation
The quantitative performance of different models and methods was compared using three commonly used performance indicators: sensitivity (Se), positive predictive value (PPV), error rate (ER) and F1, which were measured within ±75 ms of the true peak position. These metrics are defined as follows:

\[
\text{Se} (\%) = \frac{TP}{TP + FN} \times 100
\]

\[
\text{PPV} (\%) = \frac{TP}{TP + FP} \times 100
\]

\[
\text{Er} (\%) = \frac{FP + FN}{TP + FP + FN} \times 100
\]

\[
F1 (\%) = \frac{2 \times \text{Se} \times \text{PPV}}{\text{Se} + \text{PPV}}
\]

4.3. Results on CPSC-DB
Table 1 shows the r-peak detection performance on CPSC-DB when we choose different kernel sizes and branch amounts as basic RMK block. From the results, we can find that unlike traditional CV tasks using conv kernel with size 1 and 3, single branch with small kernel size performances not so good. However, the effect does not always increase along with the single branch kernel size. In fact, when we use multi-branch conv kernels of different sizes, the effect is better than the single branch in most situations, which means multi-branch conv kernels of different sizes can extract and fuse multi-scale ECG signal features and more helpful for the r-peak detection task. Based on the results in Table 1, we choose three conv branches with kernel-size of [9, 15, 21] as RMK block. Table 2 shows the performance of proposed methods and other competing methods. It can be seen from the results that the proposed model achieves the best performance on CPSC-DB with the highest F1 score of 99.11%, which means that the proposed method can indeed achieve better learning ability to detect the true peak of this low-quality ECG signal. In Table 3, we can find that benefitting from the architecture of the proposed model, after reparameterization in inference phase, the parameters of model, inference running time and theoretical FLOPs are reduced by more than 54% and 61%, and 52% respectively.

4.4. Results on MIT-DB
The performance of proposed methods (‘cross-database’ means training only on the CPSC-DB and testing on the MIT-DB, ‘fine-tune’ means training on the CPSC-DB and finetuning on the MIT-DB) and other competing methods(training only on the MIT-DB) is shown in Table 4. Cross database testing showed the RMK model has a superior robustness to detect the actual peaks and has higher F1 score than most methods. For all four types of metrics, the fine-tuned RMK model is superior to other published approaches and got the best performance with ER of 0.12%, PPV of 99.94%, F1 of 99.93%, and Se of 99.93%.

| Architecture | Se (%) | PPV (%) | Er (%) | F1 (%) |
|--------------|--------|---------|--------|--------|
| Single-1     | 98.22  | 69.44   | 31.41  | 81.36  |
| Single-3     | 97.67  | 96.60   | 5.56   | 97.10  |
| Single-9     | 98.20  | 98.24   | 3.48   | 98.22  |
| Single-15    | 98.53  | 98.54   | 2.87   | 98.53  |
| Single-21    | 98.56  | 98.63   | 2.76   | 98.59  |
Table 2. Peak detection performances on CPSC-DB

| Methods       | Se  | PPV | Er  | F1  |
|---------------|-----|-----|-----|-----|
| RPNet[2]      | 98.62 | 98.12 | 3.20 | 98.37 |
| Christov[3]   | 71.35 | 90.85 | 34.19 | 79.93 |
| Hamilton[4]   | 77.56 | 86.21 | 31.57 | 81.66 |
| SWT[5]        | 77.91 | 87.09 | 30.94 | 82.24 |
| RMK           | 99.08 | 99.15 | 1.82 | 99.11 |

Table 3. Inference process on CPSC-DB

| Parameters | Inference time(ms) | Theo FLOPs(B) |
|------------|--------------------|---------------|
| Before re-para | 6.1187 | 136.200 | 9.923617 |
| After re-para  | 2.8530 | 52.548 | 4.685857 |

Table 4. Peak detection performances on MIT-DB

| Methods                  | Se  | PPV | Er  | F1  |
|--------------------------|-----|-----|-----|-----|
| RMK cross-database       | 99.75 | 99.87 | 0.37 | 99.81 |
| RMK fine-tune            | 99.93 | 99.94 | 0.12 | 99.93 |
| PT[1]                    | 99.19 | 99.41 | 1.39 | 99.30 |
| RPNet[2]                 | 99.44 | 99.75 | 0.80 | 99.65 |
| Hamilton[4]              | 99.69 | 99.77 | 0.53 | 99.73 |
| SWT[5]                   | 99.88 | 99.84 | 0.31 | 99.86 |

Figure 3 shows the actual location of some typical r-peaks from different patients and the results of the proposed model. In this figure, the blue lines represent the ECG signal, the yellow bars and red dotted lines represent the actual ground position of r-peak and the peaks detected by RMKNet. We can find that it is hard for a normal human without professional knowledge to detect some of the R-peaks in the low quality ECG signals, but the proposed approach can detect almost all r-peaks accurately. And we also can find that when some signal points are highly similar to the r-peaks or the r-peak locates at the edge of signal, the model performs not so good which may due to the lack of pre or post features, there could be room for improvement.
Fig. 3: Ground truth location of some typical R-peaks and the results of the proposed model

5. Conclusion
In this paper, we proposed the RMKnet which is adapted from Unet with multi-branch and multi-size kernels for detecting QRS complex and r-peaks and achieves high performance and generalizes well on ECG datasets. To make sure it suitable for the real time usage, we merge small kernel branches to the maximum kernel branch through reparameterization which greatly accelerating the inference process and reducing the inference memory utilization. The proposed model shows a powerful potential of artificial intelligence in ECG analysis.

References
[1] J. Pan and W. J. Tompkins, “A real-time QRS detection algorithm,” IEEE Trans. Biomed. Eng., vol. BME-32, no. 3, pp. 230–236, Mar. 1985.
[2] S. Vijayarangan, R. Vignesh, B. Murugesan, P. Sp, J. Joseph, and M. Sivaprakasam, “RPnet: A Deep Learning approach for robust R Peak detection in noisy ECG,” Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS, vol. 2020-July, no. 1, pp. 345–348, 2020, doi: 10.1109/EMBC44109.2020.9176084
[3] I. I. Christov, “Real time electrocardiogram qrs detection using combined adaptive threshold,” Biomedical engineering online, vol. 3, no. 1, p. 28, 2004.
[4] P. Hamilton, “Open source ecg analysis,” in Computers in cardiology. IEEE, 2002, pp. 101–104.
[5] V. Kalidas, L. T.-2017 I. 17th I. Conference, and undefined 2017, “Real-time QRS detector using stationary wavelet transform for automated ECG analysis,” ieeexplore.ieee.org, Accessed: Dec. 10, 2020.
[6] Giannakakis G, Trivizakis E, Tsiknakis M, et al. A novel multi-kernel 1D convolutional neural network for stress recognition from ECG[C]//2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW). IEEE, 2019: 1-4.
[7] Ding X, Zhang X, Ma N, et al. RepVGG: Making VGG-style ConvNets Great Again[J]. arXiv preprint arXiv:2101.03697, 2021.
[8] H. X. Gao, C. Y. Liu*, X. Y. Wang, L. N. Zhao, Q. Shen, E. Y. K. Ng, and J. Q. Li. An Open-Access ECG Database for Algorithm Evaluation of QRS Detection and Heart Rate Estimation. Journal of Medical Imaging and Health Informatics, 2019, 9(9): 1853–1858.

[9] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation [Online]. 101 (23), pp. e215–e220.