A Deep Learning Method for Heartbeat Detection in ECG Image

Zewen He[1,2], Jinghao Niu[1,2], Junhong Ren[1], Yajun Shi[3], and Wensheng Zhang[1,2(✉)]

[1] Institute of Automation, Chinese Academy of Sciences, 95 Zhongguancun East Road, Beijing 100190, China
{hezewen2014, niujinghao2015, junhong.ren}@ia.ac.cn, zhangwenshengia@hotmail.com
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Department of Cardiology, Chinese PLA General Hospital, 28 Fuxing Road, Beijing 100853, China
shiyajun301@163.com

Abstract. Although heartbeat segmentation can be done very well in ECG signals for arrhythmia detecting, there're short of techniques for detecting heartbeat part from ECG images. We apply the powerful Faster R-CNN detector here, and achieves accurate detecting results. Along with the improved patch-sampling mechanism in training, detection results are more precise. The high evaluation metric on validation data and demo of real scenes demonstrate the effectiveness of our method.

Keywords: Heartbeat detection · ECG images · Faster R-CNN detector

1 Introduction

ECG is a method to transform electric wave of heart to digital signals or images. Advanced techniques [12] are developed to analyze the ECGs. For example, diseases about heart like arrhythmia can be diagnosed quickly from ECG signals. Constrained by interface incompatibility, not all ECG signal data can be transferred between different hospitals. Fortunately ECG images can be shared by smartphones easily. However there lacks detection methods on ECG images. We haven't seen related works about detecting heartbeat part from ECG images.

In computer vision community, it has make huge progress on object detection, originating from the convolutional neural network (CNN). Powerful backbones [10], delicate design on loss and well-annotated datasets contribute to this success.

Inspired by it, we try to prepare ECG images with labelled bounding-box for heartbeat detecting. Next we apply the Faster R-CNN detector to ECG images. Considering the scale variation of the heartbeat parts in ECG images, we propose and use customized patch-sampling mechanism in training to promote performance.

Our contribution can be summarized as follows:

– We apply powerful Faster R-CNN detector to ECG domains for detecting heartbeat parts, and achieve quite high accuracy.

Z. Deng (Ed.): CIAC 2019, LNEE 586, pp. 356–363, 2020.
https://doi.org/10.1007/978-981-32-9050-1_41

– We propose a patch-sampling mechanism in training, leading to finer detection.
– Detection performance on real ECG images are also high.

2 Related Works

2.1 Heartbeat Segmentation

Heartbeat segmentation means segmenting heartbeat intervals from ECG signal. These intervals usually contain R peak, QRS complex, as shown in Fig. 1. Digital filters are widely used for this task. Sophisticated methods based on neural networks, wavelet transform, filter banks have also been used. The details are given in the survey [12] about heartbeat classification for arrhythmia detection.

It should be noted that, nearly all methods process the ECG signal but not image, and these algorithms are embedded into different devices. So only images like Fig. 1 can be shared conveniently. Our method try to detect heartbeat from the ECG images.

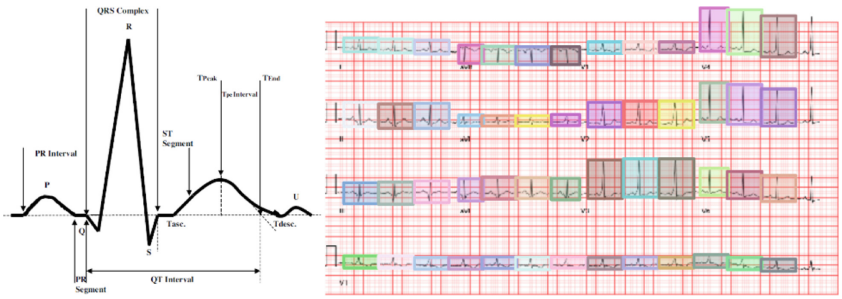

Fig. 1. Heartbeat segment in ECG image: The left means one heartbeat segment in ECG signal, and the right illustrates all heartbeat segments which is enclosed by colorful boxes. Our method takes charge of detecting these boxes quickly and accurately.

2.2 Object Detection

Classical Detectors. Early detection approaches were based on sliding-window, they classify the type of each sub-window seperately. Harr face detectors [2], HOG-based pedestrian detection [3], and part-based methods [4] belong to this. Although designed delicately and equipped with multi-scale strategy, performance is limited too.

ConvNet Detectors. Convolution neural network (CNN)-based detectors [5–7, 13] dominates object detection community recently. With enough training, they can defeat classical methods easily on multiple benchmarks. The R-CNN and its variants [5–7] gradually promotes the upper bound of performance on two-stage detectors. In particular, Faster R-CNN [7] adopts shared backbones to proposal (RoI, namely region of interest) generation and RoI classification, resulting realtime and accurate detection. Our method is based on Faster R-CNN [7]. Details will be described in Sect. 3.2.

3 Method

3.1 Heartbeat Detection from ECG Image

Traditional heartbeat segmentation means segmenting the part, like QRS complex, from ECG signal. Specifically in this paper, the task is localizing a box which encloses the QRS complex region, as shown in Fig. 1. Here we proposed a Faster R-CNN (Frcnn) based method to detect heartbeat part, from ECG image. Important components of Frcnn detector will be described. Then an improved training strategy will be introduced to lift detection performance on ECG image.

3.2 Faster R-CNN Detector

General Framework. The detection procedure of Frcnn starts feeding the image into Backbone Network to extract conv-feature, which is shared by two sub-networks. At the 1st stage, region proposal network (RPN) generates many RoIs based on this conv-feature. Then at the 2nd stage, Fast R-CNN extracts individual feature for each RoI. It then predicts the category and refines position. Finally, post-processing techniques like soft-nms [1] will be used to remove duplicates. In the following, more details of each component and the training mechanism will be introduced.

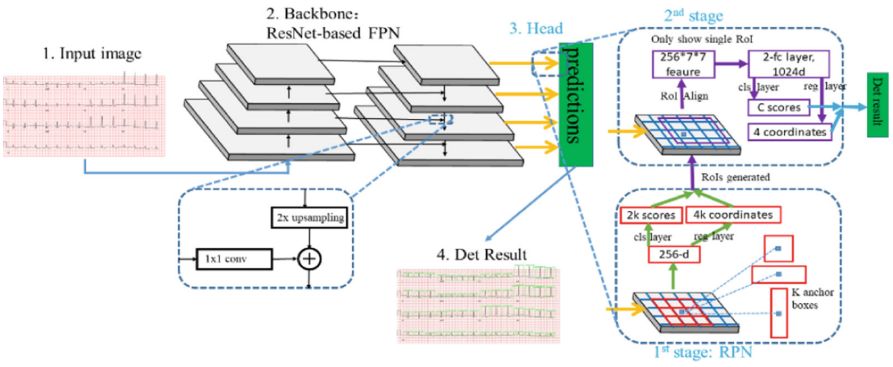

Fig. 2. Faster R-CNN on heartbeat detection

Backbone Network. It plays a significant role in extracting meaningful features for subsequent steps like RPN and Fast R-CNN. Fortunately, the development of CNN on vision is rapid and solid. Residual Network (ResNet) [10] is the outstanding, and becomes the standard configuration in conv-net based detectors.

Feature pyramid network (FPN) [11] is also proposed to detecting objects on feature of different levels. It regards the traditional ResNet as a bottom-up pathway of information, and proposes the top-down pathway and lateral connection to output multiple features with abundant details and semantics, as shown in Fig. 2. Besides, FPN detects

objects at disjoint scale ranges at corresponding levels. We choose FPN-based ResNet here, due to its better accuracy [11] than vanilla one in detection.

Head Subnet. Head predicts objects' position and categories subsequently. As shown in Fig. 2, at the 1st stage, RPN head receives the conv-feature of entire image and predefined anchor boxes [7]. At the end here, 'cls layer' predicts the probability that anchor box contain object and 'reg layer' performs bounding-box regression [5] to refine anchor box. More specifically, 'reg layer' predicts 4 values to transform old RoI (here is anchor box) to new RoI, as shown in Fig. 3.

$$
\begin{aligned}
v_x &= (G_x - P_x)/P_w \\
v_y &= (G_y - P_y)/P_h \\
v_w &= \log(G_w/P_w) \\
v_h &= \log(G_h/P_h)
\end{aligned}
$$

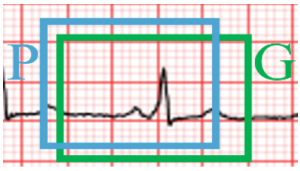

Fig. 3. bbox regression: At right, the blue rectangle RoI $P = (P_x, P_y, P_w, P_h)$ is an object proposal. The 4 elements means the P's center, width and height. And the green rectangle RoI $G = (G_x, G_y, G_w, G_h)$ is the ground-truth bbox of object. At left, the (v_x, v_y, v_w, v_h) is the prediction target for 'reg layer', the equation describes the RoI transformation.

At 2nd stage, Fast R-CNN head adopts RoI-Align [9] to extracting same-size feature for RoIs from 1st stage. At the end, 'cls layer' here conducts multi-class classification for each RoI, and 'reg layer' outputs 4 values for RoI refinement.

Network Optimization. Next we'll introduce the loss and sample strategy in training.

Loss Function. Considering Fast R-CNN, Cross-entropy loss for 'cls layer' is introduced for multiple classification on C categories, and smooth-L_1 loss [6, 7] for 'reg layer' is introduced for localization. The loss function for RPN is similar, while the C need to be 2 for classifying the foreground/background.

Pos/Neg RoI Sample for Training. Secondly, we need sample RoIs for training. The positive and negative ones are used for classification, while only the positive ones are for localization. For RPN and Fast R-CNN, Intersection-over-Union (IoUs) between input RoIs and ground-truth bboxes are calculated to determine pos/neg according to threshold in [7]. And the pos/neg ratio is set to 1:3 to reduce class-imbalance. The whole network is optimized by SGD, which details are in Sect. 4.1.

3.3 Patch-Based Training Scheme

Although Frcnn is strong enough for general object detection, it's still difficult to detect heartbeat from ECG image. Firstly, the context in ECG is almost the same red grid background. If the whole image participates in training, computation capacity is wasted in processing the repeated context region. Secondly, the scale of the bbox in ECG vary greatly. As shown in Fig. 1, some bboxes of heartbeats in I-type lead are almost

horizontal, while others in V2 are almost vertical. This makes it difficult to train detector network quickly. We turn to patch-based training for help.

What Is Patch? Here patch means a region in image which contains objects. We can just crop patches from original image for training to save computation. And cropped patches only label objects whose scale is normal, while ignore the extreme ones. These patches will be used for training network like SNIPER method [13].

Patch-Based Optimization. The core technique of patch-based optimization is finding meaningful patches. Analogously to SNIPER [13], designed Greedy Patch Generation (GPG) algorithm is as shown in Fig. 4. While SNIPER uses different ranges, uniform scale range V_s is used for consistency. In training, all patches from GPG will be used as samples. Other steps in inference are same as [13].

GPG(I_{orin} **,** B_{orin} **,** $S_{factors}$ **,** V_s **,** W **,** S **):**

$P \leftarrow \varnothing$
for $S_f \in S_{factors}$ **, do**
  Resizing I_{orin} **to** I **by** $\sqrt{S_f}$ **times in height and width**
  Generate corresponding B **for** I
  Sliding sub-window which size is $W \times W$ **with interval** S **, all these patches constitute the set** P^c
  for $b_i \in B$ **do**
    if scale(b_i**)** $\in V_s$ **then**
      B **.pop(**b_i**)**
    end if
  end for
  while $B \neq \varnothing$ **, do**
    $P_*^c = \arg\max_{P_i^c \in P^c} |\{b_i \mid b_i \in B \text{ if } b_i \text{ locates at } P_i^c\}|$
    $P^c \leftarrow P^c \setminus \{P_*^c\}$; $P \leftarrow P \cup \{P_*^c\}$
    $B \leftarrow B \setminus \{b_i \mid b_i \in B \text{ if } b_i \text{ is covered in } P_*^c\}$
  end while
end for
Return P

Fig. 4. GPG Algorithm. I_{orin} means the original image; $B_{orin} = \{b_1, \ldots, b_n\}$ contains all the objects in I_{orin}; $S_{factors}$ means the factors of scale; V_s means the scale range; W means the standard size of patch; S means the spatial interval of the sliding window.

4 Experiments

Experiments are performed on an ECG image dataset with only 1 category, namely heartbeat. All details about dataset, implementation and results are as follows.

4.1 Common Settings

ECG Image Dataset. Firstly, 764 ECG images are generated from real signals. The sizes are all 560 * 940. Secondly, each heartbeat is labeled by an bounding-box, as shown in Fig. 1. Thirdly, they are randomly divided into three disjoint subsets, namely train-set (252), val-set (252) and test-set (260). In addition, real-set consists of 10 images from smartphone. They will be used for demonstrating the effect on real scenes.

Experimental Procedure. Firstly, hyper-parameters are searched by training detection models on train-set and evaluating on val-set. Then, detectors are trained on train-set + val-set and evaluated on the test-set. Demo results will be obtained on real-set.

Implementation Details. If without specific description, following settings apply to all models. All models are implemented on the same codebase for comparison. The training and testing hyper-parameters are almost the same as Mask R-CNN [9]. The backbones and head subnets are the same as FPN [11], only plus ResNet-18-FPN. Models were trained on 4 GPUS with 1x and 2x strategy respectively. Here, 1x means that total iteration number for training is 900 and 2x means 1800, namely 29 and 59 epochs. The learning rate lr is initialized with 0.00125 * bs which is linear with mini-batch size bs like [8]. For 1x strategy, lr will be reduced by 10x at 19-th, 26-th epoch; while 38-th, 52-th epoch for 2x. Frcnn Detectors are trained and tested with single scale. The scale consists of (800, 1333) and (576, 1333). For patch-based training, the scale factor set is $S_{factors} = \{2.0, 1.0, 0.5\}$. And the scale range is $V_s = [16, 560]$.

4.2 Detection on Heartbeat

The evaluation is based on mAP (mean AP over multiple thresholds) from COCO.

Main Results. The main results are from training models on train-set+val-set, then evaluating them on test-set. Detailed comparison is shown in Table 1. The final highest mAP is 85.6, when training with scale = (576, 1333), 1x strategy and patch-based sampling, namely GPG algorithm.

Ablation Studies. They are conducted when training on train-set and evaluating on val-set. Different backbones and patch-sampling are experimented. Detailed comparison is shown in Table 1. The results shows: (1) For learning strategy, 2x always performs better than 1x; (2) scale (576, 1333) is more suitable for detection here than scale (800, 1333), due to that the ECG images are all (560, 940) here; (3) When the depth of ResNet increases, the mAP will promote; (4) Patch-based sampling is always better than original sampling strategy, under many different configurations. The (1–3) verify the effectiveness of selected hyper-parameters, and (4) demonstrate the superiority of GPG algorithm. Similar conclusion can be obtained from results on test-set.

Table 1 mAP comparison: Left are results on test-set while right are for val-set. At 1st row, 'M' means method, 'Net' means backbone network; '576 + 1x' means using 1x strategy on scale = 576. At 2nd column, 'F' means original Frcnn; '+P' means plus patch-based optimization. At 3rd column, 'R18' means ResNet-18-FPN and so on.

On val set						On test set			
M	Net	576 + 1x	800 + 1x	576 + 2x	800 + 2x	576 + 1x	800 + 1x	576 + 2x	800 + 2x
F	R18	79.7	76.7	82.3	79.8	82.7	81.1	84.2	82.6
F	R50	81.6	79.3	83.4	81.3	83.8	82.6	85.2	84.2
F	R101	82.2	81.2	83.6	83.1	84.7	84.2	85.4	85.3
+P	R18	80.9	77.6	82.9	80.6	83.8	81.2	84.8	83.2
+P	R50	82.4	80.9	83.9	82.2	84.5	82.5	85.4	84.9
+P	R101	83.1	81.8	83.8	83.6	85.1	84.9	**85.6**	85.5

Demo and Practical Application. It is necessary to evaluate the method on images from real world. Although there exists no annotation on real images, we can also compare methods based on their outputs. Detailed demo can seen in Fig. 5.

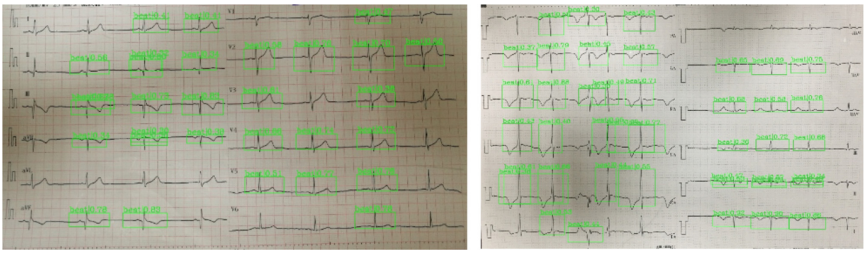

Fig. 5. Demo detection results in real scenes

5 Conclusion

We apply the powerful Faster R-CNN detector to detecting heartbeat part in ECG images, and achieves precise results. Along with the improved patch- sampling mechanism in training, higher evaluation metric can be achieved. In addition, some demo from real scenes demonstrate the effectiveness of our method.

Acknowledgement. This work was supported by the National Key R&D Program of China (No. 2016QY03D0501), by the National Natural Science Foundation of China (No. U1636220, NO. 61876183), by the Beijing Natural Science Foundation (No. 4172063).

References

1. Bodla N, Singh B, Chellappa R, Davis LS (2017) Soft-NMS-improving object detection with one line of code. In: Proceedings of the IEEE international conference on computer vision, pp 5561–5569
2. Viola P, Jones MJ (2004) Robust real-time face detection. Int J Comput Vision 57(2):137–154
3. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: International conference on computer vision & pattern recognition (CVPR 2005), vol 1. IEEE Computer Society, pp 886–893
4. Felzenszwalb PF, Girshick RB, McAllester D, Ramanan D (2010) Object detection with discriminatively trained part-based models. IEEE Trans Pattern Anal Mach Intell 32 (9):1627–1645
5. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
6. Girshick R (2015) Fast R-CNN. In: Proceedings of the IEEE international conference on computer vision, pp 1440–1448
7. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp 91–99
8. Goyal P, Dollár P, Girshick R, Noordhuis P, Wesolowski L, Kyrola A, Tulloch A, Jia Y, He K (2017) Accurate, large minibatch SGD: training imagenet in 1 hour. arXiv preprint arXiv:1706.02677
9. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask R-CNN. In: 2017 IEEE international conference on Computer vision (ICCV). IEEE, pp 2980–2988
10. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
11. Lin T-Y, Dollár P, Girshick RB, He K, Hariharan B, Belongie SJ (2017) Feature pyramid networks for object detection. In: CVPR, vol 1, p 4
12. Luz EJS, Schwartz WR, Cámara Chávez G, Menotti D (2016) ECG-based heartbeat classification for arrhythmia detection: a survey. Comput Methods Programs Biomed 127:144–164
13. Singh B, Najibi M, Davis LS (2018) SNIPER: efficient multi-scale training. In: Advances in neural information processing systems, pp 9333–9343