RESEARCH ARTICLE

Journal of Medical Imaging and Health Informatics
Vol. 8, 1–5, 2018

A Simple and Effective Method for Detecting Myocardial Infarction Based on Deep Convolutional Neural Network

Na Liu[1], Ludi Wang[1], Qing Chang[2], Ying Xing[1], and Xiaoguang Zhou[1,*]

[1]School of Automation, Beijing University of Posts and Telecommunications, Beijing, 100876, China
[2]Clinical Research Center, Jiading District Central Hospital Affiliated Shanghai University of Medicine and Health Sciences, Shanghai, 201800, China

Myocardial infarction (MI) is the main cause of sudden death in patients with cardiovascular diseases (CVD), thus timely detection of myocardial infarction is crucial for saving patients' lives. This paper presents an algorithm based on deep convolution neural network (CNN) to detect myocardial infarction, using electrocardiogram (ECG) signal from lead II. The algorithm proposed in this paper uses neither manual feature extraction nor feature selection, and instead of performing heartbeat segmentation, the method takes 3 second ECG signal segments as input. For our experiments, we conduct two datasets of denoised ECG set and original ECG set to corroborate the robustness of the algorithm to noise in ECG signal. We evaluate the model by a 10-fold cross-validation on the PTB database and achieve the state-of-the-art result: accuracy = 99.34%, sensitivity = 99.79% and specificity = 97.44% for the denoised ECG signal, and accuracy = 98.59%, sensitivity = 99.53% and specificity = 94.50% for the raw ECG signal.

Keywords: Electrocardiogram (ECG), Myocardial Infarction (MI), Deep Convolution Neural Network (CNN), MI Detection.

1. INTRODUCTION

Myocardial infarction (MI), normally known as a heart attack, is a myocardial damage occurs when blood flow diminishes or stops to a part of the heart. Myocardial infarction is a common cardiovascular disease, usually occurs after middle age, is one of the common symptoms of internal medicine, and its mortality rate can be as high as 25%. As reported in Ref. [1], approximately 15.9 million myocardial infarctions occurred globally in 2015. Therefore, the study of the detection of myocardial infarction has come into notice in the medical profession.

According to Ref. [2], an acute myocardial infarction can be diagnosed by an elevated cardiac biomarker and at least one of the following:

- Symptoms related to ischemia.
- Changes in electrocardiogram (ECG), such as ST segment changes, new left bundle branch block or Q wave.
- Changes in wall motion during imaging.
- Thrombosis at angiography or autopsy.

In the above points, using ECG to diagnose MI has the advantages of non-invasive, reproducible, real-time, simple and convenient. However, due to the small amplitude and short duration of ECG signals, manual detection of MI is not only time-consuming and difficult, but also may lead to inter and intra-observer variabilities.[3] Therefore, a computer-aided detection algorithm is necessary for effectively solving these problems.

Various methods have been conducted on MI detection. Several selected studies are summarized in Table VI, including Support Vector Machine (SVM) classifier based on multiple instance learning,[4] SVM classifier based on wavelet transform, multiscale energy and multiscale eigenspace analysis,[5] k-nearest neighbor classifier using Time-domain features,[6] k-nearest neighbor classifier using features extracted from wavelet coefficients after discrete wavelet transform (DWT)[7] and decision tree based on ECG polynomial fitting.[8] However, these studies usually need some preprocessing before classification, such as heartbeat segmentation, reference point identification, feature extraction and feature selection. These steps need to manually design complex rules, as well the selection of optimal parameters and feature sets is also difficult. Therefore, in order to achieve a simpler and more automated myocardial infarction detection algorithm, we introduce the convolutional neural network (CNN), one of the most popular network structure in deep learning.[9]

*Author to whom correspondence should be addressed.

First introduced by Lecun,[10] convolutional neural network is a kind of deep, feedforward artificial neural networks. It has been widely used in the field of pattern classification because it requires neither preprocessing nor separate feature extraction technique.[11] Nowadays, CNN has become one of the hot topics in many scientific fields including ECG signal. It has been utilized in arrhythmia detection,[12] coronary artery disease (CAD) detection[13] and beats classification.[14] In Ref. [12], a 34-layer convolutional neural network have been trained on a single-lead ECG signal to identify 14 arrhythmias, the ECG signal is sampled at 200 Hz with a sequence of annotations for every second as supervision. The result shows that the model exceeds the cardiologist performance in detecting a wide range of heart arrhythmias. In Ref. [14] researchers have developed an 11-layer CNN to detect myocardial infarction. In this model, each ECG record is divided into heartbeats, and then the label of whole ECG record is used as the label of each heartbeat as a supervision. Although this model has achieved satisfactory results, there are two properties that can be improved. First, the beat segmentation of each ECG record is complicated and may not be precise enough. Second, as mentioned above, the label of each beat is not necessarily right, for example, a record with MI label may contain sinus beats, which will result in incorrect training samples that reduce the model's accuracy.

Therefore, inspired by Ref. [12], a 13-layer CNN is trained end-to-end on 3 s ECG signal segments and a sequence of labels for every segment in this paper. This method not only eliminates the need for heartbeat segmentation, but also reduces the likelihood of introducing the wrong label compared to heartbeat training samples. The performance of the model is evaluated by a 10-fold cross-validation on the Physikalisch-Technische Bundesanstalt (PTB) dataset[15] collected from PhysioNet.[16] We achieve an average accuracy of 99.34% and 98.59% with denoised ECG signal and original ECG signal, respectively, indicating that our proposed algorithm can accurately detect MI from the unknown ECG signals even without noise removing.

2. DATA

The ECG signal used in this work are obtained from the PTB dataset.[15] The database contains 549 records from 290 subjects. Out of 290 subjects there are 148 MI subjects and 52 healthy controls (HC). Each record includes 15 simultaneously measured signals: the conventional 12 leads and the 3 Frank lead ECGs. Each signal is digitized at 1000 Hz. In this work, we use only lead II as it is a commonly used lead for basic cardiac monitoring. Further, lead II can provide good ECG morphological information.[14]

We use a total of 3,135 normal segments and 13,577 MI segments for this study. Table I presents the ECG signal we used in this work.

Table I. The ECG signal obtained from PTB database.

ECG type	Number of person	Number of segment
HC	52	3,135
MI	148	13,577
Total	200	16,712

3. METHODS

3.1. Pre-Processing

Each signal is downsampled from 1 kHz to 250 Hz for reducing the computational complexity. Then, in order to corroborate the robustness of the model to noise in ECG signal, we conduct two datasets of denoised ECG set and original ECG set.

We use the denoising method proposed in Ref. [17]: A two stage median filter is applied to remove the baseline wandering. The local window size for the first and second stage is chosen to be 125 and 249 respectively. Next, Savitzky-Golay (SG) smoothing filter is used to remove noise. The order and frame size of SG filter is selected as described in Ref. [18]. Figure 1 shows an example of each step for denoising. Finally, the signals are partitioned into short segments of 3 seconds (750 samples per segment). This yielded a total of 13,577 MI samples and 3,135 HC samples.

3.2. Network Architecture

A CNN consists of an input, an output layer, and multiple hidden layers. The hidden layers generally include convolutional layers, pooling layers, fully connected layers and normalization layers.[19] The input to our CNN network is a fixed-size 750 samples. The hidden layers in this paper consists of 13 layers: 4 convolutional layers, 4 pooling layers, 2 dropout layers, and 3 fully connected layers. Table II summarizes the details of the CNN structure.

(1) Convolutional Layer

The feature extractor composed of convolutional layer and subsampling layer is the main difference between convolutional neural network and general neural network. Different convolution kernels can extract different kinds of features. In this work, we use the valid padding and the same padding according to different requirements: the first and third layers are convoluted with valid padding, and the fifth and seventh layers are convoluted with same padding, both the convolution stride is fixed to 1 sample. The feature maps obtained for each convolutional layer are followed by a batch normalization (BN) layer, then activated with a rectified linear unit (Relu), and finally down-sampled with the max-pooling layer.

(2) Batch Normalization Layer

The role of BN is to normalize the new distribution of the output of the convolutional layer which changes as the training parameters changes. Batch normalization not only makes it easier to train a deeper network, speeds up convergence, but also has a regularization effect that prevents overfitting of models.[20]

(3) ReLU layer

The rectified linear[21] layer induces a nonlinearity in the values of the incoming layer. It only passes the values which are greater than zero, which is considered to have a certain biomimetic principle: studies in the relevant brains have shown that the coding of biological neurons is usually decentralized and sparse.[22] Normally, only about 1%–4% of neurons in the brain are active at same time. In this work, Relu is implemented as an activation function for layers 1, 3, 5, 7, 10 and 11. Also, the softmax function is used for layer 13 (last layer).

(4) Max Pooling Layer

Pooling, also known as subsampling, is often used in two forms: mean pooling and max pooling. Pooling can be seen as a special kind of convolution process, which can greatly reduce computational cost and provide translational invariance to the internal representation. The max-pooling operation is employed

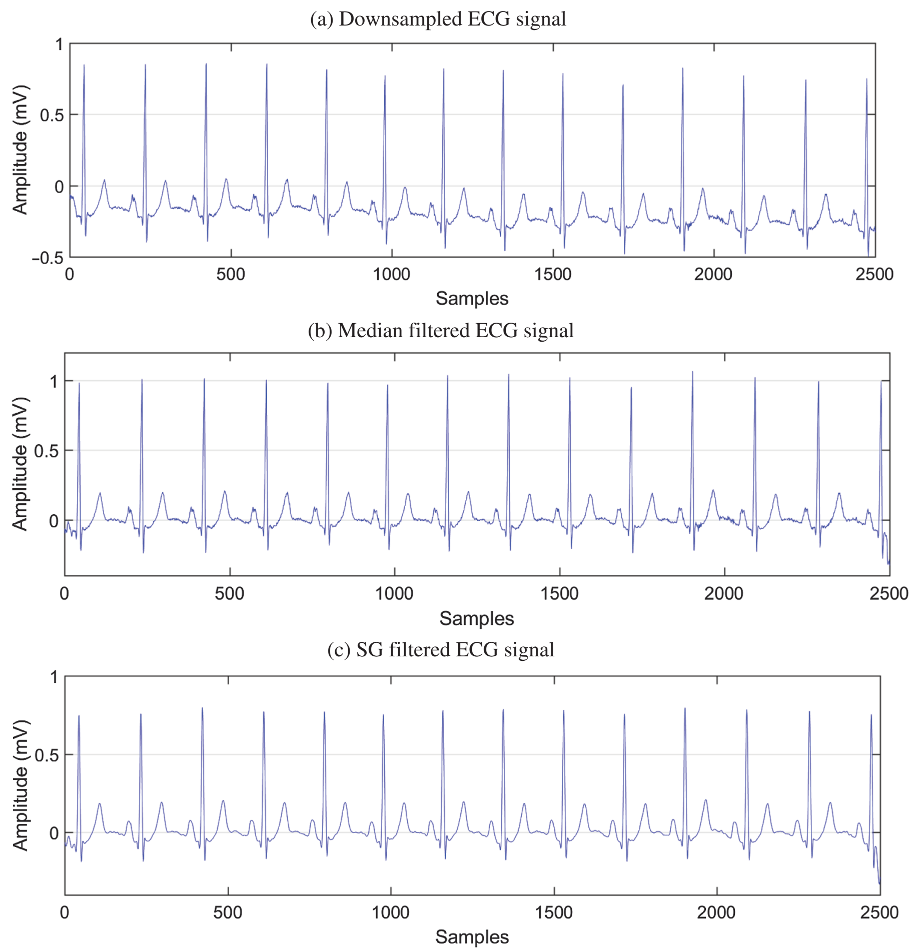

Fig. 1. An implementation of three stages for denosing.

in this work with the size of the pooling window and its stride chosen to be two samples.

(5) Dropout Layer

It is noted that dropout weakens the joint adaptability between neuron nodes, and effectively prevents over-fitting as well as improves the performance of model. In this work, dropout is used before the first and the last fully-connected layers. During training, we only need to sample the weights randomly according to the retention probability p, and use this subnet as the target network for update.[23] The retention probabilities in layers 9 and 12 are chosen to 0.25 and 0.5 respectively.

Table II. **The details of CNN structure.**

Layers	Type	Kernel size	Stride	Output shape
1	Convolution (V^a)	151×3	1	600×3
2	Max-pooling	2×3	2	300×3
3	Convolution (V)	45×10	1	256×10
4	Max-pooling	2×10	2	128×10
5	Convolution (S^b)	20×10	1	128×10
6	Max-pooling	2×10	2	64×10
7	Convolution (S)	10×10	1	64×10
8	Max-pooling	2×10	2	32×10
9	Dropout (0.25)	–	–	32×10
10	Dense	–	–	30
11	Dense	–	–	10
12	Dropout (0.5)	–	–	10
13	Dense	–	–	2

Notes: [a]Valid padding. [b]Same padding.

3.3. Training and Testing

We use the same training and testing methods for denoised signals and original signals. We employ a 10-fold cross-validation[24] strategy in this work. 90% ECG segments are used for the training of CNN, while the remainder (10%) of the ECG segments are used to test the performance of our proposed system. Out of the training ECG segments, we use 30% to validate our proposed algorithm. Figure 2 shows the proportion of ECG segments used for training, validation, and testing.

Categorical crossentropy is used as the cost function in this work. The crossentropy for two class problem is defined as:

$$L(p, y) = -y \log(p) - (1-y) \log(1-p) \tag{1}$$

Fig. 2. The proportion of ECG segments used for training, validation, and testing.

Here, $y \in \{0, 1\}$ represents the ground turth of each sample and p is the probabilty obtained from the trained model. The training is carried out by optimizing the categorical cross entropy using adam[25] optimizer. The batch size is set to 32. The weights of the last dense layer are regularized using l2 regularization with regularization parameter $\lambda = 0.001$ to prevent the model from overfitting. The learning rate is initialized to 10^{-3}, and then decreased by a factor of 10 when the validation set loss stopped decreasing for 5 consecutive epochs. The minimum learning rate is set to 10^{-5}. Training will be stopped if there is no improvement in training loss for 10 consecutive epochs or the epochs of training reached 100.

3.4. Metrics

The performance of the model is evaluated in terms of accuracy (Ac), sensitivity (Se), and specificity (Sp), which are defined as:

$$\text{Ac\%} = \frac{tp+tn}{tp+fp+tn+fn} \times 100 \tag{2}$$

$$\text{Se\%} = \frac{tp}{tp+fn} \times 100 \tag{3}$$

$$\text{Sp\%} = \frac{tn}{tn+fp} \times 100 \tag{4}$$

Here, tp is true positive prediction, fp is false positive prediction, tn is true negative prediction and fn is false negative prediction.

4. RESULTS AND DISCUSSION

Our implementation is derived from Keras neural network library. We trained our algorithm on a server with two NVIDIA Titan Xp GPUs and a 24 GB RAM.

The performance of the model is evaluated on the PTB database. The comparison of the proposed model trained on denoised signals and original signals is summarized in Table III. An average accuracy, sensitivity, and specificity of 99.34%, 99.79% and 97.44% are achieved using denoised ECG segments, respectively. Furthermore, an average accuracy of 98.59% sensitivity of 99.53% and specificity of 94.50% are obtained for ECG beat with noise. It totally takes 3557.26 seconds to complete the training and testing for denoised ECG segments data and 5065.13 seconds for ECG segments data with noise. The confusion matrix of denoised ECG segments and original ECG segments are presented in Tables IV and V for details.

As shown in Table VI, the majority of the researchers for automated detection of MI used all 12 lead ECG signals.[4~8] Also, the primary steps in these studies consist of (1) noise removal processing, (2) detection of QRS complex or R-peak, (3) extraction and selection of features, (4) classification of ECG signals. We did not execute step (1) and (2) in this work, and step (3) and (4) are carried out simultaneously in the training of CNN. Furthermore, in contrast to the researches in Table VI, we analyze the ECG signals in 3 seconds duration instead of ECG beats. This not only eliminates heartbeat segmentation but also increases the accuracy of the label.

The performance of our proposed system is comparable to the performances presented in Table VI. Our proposed model

Table III. Comparison of the proposed model trained on denoised signal and original signal.

Signal type	Ac (%)	Se (%)	Sp (%)	Time (s)
Denoised	99.34	99.79	97.44	3557.26
Original	98.59	99.53	94.50	5065.13

Table IV. Confusion matrix of denoised ECG segments across 10-folds.

	Ground truth	
Predicted class	MI	HC
MI	13548	81
HC	29	3054

Table V. Confusion matrix of original ECG segments across 10-folds.

	Ground truth	
Predicted class	MI	HC
MI	13513	172
HC	64	2963

Table VI. Summary of selected studies for automated detection of MI using ECG signals obtained from ptbdb.

Author, Year	Leads	Method	Performance (%)
Arif et al.,[6]	12 leads	• QRS detection • Time-domain features • k-nearest neighbor	Se = 99.97 Sp = 99.90
Sun et al.,[4]	12 leads	• ST detection • Multiple instance learning • Support vector machine	Se = 92.60 Sp = 82.40
Sharma et al.,[5]	12 leads	• Wavelet transform • Multiscale energy • Multiscale eigenspace analysis • Support vector machine	Ac = 96.00 Se = 93.00 Sp = 99.00
Liu et al.,[8]	12 leads	• R-peaks detection • ECG polynomial fitting • Decision tree	Ac = 94.40
Acharya et al.,[7]	12 leads	• R-peaks detection • Wavelet transform • k-nearest neighbor	Ac = 98.80 Se = 99.45 Sp = 96.27
Acharya et al.,[14]	Lead II	• R-peaks detection • 11-layer deep neural network	With noise: Ac = 93.53 Se = 93.71 Sp = 92.83 Without noise: cAc = 95.22 Se = 95.49 Sp = 94.19
This work	Lead II	• 13-layer deep neural network	With noise: Ac = 98.59 Se = 99.53 Sp = 94.50 Without noise: Ac = 99.34 Se = 99.79 Sp = 97.44

has achieved satisfactory results due to the introduction of deep learning methods into traditional ECG classification. It can avoid losses of effective features and enhance the accuracy since CNN learns the feature from the training data directly. In addition, we achieve comparable results for both with and without noise ECG segments. Therefore, this proves that our proposed method is robust to noise.

5. CONCLUSION

The timely diagnosis of MI is very critical to save lives. The traditional MI automatic detections are mostly based on 12-lead ECG. Also, denoising, heartbeat segmentation, and feature extraction are required before the classification. In this paper, we propose a novel algorithm to automatically diagnose MI using 13-layer deep CNN. The CNN is trained end-to-end on 3 s ECG signal segments from lead II. The performance of the model is evaluated by a 10-fold cross-validation on the PTB database, and we achieve satisfactory results of accuracy = 99.34%, sensitivity = 99.79% and specificity = 97.44% for the denoised ECG signal and accuracy = 98.59%, sensitivity = 99.53% and specificity = 94.50% for the original ECG signal. It is obvious that the proposed model is robust to noise, and the overall performance of our proposed algorithm is good enough. Moreover, the algorithm only uses ECG signals from lead II, which provides feasibility for applying our proposed system to wearable devices and mobile medical care.

Acknowledgment: This work was supported by 2015 China Special Fund for Grain-scientific Research in Public Interest (201513002) and the Fundamental Research Funds for the Central Universities (No. 2017RC27).

References and Notes

1. GBD 2015 Disease and Injury Incidence and Prevalence, Collaborators, Global, Regional, and National Incidence, Prevalence, and Years Lived with Disability for 310 Diseases and Injuries, 1990–2015: A Systematic Analysis for the Global Burden of Disease Study 2015 **(2016)**, pp. 1545–1602.
2. K. Thygesen, J. S. Alpert, A. S. Jaffe, M. L. Simoons, B. R. Chaitman, and H. D. White, Third universal definition of myocardial infarction. *Global Heart* 55.3, 275 **(2012)**.
3. R. J. Martis, U. R. Acharya, and H. Adeli, Current methods in electrocardiogram characterization. *Computers in Biology and Medicine* 48.3, 133 **(2014)**.
4. L. Sun, Y. Lu, K. Yang, and S. Li, ECG analysis using multiple instance learning for myocardial infarction detection. *IEEE Trans. Biomed. Eng.* 59.12, 3348 **(2012)**.
5. L. N. Sharma, R. K. Tripathy, and S. Dandapat, Multiscale energy and eigenspace approach to detection and localization of myocardial infarction. *IEEE Trans. Biomed. Eng.* 62.7, 1827 **(2015)**.
6. M. Arif, I. A. Malagore, and F. A. Afsar, Detection and localization of myocardial infarction using *K*-nearest neighbor classifier. *Journal of Medical Systems* 36.1, 279 **(2012)**.
7. U. R. Acharya, H. Fujita, V. K. Sudarshan, L. O. Shu, M. Adam, and J. E. W. Koh, Automated detection and localization of myocardial infarction using electrocardiogram: A comparative study of different leads. *Knowledge-Based Systems* 99, 146 **(2016)**.
8. B. Liu, J. Liu, G. Wang, K. Huang, F. Li, and Y. Zheng, A novel electrocardiogram parameterization algorithm and its application in myocardial infarction detection. *Computers in Biology and Medicine* 61.C, 178 **(2015)**.
9. E. Goceri and N. Goceri, Deep learning in medical image analysis: Recent advances and future trends, *International Conferences Computer Graphics, Visualization, Computer Vision and Image Processing* **(2017)**.
10. Y. Lecun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, and W. Hubbard, Backpropagation applied to handwritten zip code recognition. *Neural Computation* 1.4, 541 **(2014)**.
11. Y. Lecun and Y. Bengio, Convolutional networks for images, speech, and time series, The Handbook of Brain Theory and Neural Networks, MIT Press **(1998)**.
12. P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks **(2017)**.
13. U. R. Acharya, H. Fujita, O. S. Lih, M. Adam, J. H. Tan, and C. K. Chua, Automated detection of coronary artery disease using different durations of ECG segments with convolutional neural network. *Knowledge-Based Systems* **(2017)**.
14. U. R. Acharya, H. Fujita, O. S. Lih, Y. Hagiwara, J. H. Tan, and M. Adam, Application of deep convolutional neural network for automated detection of myocardial infarction Using ECG signals. *Information Sciences* 415 **(2017)**.
15. R. Bousseljot, D. Kreiseler, and A. Schnabel, Nutzung der EKG-signaldatenbank CARDIODAT der PTB über das internet. *Biomedizinische Technik and Biomedical Engineering* 40.s1, 317 **(1995)**.
16. A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, and R. G. Mark, Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals. *Circulation* 101.23, E215 **(2000)**.
17. L. D. Sharma and R. K. Sunkaria, Inferior myocardial infarction detection using stationary wavelet transform and machine learning approach. *Signal Image and Video Processing* 24, 1 **(2017)**.
18. C. J. Deepu and Y. Lian, A joint data compression and QRS detection scheme for wearable devices. *IEEE Transactions on Bio-Medical Engineering* **(2014)**.
19. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press **(2016)**, http://www.deeplearningbook.org/.
20. S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, *International Conference on International Conference on Machine Learning* **(2015)**, pp. 448–456.
21. V. Nair and G. E. Hinton, Rectified linear units improve restricted boltzmann machines, *International Conference on International Conference on Machine Learning Omnipress* **(2010)**, pp. 807–814.
22. D. Attwell and S. B. Laughlin, An energy budget for signaling in the grey matter of the brain. *Journal of Cerebral Blood Flow and Metabolism* 21.10, 1133 **(2001)**.
23. G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors. *Computer Science* 3.4, 212 **(2012)**.
24. R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification, 2nd edn., Wiley **(2001)**, pp. 55–88.
25. D. P. Kingma and J. Ba, Adam: A method for stochastic optimization. *Computer Science* **(2014)**.

Received: 18 March 2018. Revised/Accepted: 13 April 2018.