SHORT-SEGMENT HEART SOUND CLASSIFICATION USING AN ENSEMBLE OF DEEP CONVOLUTIONAL NEURAL NETWORKS

Fuad Noman[1], Chee-Ming Ting[1,2], Sh-Hussain Salleh[1], Hernando Ombao[2]

[1]School of Biomedical Engineering & Health Sciences, Universiti Teknologi Malaysia, Malaysia
[2]Statistics Program, King Abdullah University of Science and Technology, Saudi Arabia

ABSTRACT

This paper proposes a framework based on deep convolutional neural networks (CNNs) for automatic heart sound classification using short-segments of individual heart beats. We design a 1D-CNN that directly learns features from raw heart-sound signals, and a 2D-CNN that takes inputs of two-dimensional time-frequency feature maps based on Mel-frequency cepstral coefficients. We further develop a time-frequency CNN ensemble (TF-ECNN) combining the 1D-CNN and 2D-CNN based on score-level fusion of the class probabilities. On the large PhysioNet CinC challenge 2016 database, the proposed CNN models outperformed traditional classifiers based on support vector machine and hidden Markov models with various hand-crafted time- and frequency-domain features. Best classification scores with 89.22% accuracy and 89.94% sensitivity were achieved by the ECNN, and 91.55% specificity and 88.82% modified accuracy by the 2D-CNN alone on the test set.

Index Terms— Heart sound classification, convolutional neural network, ensemble classifiers

1. INTRODUCTION

Cardiac auscultation based on heart sound recordings or phonocardiogram (PCG) remains a primary screening tool for diverse heart pathologies. Various algorithms have been developed aiming at accurate automated classification of normal and abnormal PCGs [1]. However, the classification accuracy is still far from being reliable for diagnostics in clinical or non-clinical settings. One major challenge is to extract robust and discriminative features from the raw PCG recordings typically corrupted by various noise sources. Different time-frequency and statistical features have been employed in automatic heart sound classification. Heart-rate variability is the most widely-used feature, which however can only be extracted from long recordings containing many cardiac cycles. Here we consider the challenges in obtaining high PCG classification accuracy for single individual cardiac cycles.

Recent developments in deep learning (DL) techniques have seen remarkable success in many practical classification tasks, sometimes surpassing human-level performance [2]. This is owing to its inherent mechanism integrating both feature extractor and classifier, which permits learning of complex data representations with hierarchical levels of semantic abstraction via its multiple stacked hidden layers and hence the robust and accurate pattern classification even based on raw data or primitive features. It offers substantial gain in accuracy over traditional linear and kernel methods with shallow architecture. One popular DL architecture is the convolutional neural networks (CNN) which alternately stacks a convolutional layer to extract feature maps through sparse localized kernels with weight sharing, and a sub-sampling or pooling layer to acquire invariance to local translation. CNNs have achieved state-of-the-art performance in diverse challenging image recognition tasks [3, 4, 5].

Applications of DL to cardiac signals were introduced very recently [6, 7, 8]. CNNs have been used for normal/abnormal PCG classification using input features such as spectrogram and Mel-frequency cepstral coefficients (MFCCs) in [9] on 5-second windowed segments, and MFCC heatmaps of 3-second segments in [10]. Tschannen *et al.* [11] combined a wavelet-based deep CNN feature extractor with support vector machine (SVM) for heart-sound classification. Zhang *et al.* [12] proposed a segmental CNN model to detect cardiac abnormality with two different designs to adjust the configuration of convolutional layers filters. A DL architecture was implemented on field programmable gate array (FPGA) for real-time heart-sound classification using inputs based on gray sonogram images transformed from PCG segments [13].

In this paper, we propose a deep CNN for classification of pathology in PCG of a single heart beat. We design a new architecture called time-frequency ensemble CNN (TF-ECNN) that combines a 1D-CNN and a 2D-CNN using respectively the time-domain raw PCG signals and MFCC time-frequency representations as inputs. Our method was evaluated on the PhysioNet computing in cardiology (CinC) 2016 challenge database [14], the largest heart sound database available so far. The aim is to classify the heart sound signal from a short segment (single cardiac cycle - heartbeat) into normal and abnormal classes. We also investigated the performance of the proposed CNN model combined with different combination of input features, and compared with traditional classifiers, i.e., support vector machine (SVM), ensemble of decision trees and hidden Markov model (HMM). The hyper-

parameters tuning was carried out by using Bayesian optimization to find optimal values for model parameters [15] for all competing classifiers except HMM where the expectation-maximization algorithm was used to estimate the model parameters.

2. METHODS

In this section, we describe the main building blocks of heart sound classification algorithm consisting of preprocessing, segmentation, feature extraction and classification. For classification, we propose an ensemble of two deep CNNs that combines time-domain and frequency-domain input features, and consider three traditional approaches as baseline.

2.1. Database

We used heart sound recordings obtained from the PhysioNet CinC challenge 2016 database publicly available on PhysioNet website [16]. The dataset consists of 3153 recordings collected from healthy and pathological subjects. Recordings labeled as 'unsure' by the cardiologists regarding the normal or abnormal categories were not used, leaving a total of 2872 recordings for training and evaluation in this work.

2.2. Preprocessing

All the heart sound recordings were down-sampled to 1000 Hz and band-pass-filtered with Butterworth filter between 25 Hz and 400 Hz to eliminate the unwanted low-frequency artifacts (e.g., baseline drift) and high-frequency noise (e.g., background noise). The signals were then standardized by subtracting the mean and dividing by its standard deviation before feature extraction.

2.3. Segmentation

The whole heart sound recordings were segmented into short intervals of single beat and then classified into normal and abnormal categories. In this work we used the heart sound annotations provided with the database for segmentation of each recording into heartbeats (from the beginning of atrial activity to end of ventricular activity). Note that other data-driven unsupervised algorithms such as the Viterbi alignment can also be used to perform such segmentation. A total of 81503 segments were extracted from the whole database which were then partitioned into subject-oriented train and test datasets with balanced number of samples as shown in Table. 1.

2.4. Baseline Classifiers

We consider three baseline classifiers for comparison, namely, (1.) SVM with radial basis function kernel, (2.) ensemble of decision trees classifier and (3.) HMM.

SVM and decision tree ensemble. Following [8], a total of 58 features were extracted from each heartbeat for the SVM and ensemble of trees methods. These include 22 time-domain features (durations, skewness, kurtosis and sum of instantaneous amplitudes for each of the four heart sound states (S1, systole, S2 and diastole)) plus 36 frequency-domain features (median power spectrum for 9 frequency bands for each heart sound state). We further performed feature selection using the well-known neighborhood component analysis (NCA) [17], selecting a total of 28 features (16 time-domain and 12 frequency-domain). We carried out 5-fold cross-validation to optimize and tune the hyperparameters of the SVM and the tree ensemble classifiers. A Bayes optimization approach was used to minimize the loss function and select the best set of hyperparameters that produce best classification results. We also applied class weights when computing the classification accuracy to accommodate possible misclassification of normal class, since the database is slightly unbalanced.

Table 1. Distribution of train and test set of the Physionet CinC challenge 2016 database.

	Train		Test	
	normal	abnormal	normal	abnormal
Recordings	1150	284	1150	288
Heartbeats	32574	8170	32582	8177

HMM. Continuous HMMs with Gaussian mixture densities were used for modeling the temporal structure in PCG. We extracted a set of features as in [18]. A sequence of 12×1 MFCCs were computed over consecutive windowed frames for each heartbeat to obtain a two-dimensional $12 \times T$ time-frequency representation with T the total number of feature vectors. A 4-state HMM with left-to-right topology was employed to model the time evolution of the four distinct heart sound components in a single heartbeat. A mixture of 16 Gaussians was used as the observation model in each state. We found no practical improvement in classification accuracy for this data with larger number of Gaussian components. The HMMs were trained by using the Baum-Welch algorithm based on expectation-maximization to find the maximum likelihood estimates of the model parameters [19]. The Viterbi algorithm was used for aligning the MFCC frames to each of the four cardiac states, and to compute the likelihood score of a test example which was then classified to the HMM with the highest likelihood.

2.5. Proposed Ensemble CNN

Fig. 1 shows the architecture of the proposed time-frequency based ensemble deep CNN (TF-ECNN) model combining two distinct CNNs to capture the temporal structure in both the time-domain and frequency-domain. The first CNN (1D-CNN) accepts one-dimensional PCG time series data as input (i.e., the raw heartbeat signal). The second CNN (2D-CNN) uses the two-dimensional time-frequency feature maps of MFCCs and time-varying autoregressive (TV-AR) coefficients as input. For both the 1D-CNN and 2D-CNN, we used the same network architecture consisting of convolutional, activation, pooling and fully-connected (or dense) layers but with different sets of hyperparameters. The 1D-CNN was designed with asymmetric kernels to handle the one-dimensional PCG raw signals, while symmetric kernels were used in 2D-CNN to handle the two-dimensional PCG

Table 2. Summary of 1D-CNN model configurations.

Layer	Type	Output shape	Kernel size	Strides
1	Convolution	1000×8	6	1
2	Batch-Norm	1000×8	-	-
3	MaxPooling	500×8	2	2
4	Convolution	500×8	6	1
5	MaxPooling	250×8	2	2
6	Convolution	250×8	6	1
7	MaxPooling	125×8	2	2
8	Flatten	1000	-	-
9	Dense	512	-	-
10	SoftMax	2	-	-

Table 3. Summary of 2D-CNN model configurations.

Layer	Type	Output shape	Kernel size	Strides
1	Convolution	$96 \times 12 \times 16$	4	1
2	Batch-Norm	$96 \times 12 \times 16$	-	-
3	MaxPooling	$48 \times 6 \times 16$	2	2
4	Convolution	$48 \times 6 \times 16$	4	1
5	MaxPooling	$24 \times 3 \times 16$	2	2
6	Convolution	$24 \times 3 \times 16$	4	1
7	MaxPooling	$12 \times 1 \times 16$	2	2
8	Flatten	192	-	-
9	Dense	256	-	-
10	SoftMax	2	-	-

features.

2.5.1. Feature Extraction

The 1D-CNN was designed to classify the raw heart sound from fixed-length segments. However, the heartbeat segments are usually with variable lengths. Therefore, two approaches were used to normalize the segments durations. First, an anti-aliasing linear interpolation method was performed to normalize the heartbeats into reference duration (i.e. 1000 samples). Second, the segments with durations higher that 1200 samples were ignored (1.2% of total segments), then we zero-padded the rest of the segments to 1200 samples.

For 2D-CNN, we consider two approaches of feature extraction to obtain the two-dimensional time-frequency feature maps. First, similarly for the HMM classifier, we computed frames of short-time MFCC features on the duration-normalized PCG segments to produce feature maps of same size to represent each heart beat. Second, we computed autoregressive coefficients of 12-th order TVAR model commonly known as short-time linear predictive coefficients (LPCs) to construct alternative feature map for each segment.

2.5.2. Network Architecture & Training

Table 2 and Table 3 respectively summarize the architecture of the proposed 1D-CNN and 2D-CNN models individually. The experiments were carried out using TensorFlow platform [20] with Scikit-Optimize library which provides Bayesian optimization of the hyperparameters. We used the expectation-improvement method (with 100 iterations) to tune the CNN parameters, including learning rate, number of convolution layers, number of filters, kernel size, activation method, number of dense layers, number of nodes in dense layers, and dropout ratio of dense layers. We selected a fixed dropout ratio of 0.4 and 0.5 for the convolution layers of the 1D-CNN and 2D-CNN, respectively. All convolution layers used the zero-padding to preserve the input dimension. A batch-normalization layer was attached to the first convolutional layer to allow the model to learn different variations of the data which can give better robustness to noise typically present in real heart sound recordings. For other convolutional layers, we added dropout layer as regularization method to prevent model overfitting.

Of notes, additional experiments showed that the use of zero-padding in input segments for the 1D-CNN did not perform as well as using the duration-normalized segments with the same CNN architecture. Thus, the dropout ratio of the convolution layers was set to 0.8. The Bayesian optimization procedure suggested a 2D-CNN architecture with similar number of convolution layers with the 1D-CNN but slightly different number of dense layers. Therefore, we manually tuned the 2D-CNN architecture to match that of 1D-CNN which performed comparably with the Bayesian-optimized model. The learning rates set by the optimizer for the 1D-CNN and 2D-CNN were respectively 0.001031 and 0.000496 with batch size of 128. The Adam optimizer was used for weighs updating in the backpropagation training stage.

In the TF-ECNN, we combine both the 1D-CNN and 2D-CNN optimized above based on score-level fusion by summing over the outputs of softmax layers from two individual CNNs to produce fused class prediction probabilities.

3. EXPERIMENTAL RESULTS

We evaluate the classification performance of the 1D-CNN and 2D-CNN individually as well as the TF-ECNN, as measured by sensitivity, specificity and modified accuracy (MAcc). The MAcc is an average of the sensitivity and specificity scores. Table 4 shows the results of different classifiers and feature sets on the local hidden-test set.

Numbers in parentheses indicate performance of trained models without using the weighed-cost to control imbalances among the classes. They show that all classifiers do not perform well with a significant tradeoff between the sensitivity and the specificity. This is due to the imbalanced classes and limited abnormal data which lead to a high misclassification of abnormal segments as clearly indicated by the specificity scores. After corrections by applying class weights to limit the misclassification of abnormal class, performance of all classifiers increases except the ensemble of trees (still with low sensitivity and MAcc of below 80% but high sensitivity).

The proposed CNN models generally outperform the baseline classifiers considerably in most of the performance measures. In particular, the 2D-CNN with MFCCs achieved the best performance in specificity and MAcc, and the TF-ECNN gives the highest accuracy and the second highest

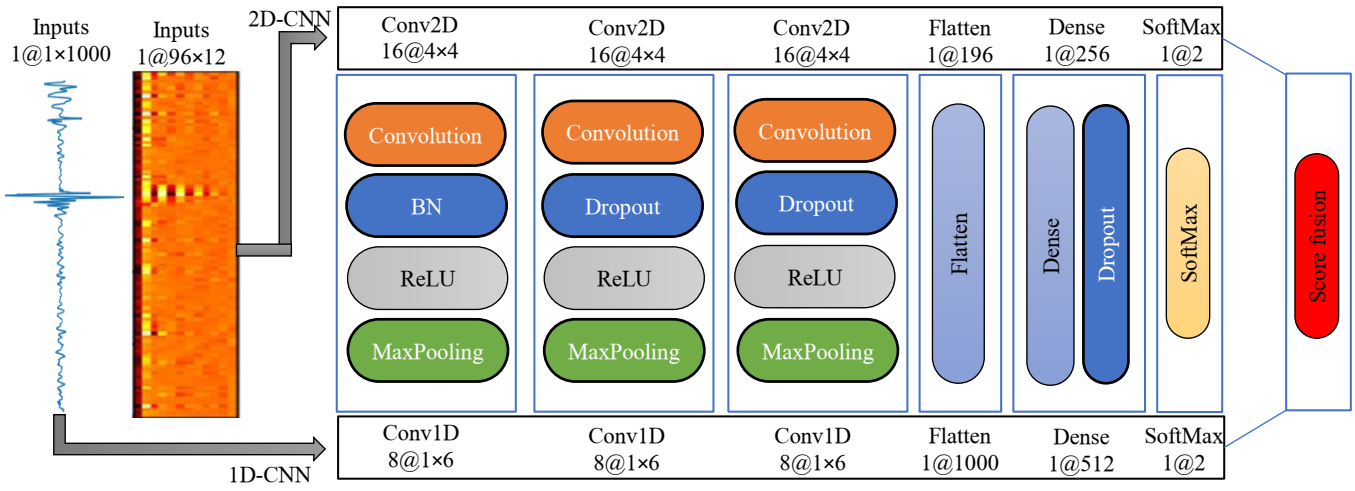

Fig. 1. Architecture of the proposed TF-ECNN model combining a 1D-CNN and a 2D-CNN taking inputs of raw signals and time-frequency feature maps, respectively. BN: Batch-normalization layer. ReLU: rectified linear unit activation function.

Table 4. Performance comparison of different classifiers on the test set. The numbers in parentheses correspond to the classifier performance before applying the weight cost for imbalanced classes.

Classifier	Features	Accuracy (%)	Sensitivity (%)	Specificity (%)	MAcc (%)
SVM	Time & Freq	84.87 (85.09)	85.82 (94.09)	81.09 (48.95)	83.46 (71.52)
Ensemble	Time & Freq	86.20 (86.23)	**90.55** (94.25)	68.84 (54.26)	79.70 (74.26)
HMM	MFCC	87.07 (n/a)	85.97 (n/a)	91.45 (n/a)	88.71 (n/a)
1D-CNN	Raw (zero-pad)	86.34 (85.63)	87.80 (95.11)	80.32 (46.41)	84.06 (70.76)
	Raw (norm-dur)	87.23 (87.52)	87.57 (91.51)	85.84 (71.64)	86.71 (81.58)
2D-CNN	TVAR	86.41 (86.91)	88.85 (91.79)	76.69 (67.45)	82.77 (79.62)
	MFCC	87.18 (89.30)	86.08 (92.49)	**91.55** (76.61)	**88.82** (84.55)
ECNN	Raw (norm-dur) + MFCC	**89.22** (89.58)	89.94 (93.07)	86.35 (75.68)	88.15 (84.37)

in sensitivity. HMM follows, performing the best among the traditional classifiers, possibly due to capability of the Markov chain in modeling the temporal structure of the four heart-sound states which is neglected by the SVM and even the CNNs. The performance of the ensemble of trees is not well-balanced, scoring highest sensitivity but with the lowest sensitivity and MAcc.

It is interesting to note that the 1D-CNN using only raw-data as input shows a satisfactory performance compared to using computationally-expensive feature extraction methods (i.e., MFCC and TVAR) in the 2D-CNN. The 1D-CNN with duration-normalized raw PCG is only 2% less than the best MAcc score obtained by the 2D-CNN with MFCC. This may suggest the advantages of the multiple hidden layers in CNNs that can learn hierarchical time-frequency features directly from the raw PCG signal. For the 2D-CNN with two-dimensional feature maps, the better time-frequency representation of the acoustic-based PCG signals using the MFCCs improves the classification performance over the TVAR. The ECNN combining both raw and MFCC features offer gains in sensitivity over the 2D-CNN using MFCC alone which however performs better in specificity, suggesting the advantage of ECNN in detecting the normal heart sounds whereas the 2D-CNN for the abnormal heart sounds.

4. CONCLUSION

We developed an ensemble of deep CNNs to classify normal and abnormal heart sounds based on short-segment recordings of individual heart beats with promising performance. The novel network architecture combines a 1D-CNN and a 2D-CNN designed respectively to learn multiple levels of representations from both the time-domain raw signals and time-frequency features. Evaluation on large PhysioNet CinC challenge 2016 database demonstrates advantages of our proposed CNN models with considerable improvement in classification performance over strong state-of-the-art baseline classifiers and feature sets. This suggests potentials of deep learning approaches for accurate heart-sound classification. Future works will consider use of sequential DL models such as the recurrent neural networks (RNNs) or long short-term memory (LSTM) RNNs [21] that could better capture the temporal dependency in the time-varying spectrum of PCG signals.

5. REFERENCES

[1] G. D. Clifford, et al., "Recent advances in heart sound analysis," *Physiol. Meas.*, vol. 38, pp. E10, 2017.

[2] S. Dodge and L. Karam, "A study and comparison of human and deep learning recognition performance under visual distortions," in *Computer Communication and Networks (ICCCN), 2017 26th International Conference on*. IEEE, 2017, pp. 1–7.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Int. Conf. Medical Image Computing and Compt.-Assisted Intervention*. Springer, 2015, pp. 234–241.

[5] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, 2017.

[6] Q. Zhang, D. Zhou, and X. Zeng, "HeartID a multiresolution convolutional neural network for ECG-based biometric human identification in smart health applications," *IEEE Access*, vol. 5, pp. 11805–11816, 2017.

[7] U. R. Acharya, et al., "A deep convolutional neural network model to classify heartbeats," *Computers in Biology and Medicine*, vol. 89, pp. 389–396, 2017.

[8] C. Potes, et al., "Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds," in *2016 Comput. Cardiol. Conf.*, 2016, pp. 621–624.

[9] T. Nilanon, et al., "Normal/abnormal heart sound recordings classification using convolutional neural network," in *Computing in Cardiology Conference (CinC), 2016*. IEEE, 2016, pp. 585–588.

[10] J. Rubin, et al., "Classifying heart sound recordings using deep convolutional neural networks and mel-frequency cepstral coefficients," in *Computing in Cardiology Conference (CinC), 2016*. IEEE, 2016, pp. 813–816.

[11] M. Tschannen, et al., "Heart sound classification using deep structured features," in *Computing in Cardiology Conference (CinC), 2016*. IEEE, 2016, pp. 565–568.

[12] Y. Zhang, et al., "Segmental convolutional neural networks for detection of cardiac abnormality with noisy heart sound recordings," *arXiv preprint arXiv:1612.01943*, 2016.

[13] J. P. Dominguez-Morales, et al., "Deep neural networks for the recognition and classification of heart murmurs using neuromorphic auditory sensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 1, pp. 24–34, 2018.

[14] C. Liu, et al., "An open access database for the evaluation of heart sound algorithms," *Physiol. Meas.*, vol. 37, no. 12, pp. 2181–2213, dec 2016.

[15] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," 2012.

[16] A. L. Goldberger, et al., "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[17] J. Goldberger, et al., "Neighbourhood components analysis," in *Adv. Neural Inf. Process. Syst.*, 2005, pp. 513–520.

[18] F. Noman, et al., "A Markov-Switching Model Approach to Heart Sound Segmentation and Classification," *arXiv Prepr. arXiv1809.03395*, 2018.

[19] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[20] M. Abadi, et al., "Tensorflow: A system for large-scale machine learning.," in *OSDI*, 2016, vol. 16, pp. 265–283.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.