This full text paper was peer-reviewed at the direction of IEEE Instrumentation and Measurement Society prior to the acceptance and publication.

ECGNet: Deep Network for Arrhythmia Classification

Balamurali Murugesan and
Vignesh Ravichandran
Healthcare Technology Innovation Centre
Indian Institute of Technology Madras
Chennai, India
balamurali@htic.iitm.ac.in

Sharath M Shankaranarayana
Department of Electrical Engineering
Indian Institute of Technology Madras
Chennai, India

Keerthi Ram, Preejith S.P and
Jayaraj Joseph
Healthcare Technology Innovation Centre
Indian Institute of Technology Madras
Chennai, India

Mohanasankar Sivaprakasam
Department of Electrical Engineering
Indian Institute of Technology Madras
Healthcare Technology Innovation Centre
Chennai, India

***Abstract*—Cardiac arrhythmias are presently diagnosed by manual interpretation of Electrocardiography (ECG) signals. Automated ECG interpretation is required to perform efficient screening of arrhythmia from long term ECG data. Existing automated ECG interpretation tools however require extensive preprocessing and knowledge to determine relevant features. Thus there is a need for a comprehensive feature extractor and classifier to analyze ECG signals. In this paper, we propose three robust deep neural network (DNN) architectures to perform feature extraction and classification of a given two second ECG signal. The first network is a Convolutional Neural Network (CNN) with multiple kernel sizes, the second network is a Long Short Term Memory (LSTM) network and the third network is a combination of CNN and LSTM based feature extractor, CLSTM network. The proposed networks are end to end networks which can be directly trained without any preprocessing. The networks were trained and tested with the MITDB ECG dataset on three classes Normal (N), Premature Ventricular Contraction (PVC) and Premature Atrial Contraction (PAC). The best model CLSTM gave an accuracy of 97.6%. Further, transfer learning is showcased on the best performing network for use with multiple ECG datasets requiring training only on the final three layers. The results showcase the potential of the network as feature extractor for ECG datasets. Our results outperform the state-of-the art works on ECG classification on several metrics.**

Keywords*—*Automated ECG interpretation; Arrhythmia; Deep Learning; Convolutional; LSTM; Physionet

I. Introduction

Cardiac arrhythmias are disturbances in the rhythm of the heart which are diagnosed by physicians by studying rhythm and the morphology of the ECG waveform [1]. Among cardiac arrhythmias, Premature Ventricular Contraction (PVC) and Premature Atrial Contraction (PAC) are highly prevalent [2], [3]. Despite being relatively benign, recent studies have shown the prognostic significance of PAC and PVC for detection of more severe arrhythmia and also the increased risks associated with long term PVC [4], [5]. Fig. 1 illustrates normal, PVC and PAC ECG rhythms. The present diagnosis procedure predominantly relies on visual inspection of ECG by a physician or clinical technician. The ECG is acquired through either a clinical 12-lead ECG recording, cardiac event recorder, Holter monitor or wireless wearable ECG patch [6] which allows for long-term unobtrusive monitoring. However manual ECG visual interpretation from such continuous wireless ECG monitors would be impractical. Automated detection and classification of arrhythmias would enable early diagnosis using the continuous wireless ECG monitoring devices. To perform automated diagnosis of arrhythmias in an ECG, the model must be able to characterize and recognize ECG rhythm morphology of the various arrhythmias while also taking the beat to beat time-relational elements into account.

Automated ECG interpretation has traditionally been done through a combination of QRS detection, feature extraction and machine learning classifiers. Early algorithms used to detect R-peaks by Pan *et al*. [7] were later replaced by wavelet transforms [8] and signal energy [9] based algorithms which have yielded more accuracy and speed. The machine learning classifiers such as Support vector machine [10] , Decision tree [11] have provided good results in the MITDB dataset. The drawback of such models is that it uses handcrafted features (time domain, frequency domain and beat to beat interval) alone without taking the raw QRS morphology into account which is crucial to determine atrial anomalies [12]. The presently used feature extraction approach requires deep domain knowledge and the features extracted lack robustness while handling different datasets and while encountering baseline wander and motion artifacts. The ECG signals are also subject variant and the extracted features for diagnosis would need deep insights about the particular abnormality [13]. Improvements in deep learning network architectures and access to powerful computing hardware has resulted in an influx of deep learning networks to diagnose cardiac arrhythmias

Fig. 1. First row: Normal ECG, Second row: PVC ECG, Third row: PAC ECG

with ECG. The latest being a 34-layer convolutional neural network (CNN) proposed by Rajpurkar *et al*. [12]. A key factor attributed by the authors for their success is their use of a well annotated dataset of 64,121 ECG records from 29,163 patients which was 500 times larger than the MIT-DB dataset. The CiNC 2017 Afib detection challenge had produced novel Deep learning networks for classifying between Normal and Atrial Fibrillation ECG [14]. Kan Luo *et al*. [15] converts the ECG to an image by a modified frequency slice wavelet transform and the resultant image is given to DNN for classification. A similar method was utilized by Rubin *et al* [16], Zihlmann *et al* [17]. through the use of a logarithmic spectrogram conversion. In the aforementioned approaches, the training part is not end-to-end due to the conversion of ECG to a spectrogram. A recurrent neural network (RNN) approach for ECG classification was proposed by Patrick *et al*. [18] which uses handcrafted features followed by a RNN network as a classifier.

The contributions of this paper is as follows:

1) We propose an improved CNN to aid in better feature extraction and thus increase the accuracy significantly over the recent state of the art.
2) We propose the use of Long Short Term Memory (LSTM) networks for the first time for ECG classification and also propose a novel scheme having a combination of CNN and LSTM to aid in faster convergence and improved accuracy.

II. METHODS

In this paper, three different architectures are proposed CNN (with inception), LSTM and CLSTM (CNN and LSTM). A CNN network proposed by Rajpurkar *et al*. [12] was used for comparison with the proposed architectures. The primary idea of the network is to build a robust DNN based feature extraction to derive features from two second ECG signals. The network would also be easily adaptable to multiple datasets requiring minimal training through the use of transfer learning.

A. Convolutional Neural Network Approach

CNN helps to learn the hierarchical representation of data and has been proven very effective in classification tasks. The major reason for this is because the CNN were able to provide robust feature vector. ImageNet classification task enabled researchers to come up with different feature extractor networks [19], [20]. Szegedy *et al*. [21] has shown that convolutional filters of different sizes learn distinct feature representations. Combining this features will provide better feature representation when compared to the one provided by single filter. Inspired by this, the ECG samples when given to the network are given to four parallel paths to convolution filters of different sizes (1x15,1x17,1x19,1x21). Feature maps extracted using the inception blocks are concatenated. An inception module in a convolutional neural network consists of convolution filters of multiple sizes as well as pooling within the same layer. It is followed by a batch normalization layer, a rectified linear unit (ReLU) activation layer and a max pooling layer. Inspired by the residual like architecture proposed by Rajpurkar *et al*. [12] for ECG classification, 15 residual blocks with two convolutional layers each were added to the network. Each convolution is preceded by batch normalization, ReLU activation block and dropout block to reduce over fitting and to accelerate the training procedure. The last layer of the residual block is given to a Batch normalization and ReLU activation block before it is given to the fully connected layers which acts as the output classifier. The important blocks of the proposed network are described below. The architecture of the proposed CNN network is shown in Fig. 2.

1) Convolutional Block: Convolutional layer helps to extract feature map on given input signal. In this paper, 1D convolution is carried out. The convolution filter size is fixed to 1x17. This filter size is determined through experimentation. The number of convolutional filters is changed for each 4^{th} residual block. This kind of approach enables the network to learn hierarchical representation as the network goes deeper. In the initial stage alone, a parallel path of different filter size is considered for convolution enabling the successive blocks to get a better representation.

2) Batch Normalization Layer: Training a DNN is complicated by the fact that distribution of each layer's inputs changes during training, as the parameters of the previous layers change. This slows down the training by requiring lower learning rates and careful parameter initialization, and makes it very difficult to train models. This phenomenon is called internal covariate shift. To overcome this, Ioffe *et al*. [22] proposed that the network training converges faster if it's inputs are whitened (linearly transforming the input to have zero means and unit variances).

3) ReLU Activation: The ReLU or Rectified Linear Unit activation function is used to introduce a nonlinearity in the values of the incoming layer. Nair *et al*. [23] has reported that using ReLU gives faster convergence and increased accuracy. The ReLU function is represented as f(x) = max(x, 0).

4) Dropout Layer: The Dropout [24] is a regularization technique where units are dropped at random. This proce-

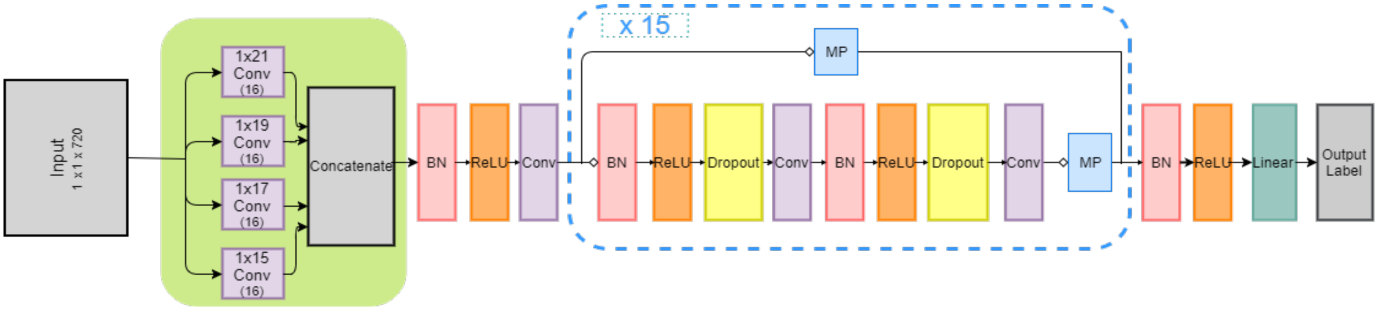

Fig. 2. Proposed CNN Architecture

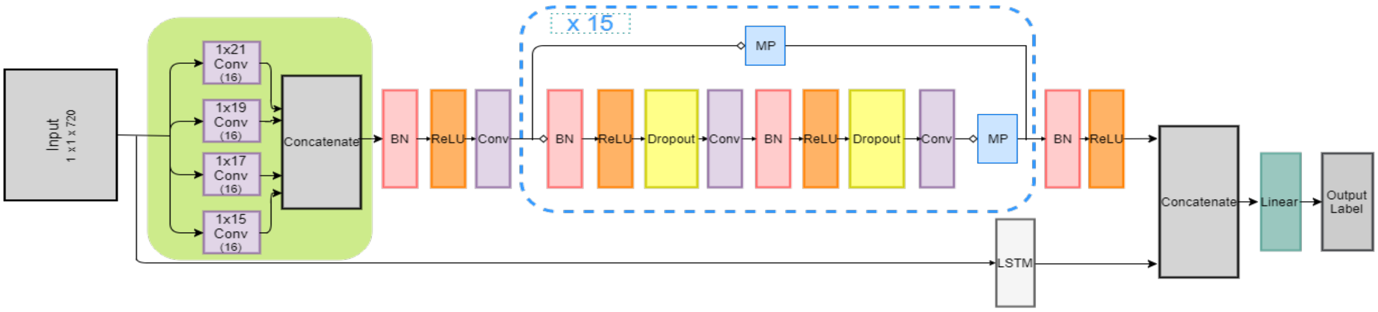

Fig. 3. Proposed CLSTM Architecture

dure helps in preventing the network to form complex co-adaptations. This significantly reduces over fitting and improves accuracy.

5) MaxPool Layer: Maxpooling helps to make the representation become approximately invariant to small shift in the input [25]. After applying maxpooling, the dimension of the input is reduced by half, also the values correspond to the maximum output of the spatial neighborhood. The spatial neighborhood covered is defined by a filter of size 2 and stride 2.

B. Long Short Term Memory Approach

An LSTM network is a modified version of a RNN which allows a network to remember long term dependencies to prevent the vanishing gradient problem [26]. LSTM networks have been used widely for time series datasets such as financial datasets with great results [27]. The LSTM network proposed consisting of 12 timesteps each having 60 samples, outputs a vector of size 40 which is fed to a fully connected layer which outputs the classified labels.

C. Convolutional Long Short Term Memory Approach

The CNN network and the LSTM networks are combined to form a new network as shown in Fig. 3. The computed 640 element output vector of the CNN network are concatenated with the 40 element output vector of the LSTM network to yield a 680 element feature vector which is then passed to the fully connected layers for classification.

This study uses ECG signals from the physionet MITDB Arrhythmia database for developing the network [28].

Abbrevations:

1) Conv- Convolutional Block
2) BN- Batch Normalization
3) ReLU- Rectified Linear Unit activation function
4) Dropout- Dropout Block
5) MP- Max Pooling Block
6) LSTM- Long Short Term Memory Block
7) Linear- Fully Connected Block
8) Concatenate-Concatenation operation

D. Dataset Description

TABLE I
DATASET DISTRIBUTION

Dataset Name	N beats	V Beats	A beats	Total Beats
MITDB	60309	12034	2584	74927
LTDB	20001	5137	0	25138
LTAFDB	30000	1318	14914	46232

It is the most commonly used database for research in ECG signal processing. It consists of 48 annotated, 30 min ambulatory ECG records from 2 leads (II and modified V1, V2, V3, V4, or V5 leads) obtained from 47 subjects and sampled at 360 Hz per channel. Since lead II ECG is commonly used in single lead ECG patch applications, Lead II data alone was used in the current study. Despite having 15 beat type annotation, this network is built to test three rhythm types Normal, PAC and PVC. The following denotations are used from here on to represent three rhythm types:

N: Normal beat

V: Premature Ventricular Contraction

A: Premature Atrial Contraction

Other annotated Lead II datasets such as the LTDB and LTAFDB datasets from Physionet were also used to train and test the model. Due to the variance in sampling rate among the other datasets, upsampling was applied to match the MITDB dataset. Samples of length two seconds are labelled accordingly and split into train and test sets. Table I describes the sample distribution of various datasets used to train and test the network.

E. Training Method

The input signal is preprocessed by L2-normalization procedure before it is given to the network. The network is designed for a fixed input of 720 samples and output of 3 classes. Random weight initialization is performed on the network weights. Cross entropy loss function as defined in equation (1) is used for the training. The learning rate is fixed to 0.01, momentum to 0.9, mini-batch size to 8 and number of epochs to 50. The weights are updated by stochastic gradient descent(SGD). The networks were trained using the parameters mentioned above. The ECG classification is performed to an input ECG signal X against a given ground truth label r, The proposed model outputs a sequence of labels R for n classes. The loss function L is as described in (1) and p is the probability of the network assigning the i-th class for a given input scene.

$$L(X, r) = \frac{1}{n}\sum_{i=1}^{n} log\, p(R = r_i|X) \tag{1}$$

The CNN (without inception), CNN (with inception) and CLSTM converged with lesser number of epochs than LSTM. The training procedure in this paper can be represented by three separate methods:

1) The class distribution of 5000N, 2500V and 2500A rhythms were taken from the MITDB. All the four networks are trained with 4500N, 2000V and 2000A.
2) The best network (CLSTM) was taken and 10 different models were obtained by training it on fixed 2000V, 2000A rhythms and on 10 sets of 4500 random N rhythms which guarantees covering 70% variations of N rhythm. This way of creating different models from sampling a dataset is called bagging [29].
3) To verify the ability of network to be used as a feature extractor, two publicly available datasets LTAFDB and LTDB are considered. In LTAFDB, 1300 from each class 1000 for training and 300 for testing is taken. In LTDB, 5000 from N and V is taken, 4500 for training and 500 for testing. The LTDB database doesn't have any A rhythm. For this datasets, the best network (CLSTM) was taken, instead of training the whole network again, transfer learning is applied wherein only the final 3 layers of the network are updated.

The discussed models are built using PyTorch [30] and run on a Nvidia GTX1060 6GB GPU.

F. Testing Method

The following are the testing procedures conducted.

1) The test data 500 from each class is taken and verified with CNN (without inception), CNN (with inception), LSTM and CLSTM.
2) In order to check the generality of the best model, the whole MITDB dataset is tested with the network.
3) The whole MITDB dataset is tested with the 10 different trained models. The output predicted from each model is taken and maximum voting is done to get the best prediction.
4) The test data of 300 from each rhythm in LTAFDB and 500 from each rhythm in LTDB (excluding A) is taken and given to the best model CLSTM.
5) The network which is updated by transfer learning is taken, to which the test data from LTDB and LTAFDB are given.

G. Evaluation Metrics

For evaluation, we use the standard metrics used for classification tasks. Accuracy is defined as the ratio of correct predictions to the total number of predictions. Precision is defined as the ratio between the number of true positives to sum of number of true positives and the number of false positives. Recall is defined as the ratio between the number of true positives to sum of number of true positives and the number of false negatives.

$$Accuracy = \frac{TP + TN}{P + N}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FP}$$

$$F1score = \frac{2 * Precision * Recall}{Precision + Recall}$$

TP = True positive, FP = False positive,
P = Total Positive, N = Total Negative

III. Results and Discussion

The CNN (without inception), CNN (with inception), LSTM and CLSTM models were trained on 4500N,2000V and 2000A rhythms and tested with a 500 rhythms of the three classes. The accuracies obtained are reported in Table II. From this, it can be inferred that CLSTM gave best accuracy followed by CNN (with inception layer),CNN (without inception layer) and LSTM network. The improved performance of the CLSTM network can be attributed to having features from both and CNN and LSTM.

The best model (CLSTM) was verified with the entire MITDB dataset. The obtained classification measures are reported in Table III. The low average F1-score observed is due to low precision in A and low recall in N. It is clear that the problem is caused by class imbalance in training, In particular it is caused by low number of N rhythms considered for training. This can be resolved by considering the predictions made by maximum voting where the prediction is made by ensemble of 10 models whose datasets are created by bagging. As can be seen from Table IV, there is an increase in precision of A, recall of N and average F1-score.

Testing with the other datasets (LTDB, LTAFDB), the best model CLSTM was used in two ways, one is with the pretrained weight while the other is with the transfer learning applied where only the final 3 layers are updated.

TABLE II
ACCURACIES OF MODELS TRAINED ON MITDB

Architecture	Accuracy
LSTM	89.4%
CNN without inception layer	93.4%
CNN with inception layer	95.7%
CLSTM	97.6%

TABLE III
SCORES WITHOUT BAGGING

Class	Precision	Recall	F1-score	Total
Normal	99%	88%	93%	60309
PVC	85%	96%	90%	12034
PAC	30%	93%	46%	2584
Avg	95%	89%	91%	74927

TABLE IV
SCORES WITH BAGGING

Class	Precision	Recall	F1-score	Total
Normal	99%	98%	99%	60309
PVC	97%	97%	97%	12034
PAC	75%	96%	84%	2584
Avg	98%	98%	98%	74927

TABLE V
ACCURACIES OF ALTERNATE DATASETS

Dataset	Accuracy	
	Pretrained weight	Transfer Learning
LTAFDB	67%	97%
LTDB	80%	98%

Table V shows the comparison of accuracies of the two datasets LTAFDB and LTDB. The increase in accuracies after applying transfer learning showcases the ability of the network before the final 3 layers to act as a feature extractor.

IV. CONCLUSION

Three end-end deep learning networks to perform feature extraction of ECG and classification were proposed. The model performance was validated with multiple datasets such as the MITDB, LTDB & LTAFDB arrhythmia databases. Out of the three models the CLSTM model had shown an overall superior performance with an accuracy of 97.6%. Further tests using multiple datasets revealed the ability of the network in being a good feature extractor for any ECG arrhythmia dataset requiring only minimal training. In future, we would like to extend the network to classify additional classes of arrythmias such as Atrial fibrillation, Atrial flutter, Bigmeny, SVTs and other ectopic rhythms. To realize this, however, enough labelled data must be collected in each of these abnormalities. Further data augmentation techniques would need to be explored. Training with noisy ECG data would enhance network performance in real world. Adapting similar networks for multi-lead classification can also be explored in the future. The network would also present a potential choice for individual automatic heartbeat classification used in wireless ECG monitoring devices and can offer a data abstracted service for diagnosing rhythm defects.

ACKNOWLEDGMENT

We would like to acknowledge PhysioNet, from where we have obtained the datasets.

REFERENCES

[1] J. A. Kastor, *Arrhythmias*. WB Saunders Company, 2000.

[2] H. L. Kennedy, J. A. Whitlock, M. K. Sprague, L. J. Kennedy, T. A. Buckingham, and R. J. Goldberg, "Long-term follow-up of asymptomatic healthy subjects with frequent and complex ventricular ectopy," *New England Journal of Medicine*, vol. 312, no. 4, pp. 193–197, 1985.

[3] D. Conen, M. Adam, F. Roche, J.-C. Barthelemy, D. F. Dietrich, M. Imboden, N. Künzli, A. von Eckardstein, S. Regenass, T. Hornemann *et al.*, "Premature atrial contractions in the general population: frequency and risk factors," *Circulation*, pp. CIRCULATIONAHA–112, 2012.

[4] M.-S. Ahn, "Current concepts of premature ventricular contractions," *Journal of lifestyle medicine*, vol. 3, no. 1, p. 26, 2013.

[5] C.-Y. Lin, Y.-J. Lin, Y.-Y. Chen, S.-L. Chang, L.-W. Lo, T.-F. Chao, F.-P. Chung, Y.-F. Hu, E. Chong, H.-M. Cheng *et al.*, "Prognostic significance of premature atrial complexes burden in prediction of long-term outcome," *Journal of the American Heart Association*, vol. 4, no. 9, p. e002192, 2015.

[6] S. Preejith, R. Dhinesh, J. Joseph, and M. Sivaprakasam, "Wearable ecg platform for continuous cardiac monitoring," in *Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the*. IEEE, 2016, pp. 623–626.

[7] J. Pan and W. J. Tompkins, "A real-time qrs detection algorithm," *IEEE transactions on biomedical engineering*, no. 3, pp. 230–236, 1985.

[8] S. Barmase, S. Das, and S. Mukhopadhyay, "Wavelet transform-based analysis of qrs complex in ecg signals," *arXiv preprint arXiv:1311.6460*, 2013.

[9] J. Kim and H. Shin, "Simple and robust realtime qrs detection algorithm based on sp atiotemporal characteristic of the qrs complex," *PloS one*, vol. 11, no. 3, p. e0150144, 2016.

[10] Z. Zidelmal, A. Amirou, D. Ould-Abdeslam, and J. Merckle, "Ecg beat classification using a cost sensitive classifier," *Computer methods and programs in biomedicine*, vol. 111, no. 3, pp. 570–577, 2013.

[11] A. T. Sadiq and N. H. Shukr, "Classification of cardiac arrhythmia using id3 classifier based on wavelet transform," *Iraqi J. of Sci.*, vol. 54, no. 4, pp. 1167–1175, 2013.

[12] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," *arXiv preprint arXiv:1707.01836*, 2017.

[13] J. Wiens and J. V. Guttag, "Patient-adaptive ectopic beat classification using active learning," in *Computing in Cardiology, 2010*. IEEE, 2010, pp. 109–112.

[14] G. D. Clifford, C. Liu, B. Moody, L.-w. H. Lehman, I. Silva, Q. Li, A. Johnson, and R. G. Mark, "Af classification from a short single lead ecg recording: the physionet/computing in cardiology challenge 2017," in *Computing in Cardiology, 2017*. IEEE, 2017, pp. 1–4.

[15] K. Luo, J. Li, Z. Wang, and A. Cuschieri, "Patient-specific deep architectural model for ecg classification," in *Journal of healthcare engineering, 2017*. Hindawi, 2017.

[16] J. Rubin, S. Parvaneh, A. Rahman, B. Conroy, and S. Babaeizadeh, "Densely connected convolutional networks and signal quality analysis to detect atrial fibrillation using short single-lead ecg recordings," *arXiv preprint arXiv:1710.05817*, 2017.

[17] M. Zihlmann, D. Perekrestenko, and M. Tschannen, "Convolutional recurrent neural networks for electrocardiogram classification," *arXiv preprint arXiv:1710.06122*, 2017.

[18] P. Schwab, G. Scebba, J. Zhang, M. Delai, and W. Karlen, "Beat by beat: Classifying cardiac arrhythmias with recurrent neural networks," *arXiv preprint arXiv:1710.06319*, 2017.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[21] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich *et al.*, "Going deeper with convolutions," *CVPR*, 2015.

[22] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *International conference on machine learning*, pp. 448–456, 2015.

[23] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, 2010.

[24] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[25] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," *International conference on artificial neural networks*, pp. 92–101, 2010.

[26] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[27] K. Bandara, C. Bergmeir, and S. Smyl, "Forecasting across time series databases using long short-term memory networks on groups of similar series," *arXiv preprint arXiv:1710.03222*, 2017.

[28] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[29] T. Hothorn and B. Lausen, "Bagging combined classifiers," *Classification, Clustering, and Data Analysis*, pp. 177–184, 2002.

[30] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "Pytorch," 2017.