Fetal Electrocardiogram Recognition using Multilayer Perceptron Neural Network

Boyang Wang and Jafar Saniie
Department of Electrical and Computer Engineering
Illinois Institute of Technology, Chicago, Illinois, U.S.A.

***Abstract* – Fetal Electrocardiography (FECG) signal contains valuable and meaningful information that would help doctors to make decisions during pregnancy and labor. It is also an important indicator of the fetal status. However, extracting FECG from non-invasive sensors is not easy since the FECG signal is weak compared to the Maternal ECG (MECG) signal. In conventional signal processing methods, it requires an adaptive filter with the MECG signal and the mixture of Electrocardiography (ECG) signal to reveal the FECG signal. This procedure requires significant computation power and multiple sensors applied on the pregnant women. As machine learning algorithms become more and more popular, applying neural network to signal processing is widely adapted in all types of applications. This paper presents a method based on neural network to recognize the FECG signal from the abdominal ECG signal acquired by non-invasive sensors. Training and evaluation procedure are achieved in TensorFlow on a heterogeneous platform. This algorithm can precisely identify both MECG and FECG signal from the maternal abdominal ECG signal.**

Keywords– FECG recognition, non-invasive, Neural Network

I. INTRODUCTION

The FECG signal is an electrical signal that is acquired by applying electrodes to abdomen of pregnant women [1]. FECG contains very useful information of fetal status which can help doctors to make important decisions. As the Maternal ECG (MECG) is usually much stronger than the FECG with non-invasive abdominal ECG (AECG) signal, it is difficult to extract FECG signal from AECG. This paper presents an algorithm to recognize MECG and FECG from AECG signal acquired non-invasively. A few other non-invasive methods such as doppler ultrasound or magnetometer can detect the fetal heartrate [2, 3]. However, all of them requires professional skill and heavily relies on computation power. The method presented in this paper is based on multilayer perceptron neural network. Collecting AECG signal using electrodes is not as sophisticated compared to ultrasound transducers. Once the training procedure is done, it doesn't require heavy computation power to detect the fetal heartrate. Results show that the algorithm can detect both MECG and FECG from AECG precisely.

The neural network training and evaluating process is done by using Google TensorFlow with GPU support [4]. A few python libraries are used to add visualization to the program. A desktop computer is used to run all the tests. It is a heterogeneous platform contains an Intel(R) Core(TM) i7-6700 CPU @ 3.4 GHz and a NVIDIA GeForce GTX 1070 @ 1.506 GHz as an algorithm accelerator.

Section II of this paper discusses the background of the FECGSYN and Multi-Layer Perceptron (MLP) Neural Network. Some basic information about using TensorFlow for machine learning algorithm is also introduced in this section. In Section III, the implementation of FECG and MECG extraction based on MLP neural network is discussed in detail. Also, the results and performance are presented in this section. Section IV concludes the paper.

II. BACKGROUND

The test dataset is generated by using an opensource fetal ECG synthetic simulator called FECGSYN in MATLAB. The reason that we decided to use a simulation database is that the collection of FECG signal varies too much in different scenario. The main purpose of this study is to detect FECG using neural network but not how to obtain better FECG signal. FECGSYN can generate maternal-fetal ECG mixtures with realistic amplitudes, morphology, beat-to-beat variability, heart rate changes and noise [5, 6].

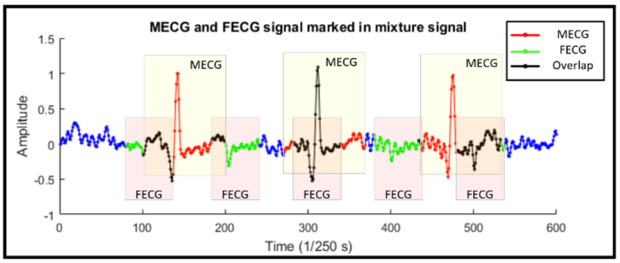

Figure 1. Abdominal ECG Signal

Figure 1 shows one channel of simulated signals by using FECGSYN in MATLAB. In this setup, the sample rate is set to 250 Hz; Mother's heart rate is set to 90 beats per minute (bpm) and the fetal heart rate is set to 150 bpm. FECGSYN returns a simulated dataset contains 8 channels of maternal AECG data. Figure 1 is the AECG signal with both MECG and FECG marked with different color. As shown in this figure, the third FECG is completely overlapped with the second MECG signal. The purpose of this algorithm is to detect FECG and MECG in maternal AECG signal by using MLP neural network algorithm.

978-1-5386-5398-2/18/$31.00 ©2018 IEEE

Multilayer perceptron neural network is a class of feedforward artificial neural network. It is an artificial neural network with at least three layers. MLP neural network is trained using supervised method called backward propagation. The algorithm of using MLP neural network for recognition has been discussed in other papers [7, 8]. The programming language that is used to build a neural network in TensorFlow is Python. Python is an interpreted high-level programming language which is not a good option when performance is the major consideration. Some python libraries like NumPy can accelerating some expensive algorithms by executing the algorithms outside the python environment. However, such scheme will inevitably add a lot of overhead and slow down the processing. The overhead is much more worse when the acceleration is done by using GPUs. TensorFlow adapts similar scheme by executing the expensive algorithms in other programming languages like C++ and CUDA. It has its own solution to deal with the overhead. Instead of switching to other programming languages each time when it meets an expensive operation, TensorFlow runs entirely outside Python environment. Programmers use Python to structure the machine learning algorithms and TensorFlow will take over and run this algorithm in a more efficient method. Although the overhead is still inevitable, TensorFlow can dramatically increase the performance by avoid doing switching each time when expensive operations appears.

III. Implementation and Result in TensorFlow

One of the most important steps of building a neural network is to properly prepare the training input datasets. As it is presented in 2, the AECG mixture signal will be passed into a FIFO to be buffered. For each iteration, a new data is inserted into the buffer at the beginning of the queue, the original data in the buffer will be shift towards the end of list by one. The data at the end of the list will be discarded. The data in the buffer will be used as the input of the neural network. By examine a single FECG and MECG signal, the buffer size is set to 30. After the data passed through the trained neural network, it is supposed to detect if there is a FECG or MECG signal in the AECG signal.

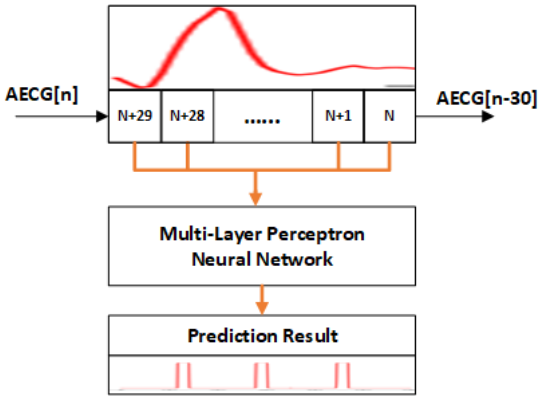

Figure 2. Block diagram of the training system

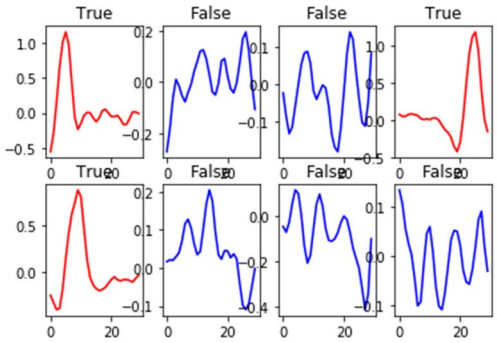

Figure 3. MECG Training Inputs

To validate the idea of using MLP neural network for identifying FECG signal. A simple task to detect MECG signal from AECG signal is implemented to show capability of the neural network. Figure 3 presents a random collection of 8 input datasets for training the neural network. Signal marked in red indicate that the current frame is a MECG signal and vice versa. Because MECG has a very significant amplitude variation, the neural network should be able to identify it very nicely without any problem.

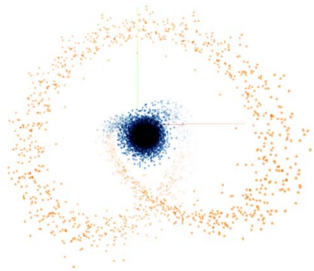

Figure 4. Embedding Projector for MECG Training Input

For the instinct that most of the input dataset for the neural network are in high dimension, we need an intuitive way to interpret the input data. TensorFlow offers a very useful tool called embedding projector. One of the method to project the high dimension data into lower dimension is by using Principle Component Analysis (PCA) algorithm. PCA algorithm in TensorFlow will compute the top 10 principle component as basis and allow users to choose two or three to project the high dimension data into 2D or 3D space. Since the buffer size is set to 30, our input data will be in the dimension of 30. Figure 4 shows the sliced AECG signal presented in a 3D space. MECG signal is marked in orange, and non-MECG is marked in blue.

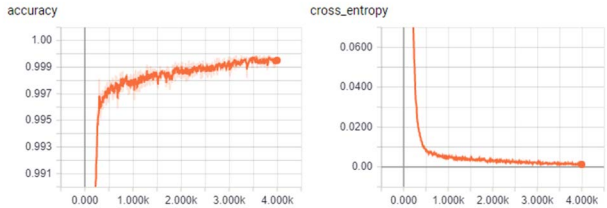

Figure 5. MECG Detection Learning Procedure

After training the neural network with 10000 sets of training input for 4000 iterations, the accuracy converges to almost one. The training procedure takes about 52.3 second when GPU support is turned on. Figure 5a and 5b are the accuracy and cross entropy compare to the training iterations plotted in Tensor Board. The training converges easily after 1000 iterations. And the training accuracy is more than 99%.

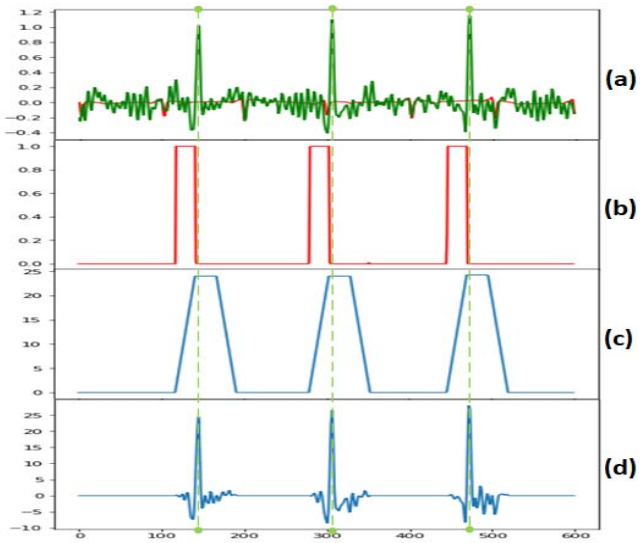

Figure 6. MECG Detection Result

Good training accuracy doesn't mean that he neural network is going to be effective on certain task. Figure 6a is the AECG signal in green and FECG signal in red. Figure 6b is the training output plotted in python using matplotlib. Figure 6c is obtained by extending all the points of the training result to 30 samples corresponding to its original position in the AECG signal and stack them together. The result shows that this algorithm can precisely detect the MECG signal. Figure 6d is obtained by multiply AECG signal in Figure 6a and Figure 6c.

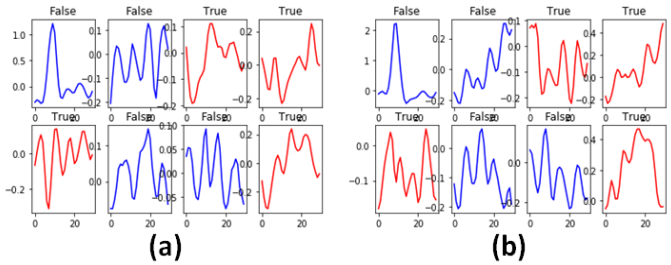

Figure 7. FECG Training Inputs

After successfully detect the MECG signal with one channel of AECG input, the following part of this section is to discuss the feasibility of detecting FECG signal with AECG signal using neural network. Different from MECG detection, FECG signal is hidden in the MECG signal and the noise. Figure 7 shows 8 random selected training inputs from channel 1 AECG signal plotted in (a) and channel 2 plotted in (b). The signal is marked in red when the frame contains FECG signal. By training the neural network with channel 1 AECG signal, the training result is not ideal. So that channel two is also used for training the neural network. Figure 7a shows the training input with channel one plotted in embedding projector. Figure 7b shows the training input plotted in embedded projector with two channels of AECG signal, the training input is in 60 dimensions. Training input with FECG is marked in orange. Data points marked in blue means that it is a non-FECG frame.

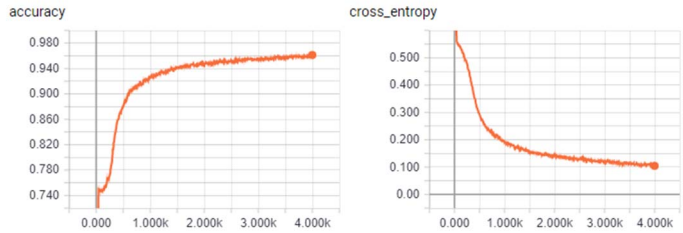

Figure 8. Learning Procedure of FECG Detection with Single Channel AECG Signal

Training the neural network with 10000 training input from channel 1 for 4000 iterations takes 48.2 seconds. The training accuracy converge to 0.962 at the end of the training. Figure 8 is the accuracy and cross entropy compare to the training iterations plotted in Tensor Board.

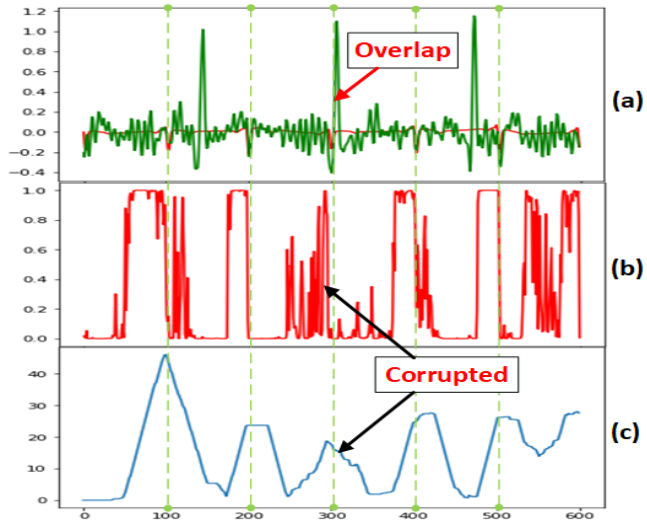

Figure 9. FECG Detection Result with Single Channel AECG Signal

Figure 9 is the training result obtained using the same procedure in Figure 6. Figure 9a indicates that the third FECG signal is overlapped with the second MECG signal. Figure 9b is the output of forward propagation by passing test data into the trained neural network. Figure 9c is the post-processed label by expanding the training output. The validation result shows that using one channel of mixture signal can roughly detect FECG signal in AECG. However, this result is not very well compared to MECG detection. To increase the accuracy of the neural network, one more channel of AECG signal is used as the training input.

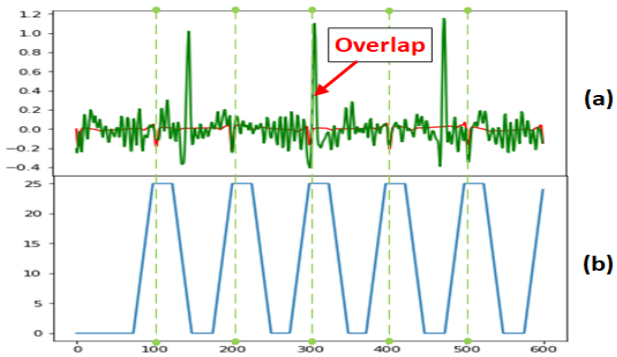

Figure 10. FECG Detection Result with Two Channels of AECG Signals

By using channel 1 and channel 2 together as the input, the neural network is again trained for 4000 iterations with 10000 of training examples. The training input is in 60 dimensions which means that each training input contains 60 data points. The training procedure takes 53.9 seconds and the training accuracy reached to almost 1 after 4000 iterations. Figure 10b is the expanded training output using training input from both channel 1 and 2. This result shows that the neural network can easily detect hidden FECG signal by using two channels of AECG signal.

IV. Conclusion

In this paper, an MLP neural network algorithm is presented to detect the MECG and FECG signal from the mixture signal acquired using non-invasive sensors. To test the algorithm, an opensource FECG signal simulator FECGSYN is used to generate the mixture signal for training the neural network. This algorithm successfully characterizes MECG and FECG with a single channel AECG mixture signal. Also, with two channels of AECG signals, the neural network can detect FECG signal very accurately.

In the future, an embedded system can be designed with ECG sensors and necessary processing power to detect Fetal heart rate using AECG signal. Also, we will target on using neural network for cardiac disease detection. This will help physicians to make diagnose efficiently.

References

[1] Solum, Thore & Ingemarsson, Ingemar & Nygren, Åke. (1980). The accuracy of abdominal ECG for fetal electronic monitoring. Journal of perinatal medicine. 8. 142-9. 10.1515/jpme.8.3.142, 1980.

[2] C. S. Lee, M. Masek, C. P. Lam and K. T. Tan, "Towards higher accuracy and better noise-tolerance for fetal heart rate monitoring using Doppler ultrasound," TENCON 2009 - 2009 IEEE Region 10 Conference, Singapore, pp. 1-6, 2009.

[3] D. Rassi, "Fetal heart monitoring by SQUID magnetometry," IEE Colloquium on Biomagnetic Methods in Medicine, London, pp. 8/1-8/3, 1993.

[4] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al., "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," arXiv preprint arXiv:1603.04467, 2016.

[5] Behar J., Andreotti F., Zaunseder S., Li Q., Oster J. and Clifford G D., An ECG model for simulating maternal-foetal activity mixtures on abdominal ECG recordings. Physiol Meas 35(8), pp.1537-50, 2014

[6] Andreotti F., Behar J., Zaunseder S.,Oster J. and Clifford G D., An Open-Source Framework for Stress-Testing Non-Invasive Foetal ECG Extraction Algorithms. Physiol Meas 37(5), pp. 627-648, 2016.

[7] B. Wang and J. Saniie, "Ultrasonic flaw detection based on temporal and spectral signals applied to neural network," 2017 IEEE International Ultrasonics Symposium (IUS), Washington, DC, pp. 1-4, 2017.

[8] B. Wang and J. Saniie, "Ultrasonic target echo detection using neural network," 2017 IEEE International Conference on Electro Information Technology (EIT), Lincoln, NE, pp. 286-290, 2017.