PPG-based Personalized Verification System

- PPSNet -

Dae Yon Hwang
Electrical & Computer Engineering
University of Toronto
Toronto, Canada
dy.hwang@mail.utoronto.ca

Dimitrios Hatzinakos
Electrical & Computer Engineering
University of Toronto
Toronto, Canada
dimitris@comm.utoronto.ca

***Abstract*— Convergence between online and off-line systems gives us a great chance to enrich our societies but it also requires a high secure system to verify true user from fraud. In this paper, we propose a novel deep learning-based verification model using Photoplethysmography (PPG) signals. The goal of this paper is to build a personalized data-driven network by employing convolution neural network (CNN) with long-short term memory (LSTM), to model the time-series sequence inherent within the PPG signal. After building each personalized network, each network can be applied to distinguish a true user from others. The proposed network was evaluated on the BioSec. Lab PPG dataset at University of Toronto, which achieved an average of 10-fold cross-validation accuracy of 96% (in single-session) and 72.7% (in two-sessions).**

Keywords*— *PPG, CNN, LSTM, Machine Learning, Verification

I. Introduction

Today, various online and off-line systems are mixed, and they give great advantages in our life. We naturally use them in many ways, such as online shopping, mobile game, watching movies, and buying stocks. With the potential risk, our money should flow through online network. It is obvious that there can be big drawbacks correlated with such advantages (ex. fraud, spoofing, and various attacks from hackers). Thus, the current system needs to be made more secure.

Recently, many people focus on the usage of physiological signals to get information about biometric authentication, heart rate and blood pressure calculation. The examples of physiological signals are electrocardiograms (ECG) [1] and photo-plethysmography (PPG) [2]. Although ECG has been shown to be more accurate than PPG in terms of identification, the cost of the device used for PPG measurement is lower and the device has better accessibility and portability. In addition, PPG signals can be easily acquired from various positions such as earlobes, fingertips or wrist. For these reasons, PPG signals are considered to build a secure system to protect user information from attacks.

The goal of this paper is to build PPG-based Personalized Verification System, *PPSNet*, which is mainly for user verification. Furthermore, it can be extended to verify user from others when we are doing E-transaction or using blockchain techniques. It has been cleared that different person has different characteristics of PPG signal and this is a great advantage to build a personalized system.

The main point of this paper is to utilize Deep Neural Network (DNN) to build a completely data-driven approach based on convolution neural network (CNN) and long-short term memory (LSTM). Since these networks make *PPSNet* learn the temporal biological features of each subject, feature selection and extraction process involved in popularly classification schemes are no longer required.

II. Related Works

These days, many research groups are trying to apply deep learning methods on physiological signals to detect specific target from others. There are many trials and good results in ECG area. In [3], heartbeat datasets were classified into atrial fibrillation rhythms, normal rhythms, and other rhythms. It shows that ensemble of 5 network with CNN and LSTM gave interesting classification result which was 79.2% accuracy without data augmentation. Also, Heart Rate Variability (HRV) from ECG was used to detect the diabetes [4]. CNN-LSTM combination was applied and it yielded 90.9% accuracy.

As ECG, there are also many recent papers that used deep learning methods in PPG signals. One recent work [5] had focused on a two-stage procedure involving clustering (with 11 hand-crafted features) and deep learning models (Restricted Boltzmann Machines and Deep Belief Networks). Average accuracy was 96.1% which seems interesting to pave the way for future research. In [6], atrial fibrillation was detected from PPG datasets by using CNN. Wavelet power spectrum of PPG signal was utilized to classify each patient into one of two classes of atrial fibrillation or normal. According to [7], a network with CNN and LSTM was built to classify one user from others. Two CNN in conjunction with two long short-term memory layers were employed for modeling the time-sequence inherent within PPG signal. This group classified one subject from 11 subjects, and it gave 96% accuracy.

From these papers, many research groups adopted CNN in conjunction with LSTM to learn the discriminant features from the physiological signals. Thus, we use a similar network structure for *PPSNet* to build a personalized verification system.

III. Data

Database is obtained from BioSec. Lab at University of Toronto. Every PPG signal was collected with Plux pulse sensor.

978-1-7281-0319-8/19/$31.00 ©2019 IEEE

It has both LED and Photo Diode on same side, thus it is reflective type of PPG sensor. Sampling rate for all recording was fixed at 100 Hz. In this paper, we consider two cases which are single-session and two-sessions. In single-session, PPG signals were recorded in relax condition for 3 minutes from fingertip using sensor. In two-sessions, 3 minutes PPG signals were recorded from each subject over two times within at least 2 weeks of time differences. For both cases, we utilized 20 subjects to train and test our proposed network. More details about data will be covered in Sec. V.

IV. Methods

A. *Convolution Neural Networks (CNN) [8]*

CNN is a deep artificial neural network that is applied primarily to classify images, cluster them by similarity, and perform object recognition within scenes. This neural network can identify faces, street signs, tumors, and also apply for text analytics. Recently, many people tried to adopt CNN on classification task in physiological signal because of its powerful ability of automatic feature extraction. Usually, CNNs are composed of an initial layer of convolutional filters, followed by non-linearity, sub-sampling, and regularization. There are various hyperparameters that should be carefully tuned. In this paper, we only show the best hyperparameters from exhaustive searching. Details about hyperparameters in CNN will be shown in Sec. V.

B. *Long-Short Term Memory (LSTM) [9]*

A drawback of using the CNN is that the generated features are not completely phase invariant. Depending on the start point of the heartbeat in PPG signal, the relevant features will be changed. Further, data division which was conducted on preprocessing step can also cause phase variant, but CNN does not consider this problem. To solve this issue, we used Recurrent Neural Network (RNN), especially LSTM, which works well at understanding the sequence of historical local trends of the data.

Here, two hyperparameters are controlled: the number of layers and the hidden units. As in CNN, there is no exact criteria to select the hyperparameters. Thus, we chose the best sets from exhaustive searching which is covered in more details in Sec. V.

C. *Bagging [10]*

Bagging is one of the famous ensemble method that is typically applied to reduce the variance, while maintaining the bias. It is possible because we implement average predictions in different subsets of input dataset. If each single classifier from subsets is unstable which means high variance, the averaged classifier has the smaller variance than a single original one. To apply bagging, we need to sample the input data, with replacement, to generate multiple sets of input data. For each of those sets, the same structure of model will be used for training. To predict the result on an unseen test data, data goes through these individual models and the predictions are averaged to get the final decision.

V. Algorithms

In this section, the structure of the proposed network is introduced. Then, each step is covered in more details.

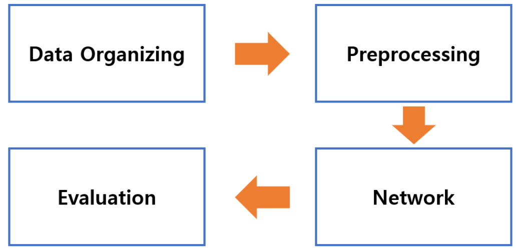

Fig. 1. The structure of Algorithms

Figure 1 shows the general structure of the proposed algorithm in this paper. As mentioned before, the main point of this paper is the feature extraction stage to build a personalized verification system. First of all, in the data organizing stage, data needs to be organized for training and testing. After that, preprocessing stage uses the organized data to remove the noise and prepare it before applying to the network. At network, CNN and LSTM are used to construct a network and it is applied to implement feature extraction to distinguish user which is the goal of this paper. Finally, the result is evaluated by several measurements.

A. *Data Organizing*

In this paper, data was collected from 20 subjects for two different scenarios. First case is the single-session training and testing. The data for training and testing are from one session without overlapping. With training data, a network is trained to find useful features, and it is evaluated on the test data. [5], [6] and [7] are all focused on this single-session and thus, *PPSNet* is expected to give great results in single-session case.

Second case is the two-sessions training and testing. The data for training comes from first session, and the data for testing is generated from second session. Features are extracted from training data, and they are utilized to distinguish certain user in test data. PPG is based on heartbeat which can be easily affected by stress, exercise, diet and others. For this reason, even if same person measures PPG signal, each session can show different characteristics. Thus, it is challenging to find time-stable features. To our best knowledge, no other groups have published research about feature extraction by deep learning methods in two-sessions, thus this paper can be the stepping stone for future research.

B. *Preprocessing*

Preprocessing is the important step when we consider machine learning models because they are sensitive to noise, translation variance, size variance and others. For preprocessing stage, filtering, dividing, and normalization are applied.

The 4th order of Butterworth filter was used to restrict the high frequency noise components and find the signal of interest. Cut-off frequency for filter is 0.1-18 Hz.

In single-session, after filtering, each subject was divided to have the same number of samples. Each subject has about 18,000 samples and they were divided into 18 pieces, which means each piece contained 1,000 samples (in other words, the number of features is 1,000). The reason for such division is that the data should have consistent length to be applied in a network. Since there are 20 subjects with 18 data, the number of total dataset after preprocessing in single-session is 360. Finally, we normalized each divided data with zero mean and unit variance.

In two-sessions, the ways for preprocessing are almost the same as in single-session. The difference is, in two-sessions, some subjects from the second session contain big oscillations in first and last part of data which seems to be caused by a problem when measurement was started and finished (ex. wave hands, or move around). For this reason, we excluded first and last divided data in all subjects when we built test data from the second session. In other words, there are 18 datasets for each subject in training set and 16 datasets for each subject in test set. After preprocessing, the number of training data is 360 and the number of test data is 320. Same as before, we normalized each divided data with zero mean and unit variance. Figure 2 shows example of one subject, depending on preprocessing, in single-session.

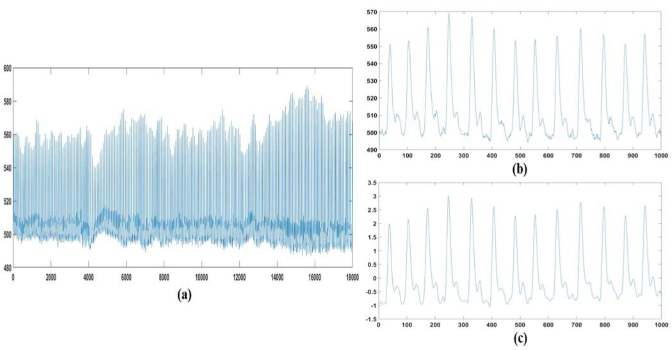

Fig. 2. Visualization of data, depending on preprocessing (One subject in single-session). (a) Original data before preprocessing. (b) Cropped data from (a) without preprocessing. (c) After preprocessing from (b). For one subject in single-session, there are 18 preprocessed datasets as (c). In all figures, X-axis is the number of samples and Y-axis is magnitude.

C. Network

In this paper, 1D data (PPG signal) was utilized, meaning 1D CNNs with LSTMs are required to build a network. In both cases, 10-fold cross-validation is used for training and validating the network. Cross-validation is useful to control the bias and variance of models since most of data are used for fitting and validating. To prevent overfitting, early stopping method is considered. Also, weighted loss is applied to offset the class imbalance (distinguish one user from others, which are 1 versus 19). For both sessions, Rectified Linear Units (RELU) is selected as activation function in CNN and Root Mean Square Propagation optimizer is used to optimize our network.

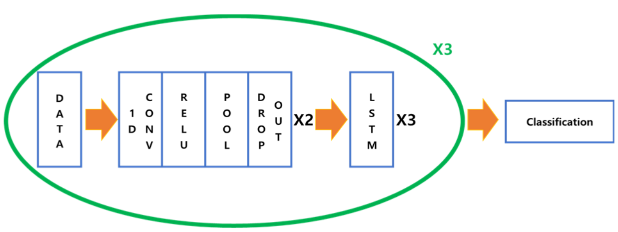

Fig. 3. Single-session Network (3 bagging models consisted of 2 CNN and 3 LSTM layers)

Figure 3 shows the network for single-session. Here, black bold number means the number of layers and green bold number shows the number of bagging models. After preprocessing, 75% of datasets was applied for training and 25% of datasets was for testing. Preprocessed data goes through 2 layers of CNN which consists of convolution, RELU, maxpooling, and dropout. Hyperparameters for CNN are as follows: the number of filters=40 (both layers), the size of filters=30 (first layer), 50 (second layer), the size of pool=4 (both layers), the amount of dropout=50% (both layers). After CNN, 3 layers of LSTM are applied to learn the time-series data. The number of hidden units for each LSTM is 60. As mentioned before, there is no criteria for setting these hyperparameters and thus, we selected the best one that gave high accuracy with low equal error rate (EER).

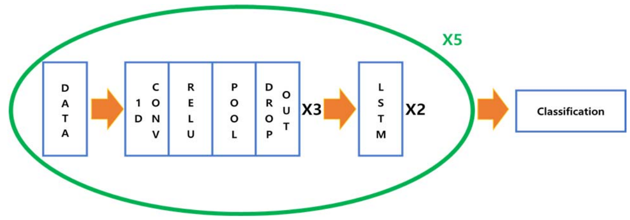

Fig. 4. Two-sessions Network (5 bagging models consisted of 3 CNN and 2 LSTM layers)

Figure 4 introduces the network applied in two-sessions. The network structures for single-session and two-sessions are similar but they have different number of bagging models, layers and hyperparameters. After preprocessing stage, data from first session is used for training while data from second session is applied for testing. There are 3 layers of CNN to learn time-stable features. Hyperparameters used for CNN are as follows: the number of filters=60 (all layers), the size of filters=30 (first layer), 50 (second layer), 70 (third layer), the size of pool=2 (first and second layers), 4 (third layer), the amount of dropout=50% (all layers). After passing CNN, 2 layers of LSTM are used with 50 hidden units for each LSTM. From several trials, these hyperparameters showed the best results in terms of accuracy and EER.

VI. RESULTS

We considered several measurement methods to evaluate the results: accuracy, equal error rate (EER), and execution time. The accuracy comes from the true predictions divided by total number of datasets. EER is the point when false rejection rate and false acceptance rate are same. Thus, it is a good indicator for measuring system performance. In this paper, the network was trained on Tensorflow 1.2.0 with Nvidia Geforce 940MX. This GPU is not high quality, thus execution time has room for improvement.

TABLE I. SINGLE-SESSION VERIFICATION RESULT

Verification Result in Single-Session	
Average of Training data Accuracy	99.8%
Average of Training data EER	0.05%
Average of Training Time	1821 seconds
Average of Test data Accuracy	96%
Average of Test data EER	3.6%
Average of Test Time	25 seconds

Table I shows the average verification results in single-session. After each network training is finished, we applied the whole training data (or test data) into the network to calculate each network's accuracy and EER. The results of each trained personalized network are averaged and we called them as verification results (Table I). The verification result of the training set is almost perfect. When looking at the test set, the result is also promising, comparing to recent paper [7], even though we considered more subjects. Above all, EER is very low which means our single-session system is highly secure with accurate recognition of user. Average training time is around 30 minutes which seems not very long for practical application and average of test time is very short which is 25 seconds.

TABLE II. COMPARE BETWEEN *PPSNET* AND OTHERS

Compare Verification Result in Single-Session			
Methods	[5]	[7]	*PPSNet*
Number of Subjects	11	12	20
Models	Restricted Boltzmann, Deep Belief Network	CNN, LSTM	CNN, LSTM
Verification Scheme	Classify target inside each cluster	1 vs 11	1 vs 19
Accuracy	96.1%	96%	96%
Sensitivity	96.1%	84%	96.4%

Table II compares the verification results among PPSNet and state of the art works who used deep learning verification models with PPG. Compared to [5] which did verification inside each cluster, our model shows similar results, even though PPSNet classified target from others without clustering. Furthermore, PPSNet has better accuracy and sensitivity than [7] which shows similar network structure as ours.

TABLE III. TWO-SESSIONS VERIFICATION RESULT

Verification Result in Two-Sessions	
Average of Training data Accuracy	99.9%
Average of Training data EER	0.005%
Average of Training Time	2743 seconds
Average of Test data Accuracy	72.7%
Average of Test data EER	30.2%
Average of Test Time	75 seconds

Table III explains the average verification results in two-sessions. From our searching, there is no published paper that experiments the verification performance in two-sessions, thus this can be valuable to pave the way for further research. We assumed that *PPSNet* can learn time-stable features to distinguish true user on test data from inaccessible session. For training data, the average accuracy and EER is good as single-session. When we consider test data, the average accuracy is quite low and the average EER is considerably high, comparing to single-session. However, this experiment shows the possibility that deep learning model can learn time-stable features to find target on test data which was never been shown. Compared to single-session, average of training time increases about 15 minutes because of more bagging models and CNN layer. Although two-sessions network requires quite more training time, average of test time is short which is 75 seconds.

VII. DISCUSSION

In this paper, the network constructed by CNN and LSTM yields promising results to remove the need for extracting hand-crafted features for biometric verification. Furthermore, we build a personalized data-driven network for each individual that can be extended to apply for verification in highly secure systems. In single-session, our network shows 96% average accuracy with 3.6% average EER on test data. In addition, in two-sessions, *PPSNet* yields 72.7% average accuracy with 30.2% average EER which shows possibility for expansion on multi-session classification by deep learning methods.

Future explorations would be applying more CNN layers to improve the performance in two-sessions. Deeper network can learn more various information which can be useful to distinguish true target from others. Also, we will build identification system that can discriminate all users with single network.

VIII. ACKNOWLEDGMENT

This work was funded by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the Royal Bank of Canada (RBC).

REFERENCES

[1] A. Lourenc ̧o, H. Silva, and A. Fred, "Unveiling the Biometric Potential of Finger-Based ECG Signals" Computational intelligence and neuroscience, vol. 2011, p. 5, 2011.

[2] Y. Y. Gu, Y. Zhang and Y. T. Zhang, "A novel biometric approach in human verification by photoplethysmographic signals," 4th International IEEE EMBS Special Topic Conference on Information Technology Applications in Biomedicine, 2003., Birmingham, UK, 2003, pp. 13-14.

[3] M. Zihlmann, D. Perekrestenko and M. Tschannen, "Convolutional recurrent neural networks for electrocardiogram classification," 2017 Computing in Cardiology, Rennes, 2017, pp. 1-4.

[4] Swapna G, Soman KP, Vinayakumar R, "Automated detection of diabetes using CNN and CNN-LSTM network and heart rate signals" International Conference on Computational Intelligence and Data Science, 2018

[5] Jindal, Vasu, et al. "An adaptive deep learning approach for PPG-based identification." Engineering in Medicine and Biology Society, 2016 IEEE 38th Annual International Conference of the. IEEE, 2016.

[6] S. P. Shashikumar, A. J. Shah, Q. Li, G. D. Clifford and S. Nemati, "A deep learning approach to monitoring and detecting atrial fibrillation using wearable technology," 2017 IEEE EMBS International Conference on Biomedical & Health Informatics, Orlando, FL, 2017, pp. 141-144.

[7] L. Everson et al., "BiometricNet: Deep Learning based Biometric Identification using Wrist-Worn PPG," 2018 IEEE International Symposium on Circuits and Systems, Florence, Italy, 2018, pp. 1-5.

[8] D. Karunakaran, "Simple Image classification using deep learning - deep learning series 2," *medium.com*, 04-May-2018. [Online]. Available: https://medium.com/intro-to-artificial-intelligence/simple-image-classification-using-deep-learning-deep-learning-series-2-5e5b89e97926. [Accessed: 30-Jan-2019].

[9] C. Olah, "Understanding LSTM Networks," *Understanding LSTM Networks*. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/. [Accessed: 30-Jan-2019].

[10] SeattleDataGuy, "Boosting and Bagging: How To Develop A Robust Machine Learning Algorithm," *Hacker Noon*, 21-Nov-2017. [Online]. Available: https://hackernoon.com/how-to-develop-a-robust-algorithm-c38e08f32201. [Accessed: 30-Jan-2019]