# ChaLearn LAP Real Versus Fake Expressed Emotion Challenge @ICCV 2017

July 2017

## 1 Team details

- Team name: HCILab
- Team leader name: Huynh Xuan Phung
- Team leader address: Department of Computer Engineering, Sejong University, Seoul, South Korea , phone: +821028423785 and email:phunghx@gmail.com
- Rest of the team members: Yong-Guk Kim
- Team website URL: http://ce.sejong.ac.kr/~ykim/
- Affiliation: Sejong University

## 2 Contribution details

- Title of the contribution: Long short term memory network with Parametric Bias and its Application to Real versus Fake Emotion Recognition
- Final score: 66.7%
- General method description: Tani et al., 2004 proposed the RNNPB which based on the properties of mirror neurons in the rostral part of inferior area 6 of macaque monkeys. This model is used in a robot experiment. We changed the RNN by LSTM (long-short-term memory) that we call LSTM-PB network. Also, we used gradient boosting machine to predict the score of real/fake emotion. To train LSTM-PB, we use the facial landmarks.

## • References

[1] Goodfellow, Ian and Bengio, Yoshua and Courville, Aaron. *Deep learning*. MIT press, 2016.

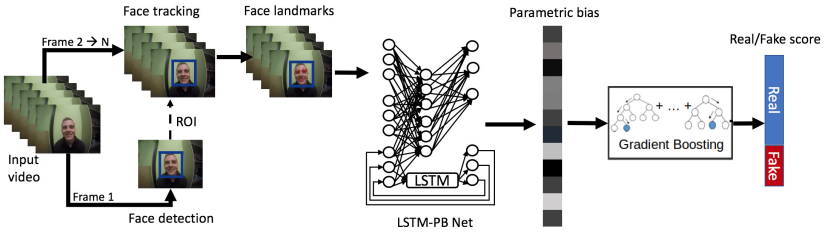


Figure 1: The framework of our proposed method.

[2] Tani, Jun and Ito, Masato and Sugita, Yuuya. *Self-organization of distributedly represented multiple behavior schemata in a mirror system: reviews of robot experiments using RNNPB*. Neural Networks, Journal, Elsevier, 2004.

[3] Chen, Tianqi and Guestrin, Carlos. *Xgboost: A scalable tree boosting system*. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016.

[4] Bolme, David S and Beveridge, J Ross and Draper, Bruce A and Lui, Yui Man. *Visual object tracking using adaptive correlation filters*. Computer Vision and Pattern Recognition (CVPR), 2010.

[5] King, Davis. *Dlib c++ library*. http://dlib.net.

[6] Bradski, Gary. *The OpenCV Library*. Dr. Dobb's Journal: Software Tools for the Professional Programmer, Miller Freeman Inc., 2000.

[7] R. Collobert. Torch. *NIPS Workshop on Machine Learning Open Source Software,*. 2008.

- Figure 1 illustrates the diagram of our approach
- Describe data preprocessing techniques applied: Our experiment is to test our method on the face region. To deal with this problem, the preprocessing technique performs the following steps. First, we decoded video into a sequence of frames. Second, Haar Cascade algorithm detects a face on the first frame. Finally, MOSSE (Minimum Output Sum of Squared Error) tracker follows this ROI on the remained frames.

# 3 Recognition of fake and true emotions

## 3.1 Features / Data representation

Describe features used or data representation model: this is the facial landmarks

### 3.2 Dimensionality reduction

Dimensionality reduction technique applied: we obtained the 68 landmarks from a face. Then we reduced to 40 key points. These points are covered in detail the main components of a face, such as eyes, eyebrows, nose, mouth, and cheeks.

### 3.3 Compositional model

Compositional model used, i.e. pictorial structure: None

### 3.4 Learning strategy

First, we learn LSTM-PB network for each facial expression. There are six emotions: anger, contentment, disgust, happiness, sadness, and surprise with six different networks, respectively. We adopt a two-stage of training procedure: first, we learn the optimal weights of LSTM-PB network for predicting the landmarks on the next frame by a back-propagation algorithm. After we finish all frames in a video, we next learn the optimal values of parametric bias by accumulating gradients of the previous stage. We use a stochastic gradient descent algorithm with an adaptive learning rate to optimize the weights of LSTM-PB. The learning rate is 0.01 and decay of 0.9 after 10 epoch that isn't improve the performance. Parametric bias is updated with a scale of 0.9. Each epoch learns 80 videos, and we stop training after 100 epochs. After we have six networks, we generated the parametric bias vector for each video. Our method also used gradient boosting to train a Real/Fake discrimination in parametric bias space. We compute model selection using 5-fold cross-validation to select the best gradient boosting on source domain. The gradient boosting has 5000 trees.

### 3.5 Other techniques

Other technique/strategy used not included in previous items: testing technique. To improve the performance, we find the pairs of videos that show the same subject with each facial expression. The structural similarity of the face region in the first frame between videos detects two videos are depended on a subject or not. Since a video expressed the genuine emotion and another is fake emotion, we assign a higher bossing score for fake emotion while the rest one for real expression.

## 4 Global Method Description

- Total method complexity: LSTM-PB has total 115792 parameters with 4 layers of Grid-LSTM while xgboost has 5000 ensemble trees.
- Pre-trained or external methods have been used: dlib library for facial landmarks detection and OpenCV Haar Cascade face detection.

- Which additional data has been used in addition to the provided training and validation data (at any stage, if any): None
- Qualitative advantages of the proposed solution: From the perspective of computational neuroscience, our proposed method is potential for explaining the theory of mirror neurons. From the viewpoint of computer vision, our framework can recognize a video that has a variety of number frames.
- Results of the comparison to other approaches. We also implemented a 3DCNN network to classify real and fake emotion. The results are same as the random case that is about 53%.
- Novelty degree of the solution and if is has been previously published: Not yet compared.

# 5 Other details

- Language and implementation details (including platform, memory, parallelization requirements):Ubuntu 16.04 64bit, CPU core i7, 32 Gb RAM, GPU NVIDIA Titan X 12 Gb. We used three languages in our experiment: C++, python, and Lua.
- Detailed list of prerequisites for compilation: g++ 5.4, cuda 8.0, torch 7.1, python 2.7, opencv 2.4.9, xgboost 0.16, sklearn 0.18, scikit-image, dlib 19.04
- Human effort required for implementation, training and validation: detect face manually if the OpenCV library can not find the face.
- Training/testing expended time: training time is about two days while testing time for 60 videos is 15 minutes.
- General comments and impressions of the challenge: This challenge provides a great opportunity for researchers to approach the real-world application.