Hybrid Message Passing with Performance-driven Structures for Facial Action Unit Detection Supplementary Material

Tengfei Song[1], Zijun Cui[2], Wenming Zheng[1], and Qiang Ji[2]
[1]Southeast University
[2]Rensselaer Polytechnic Institute
songtf@seu.edu.cn, cuiz3@rpi.edu, wenming_zheng@seu.edu.cn, qji@ecse.rpi.edu

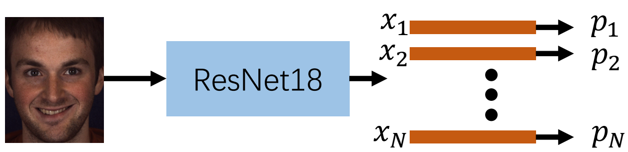

Figure 7. The framework to generate the AU features by using ResNet18.

1. The Generation of AU features

In this part, we introduce the details about how to extract disentangled AU features, which are further fed into message passing neural network. As shown in Figure 7, we use the ResNet18 to extract the disentangled AU features. $X = x_1, ..., x_N \in \mathcal{R}^{d\times N}$ are N features corresponding for N AUs. In this paper, d is set to 512. $[x_1, ..., x_N]$ are reshaped by the output of ResNet18. Each feature vector is fed into a fully connected layer to predict the probability of the corresponding AU. The predicted probability for the ith AU can be calculated by

$$p_i = \sigma(w_i^T x_i + b_i), \tag{13}$$

in which σ is the sigmoid function. w_i and b_i are the parameter of the ith full connection layer. To alleviate the influence of the imbalanced data. We calculated the weighted loss function to train the model and the loss function can be written as

$$\mathcal{L} = -\sum_{i=1}^{N} \tau_i [y_i log\ p_i + (1 - y_i) log(1 - p_i)], \tag{14}$$

in which y_i denote the ground truth and τ_i are constant weight. We directly use τ_i provided in [1].

2. Pseudo Code to Sample Multiple Graph Structures

Algorithm 1 Obtain multiple graph structures $\{\mathbf{A}^n\}_{n=1}^N$ by Performance-based MCMC sampling

Require: Training data $\mathcal{D}$
Ensure: N samples $\{\mathbf{A}_n\}_{n=1}^N$
Initial graph structure $\mathbf{A}^1$;
Fully train the neural network and estimate the performance, $i.e.$, acc($\mathbf{A}_1$, w).
for n = 1 to N **:**
  Generate a new sample $\mathbf{A}_{n+1}$ based on proposal probability Equ (1);
  Fully train the neural network and estimate the performance, $i.e.$, acc($\mathbf{A}_{n+1}$, w).
  Calculate the acceptance probability $p^{acc}(\mathbf{A}_{n+1})$ by Equ (4);
  if accept:
    $\mathbf{A}_{n+1} = \mathbf{A}_{n+1}$;
  else:
    $\mathbf{A}_{n+1} = \mathbf{A}_n$;
  end
end

References

[1] Zhiwen Shao, Zhilei Liu, Jianfei Cai, and Lizhuang Ma. Deep adaptive attention for joint facial action unit detection and face alignment. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 705–720, 2018. 1