Audio-Driven Emotional Video Portraits
(Supplementary Material)

Xinya Ji[1] Hang Zhou[2] Kaisiyuan Wang[3] Wayne Wu[4]* Chen Change Loy[5]
Xun Cao[1]* Feng Xu[6]*
[1]Nanjing University, [2]The Chinese University of Hong Kong,
[3]The University of Sydney, [4]SenseTime Research,
[5]S-Lab, Nanyang Technological University, [6]BNRist and school of software, Tsinghua University
{xinya@smail., caoxun@}nju.edu.cn, zhouhang@link.cuhk.edu.hk, ccloy@ntu.edu.sg,
kaisiyuan.wang@sydney.edu.au, wuwenyan@sensetime.com, xufeng2003@gmail.com

In this supplementary material, we present more information on our architecture as well as implementation details and more qualitative results from our experiments. We strongly recommend watching the supplementary video where more animation results and comparisons are shown.

1. Network Architecture

The two components of our *EVP* algorithm have been briefly introduced in Section 3 of the main paper. Here we provide more details of the network architecture.

1.1. Cross-Reconstructed Emotion Disentanglement

The two encoders E_c and E_e in Sec.3.2 are composed of convolutional neural networks(CNN) followed by multilayer perceptrons(MLP). They extract the content and emotion features of the input audio clips separately. Moreover, we set the channel size of the content audio embedding $E_c(x)$ and the emotion audio representation $E_e(x)$ to be 256, 128 based on our experiments.

1.2. Target-Adaptive Face Synthesis

Audio-to-landmark Module. Then channel size of the identity embedding f_a in the audio-to-landmark network is 256. Here we use the long short-term memory (LSTM) network to predict the landmark displacement l_d since it can capture sequential relationships between audio signals and landmark animations.

3D-Aware Keypoint Alignment. The parametric 3D face model [1] here recovers low-dimensional pose $\boldsymbol{p} \in \mathbb{R}^6$, geometry $\boldsymbol{g} \in \mathbb{R}^{199}$ and expression parameters $\boldsymbol{e} \in \mathbb{R}^{29}$ for each pair of predicted landmarks and detected ones by solving a non-linear optimization problem. We obtain the pose-invariant 3D landmarks $\boldsymbol{L}_p^{3d}$ from the geometry and expression parameters:

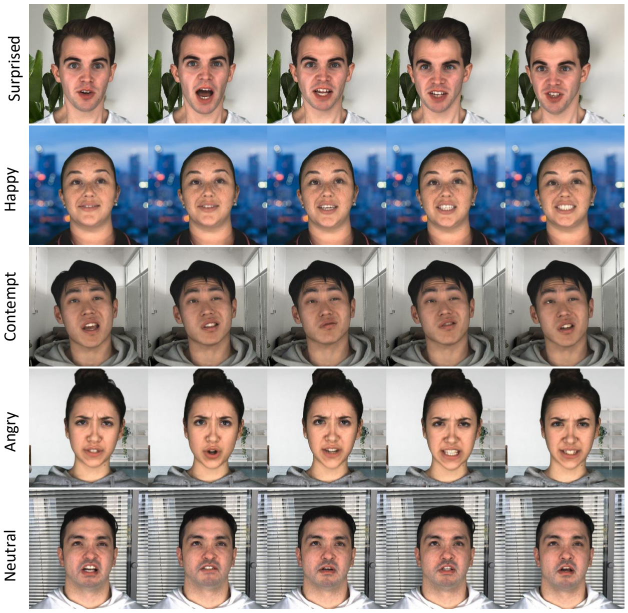

Figure 1: **Qualitative results of EVP.** Each row shows characters with different emotions listed on the leaf side.

$$\boldsymbol{L}_p^{3d} = \boldsymbol{m} + \sum_{k=1}^{199} \boldsymbol{g}_k \boldsymbol{b}_k^{geo} + \sum_{k=1}^{29} \boldsymbol{e}_k \boldsymbol{b}_k^{exp}. \tag{1}$$

where $\boldsymbol{m} \in \mathbb{R}^3$ is the average facial landmark positions, $\boldsymbol{b}_k^{geo}$ and $\boldsymbol{b}_k^{exp}$ are geometry and expression basis computed via principal component analysis (PCA) on high-quality facial scans and blendshapes.

Edge-to-Video Translation Network. Following [3], we adopt a conditional-GAN architecture for our edge-to-video translation network. The generator G is designed in a coarse-to-fine manner [4], aiming to transfer the predicted motion to the target video frames under the guidance of the edge map. Se can get the reconstruction loss:

$$L_{recon} = \|\hat{x}_t - G(x_t|e_t)\|_2. \tag{2}$$

where $\hat{x}_t$ is the generated frame, x_t is the target frame and e_t is the edge map. In terms of the discriminator, we adopt two discriminators D_I and D_V. Specially, the image discriminator D_I promises the fidelity of the generated frames. It takes image pair (x_t, e_t) as input.While the video discriminator D_V guarantees the temporal dynamics between consecutive frames. It takes consecutive images pairs $(x_{t-K}^{t-1}, w_{t-K}^{t-1})$ as input, where w_{t-K}^{t-1} denotes the optical flow for the K consecutive real images. Thus the GAN loss is written as:

$$L_{GAN} = \min_{G} \max_{D_I} L_I(G, D_I) + \min_{G} \max_{D_V} L_I(G, D_V), \tag{3}$$

where L_I is the LSGAN loss on images defined by the conditional discriminator. We also use VGG feature loss which computes the feature map distances between generated ones and real-images from a pre-trained VGG network:

$$L_{vgg} = \|\mathrm{VGG}(\hat{x}_t) - \mathrm{VGG}(G(x_t|e_t))\|_1. \tag{4}$$

The overall loss functions can be summarized as below:

$$L = L_{GAN} + \lambda_{recon} L_{recon} + \lambda_{vgg} L_{vgg}, \tag{5}$$

where λ_{recon} and λ_{vgg} represent loss weights.

2. More Details and Results

2.1. Implementation Details

We trained our EVP network using Pytorch [2]. We use the Adam optimizer where the learning rate is 10^{-4}, beta1 is 0.5 and beta2 is 0.999. For loss weights, we empirically set the loss weight λ_{cla} and λ_{con} in the *Cross-Reconstructed Emotion Disentanglement* part as 1, and set the weight λ_{recon} and λ_{vgg} in the *Edge-to-Video Translation Network* as 2. It takes about 6 hours to train the cross-reconstructed Emotion disentanglement network, 2 hours for the audio-to-landmark network, and 48 hours to train the rendering-to-video translation network. The whole network is trained and tested on a single NVIDIA GTX 1080Ti.

2.2. Qualitative Results

We show the image results of our EVP algorithm in Figure 1 and make comparisons with the other methods on various sequences as shown in the accompanying video. Our algorithm generates emotional talking faces for different identities, head poses and backgrounds which is better than the other methods.

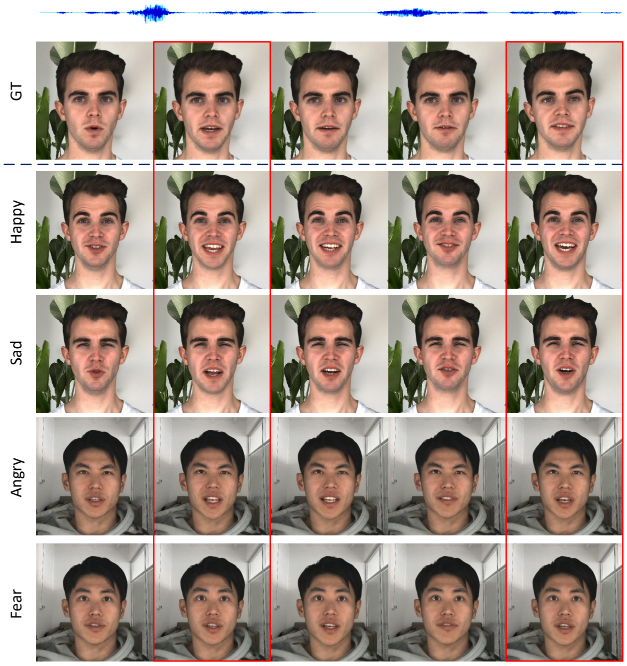

Figure 2: **Results of different input for content and emotion encoders.**

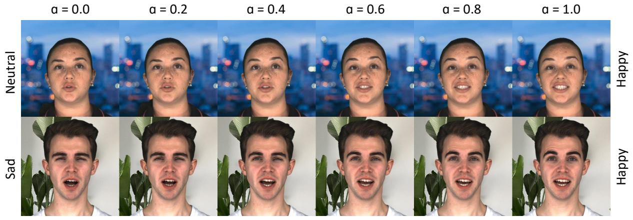

Figure 3: **Emotion category and intensity manipulation.** Here α represents the interploation weight.

To further validate the disentanglement of emotion and content features from audio signals, we use the same audio input for the content encoder and different inputs for the emotion encoder. As shown in Fig. 2, synthesized faces with the same speech content but varying emotions share identical lip movements(in the red box). As for the same emotion, no matter what the speech content is, the emotion is consistently expressed in the generated frames. More results shown in the accompanying video also prove that the speech content and emotion information are successfully decoupled from the audio signals.

Moreover, the learned emotion latent space is continuous, which enables us to edit the emotion in talking face videos, such as emotion category and intensity. We show the emotion editing results in Fig. 3, where the left column is the source emotion E$_s$ and the target emotion E$_t$ is placed on the right. By tuning the weight $\alpha \in (0, 1)$, we can linearly blend the source and target expressions. Meanwhile, the mouth shape remains unchanged, thus achieving emotion manipulation.

References

[1] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 2013. 1

[2] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems (NeurIPS)*, 2019. 2

[3] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1

[4] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018. 2