VideoReTalking: Audio-based Lip Synchronization for Talking Head Video Editing In the Wild

KUN CHENG*, Xidian University, China
XIAODONG CUN*, Tencent AI Lab, China
YONG ZHANG, Tencent AI Lab, China
MENGHAN XIA, Tencent AI Lab, China
FEI YIN, Tsinghua University, China
MINGRUI ZHU, Xidian University, China
XUAN WANG, Tencent AI Lab, China
JUE WANG, Tencent AI Lab, China
NANNAN WANG, Xidian University, China

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Fig. 1. Given an arbitrary talking video and another audio, our method can synthesize a photo-realistic talking video with accurate lip-audio synchronization with retouched face expressions. Natural face © European Central Bank (CC BY).

Both authors contributed equally to this research. Xiaodong Cun is the corresponding author. Project page: https://vinthony.github.io/video-retalking/

We present VideoReTalking, a new system to edit the faces of a real-world talking head video according to input audio, producing a high-quality and lip-syncing output video even with a different emotion. Our system disentangles this objective into three sequential tasks: (1) face video generation with a canonical expression; (2) audio-driven lip-sync; and (3) face enhancement for improving photo-realism. Given a talking-head video, we first modify the expression of each frame according to the same expression template.
using the expression editing network, resulting in a video with the canonical expression. This video, together with the given audio, is then fed into the lip-sync network to generate a lip-syncing video. Finally, we improve the photorealism of the synthesized faces through an identity-aware face enhancement network and post-processing. We use learning-based approaches for all three steps and all our modules can be tackled in a sequential pipeline without any user intervention. Furthermore, our system is a generic approach that does not need to be retrained to a specific person. Evaluations on two widely-used datasets and in-the-wild examples demonstrate the superiority of our framework over other state-of-the-art methods in terms of lip-sync accuracy and visual quality.

CCS Concepts: • Computing methodologies → Animation.

Additional Key Words and Phrases: Facial Animation, Video Synthesis, Audio-driven Generation

ACM Reference Format:
Kun Cheng, Xiaodong Cun, Yong Zhang, Menghan Xia, Fei Yin, Mingrui Zhu, Xuan Wang, Jue Wang, and Nannan Wang. 2022. VideoReTalking: Audio-driven Generation of Talking Video by an Input Audio through Expression Editing and Lip-sync Networks. Natural face © ONU Brasil (CC BY).

Fig. 2. Our method modifies the original video and generates a lip-syncing video by an input audio through expression editing and lip-sync networks. Natural face © ONU Brasil (CC BY).

Inspired by previous inpainting-based talking-head video editing approaches [Prajwal et al. 2020], we present a new system to edit the talking lips to match the input audio with more stable lip-sync results and better visual quality. Previous works consider the original frames in the video as the head pose references. However, we have found that lip generation is very sensitive to these references, and directly using original frames as basis for lip generation often produces out-of-sync results. To this end, as shown in Figure 2, we employ a divide-and-conquer strategy by neutralizing the facial expressions first, then use the modified frames as pose references for lip generation, which is more accurate given that all reference faces now have the same canonical expression. Finally, in contrast to previous works that often produce low-resolution and blurry results, we produce photo-realistic results via the proposed identity-aware enhancement network and the restorations [Wang et al. 2021c; Yang et al. 2021] based on StyleGAN’s facial prior [Karras et al. 2019].

Specifically, given an arbitrary talking video, we first crop the face region and extract the pose and expression coefficients of the 3D Morphable Model (3DMM) by a deep neural network [Deng et al. 2019b]. We then use the parameters of the 3DMM with a standard neutral template expression and re-generate a video through the semantic-guided expression reenactment network similar to [Ren et al. 2021]. By doing so, we obtain a video with the same canonical expression across all the frames, and they will be considered as the structure references for our lip-sync network. Interestingly, we can also synthesize talking head videos with different emotions by changing the expression template. For example, by changing the lip shape of the expression template to match the “happy” emotion, this lip shape will be taken into account in the lip-sync network, causing the talking-head video exhibits the same emotion.

After expression neutralization, a lip-sync network is then applied to synthesize photo-realistic lower-half faces using the synthesized expression as the conditional structure information. Specifically, we design an hourglass-like network with the Fast Fourier Convolution block [Chi et al. 2020] as the basic learning unit, since it achieves great success in the general image inpainting task [Suvorov et al. 2021]. As for the audio injection, we use the Adaptive Instance Normalization (AdaIN) block [Huang and Belongie 2017] to modulate the audio features in global. Similar to [Prajwal et al. 2020], we use a pre-trained lip-sync discriminator to ensure the audio-visual synchronicity.

Although previous steps can synthesize talking-head videos with relatively accurate lip shapes, the visual quality is still limited by the low-resolution training datasets [Afouras et al. 2018; Nagrani et al. 2017]. To solve this problem, we design an identity-preserving face enhancement network to produce high-quality outputs by progressive training. The enhancement network is trained on an enhanced LRS2 dataset [Afouras et al. 2018] enhanced by the face restoration method [Yang et al. 2021]. We also apply the StyleGAN prior guided face restoration network [Wang et al. 2021c] to remove visual artifacts around the teeth.

All the above modules can be applied in sequential order without manual intervention or fine-tuning. We conduct extensive experiments to evaluate our framework on several existing benchmarks as well as in-the-wild videos. Results show that the proposed system
can produce videos with much higher visual quality than previous methods while providing accurate lip synchronization.

2 RELATED WORK

We review the related methods from two aspects, including the visual dubbing task which aims to edit the input video through audio, and single image animation using the audio as conditions.

2.1 Audio-based Dubbing in Video Editing

2.1.1 Arbitrary-subject methods. Arbitrary-subject methods aim at building a general model that does not need to be retrained for different identities. Speech2Vid [Chung et al. 2017] can re-dub a source video with a different segment of audio thanks to the context encoder. Reconstructing the lower-half face by inpainting is popular recently [KR et al. 2019; Park et al. 2022; Prajwal et al. 2020]. For example, LipGAN [KR et al. 2019] design a neural network to fill the lower-half face as a pose prior. Wav2Lip [Prajwal et al. 2020] extends LipGAN using a pre-trained SyncNet as the lip-sync discriminator [Chung and Zisserman 2016] to generate accurate lip synchronization. Based on Wav2Lip, SyncTalkFace [Park et al. 2022] involve the audio-lip memory to store the lip motion features implicitly and retrieve them at inference time. Another category of the methods predicts the intermediate representation first, and then, synthesizes the photo-realistic results by image-to-image translation networks, for example, the facial landmarks [Xie et al. 2021] and the facial landmarks based on 3D face reconstruction [Song et al. 2022]. However, all these methods are struggling to synthesize the high-quality results with editable emotion.

2.1.2 Personalized methods. Personalized visual dubbing is easier than the generic one, since these methods are limited to the certain person in the known environment. For example, SynthesizeObama [Suwajanakorn et al. 2017] can synthesize the mouth region of Obama by the audio-to-landmark network. Inspired by the face reenactment methods [Kim et al. 2018; Thies et al. 2019], recent visual dubbing methods focus on generating the intermediate representation from audio, and then, rendering the photo-realistic results by the image-to-image translation networks. For example, several works [Thies et al. 2020; Wen et al. 2020; Zhang et al. 2021b] focus on the expression coefficient from the audio features and render the photo-realistic results by the image generation networks [Kim et al. 2018; Thies et al. 2019; Wang et al. 2018]. Facial landmarks [Lu et al. 2021] and edges [Ji et al. 2021] are also popular choices by projecting the 3D rendered faces since it contains sparser information. Furthermore, 3D mesh-based [Lahiri et al. 2021] and NeRF [Mildenhall et al. 2020]-based methods [Guo et al. 2021] are also powerful. Although these methods can synthesize the photo-realistic results, they have relatively limited applications because they need to retrain the model on the specific person and environment.

2.2 Audio-based Single Image Facial Animation

Different from the visual dubbing, single image face animation aims to generate the animation by single driven audio, and it has also been influenced by the video-driven face animation. For example, [Song et al. 2018] generate the motion from audio using the recurrent neural network. [Zhou et al. 2019] disentangle the input to subject-related information and speech-related information by adversarial representation learning. [Vougioukas et al. 2020; Zhou et al. 2021] consider the audio as the latent code and drive the face animation by an image generator. The intermediate representation is also a popular choice in this task. ATVG [Chen et al. 2019] and
MakelTalk [Zhou et al. 2020] first generate the facial landmarks from audio, and then, render the video using a landmark-to-video network. Dense flow field is another active research direction [Siarohin et al. 2019; Yin et al. 2022]. [Zhang et al. 2021a] predict the 3DMM coefficients from audio and then transfer these parameters into a flow-based warping network. [Wang et al. 2021b,a] borrow the idea from video driven face animation [Siarohin et al. 2019].

3 FRAMEWORK

Technically, our method is a cross-modal video inpainting framework to fill the masked lower-half face under the guidance of the driven audio and the emotion-modulated reference frame. To this end, we design a lip-sync network (L-Net in Sec. 3.2), which uses the masked lower-half face frames, the given audio, and the original video frames as input to generate a lip-syncing video. However, there are two major problems if we use the L-Net only. The first is the information leak caused by the reference frame, where the generated lip still relies on the reference heavily. The other is the low visual quality since current large-scale talking head datasets are in low resolution.

To this end, except L-Net, we propose two additional modules as shown in Figure 3. First, to solve the information leak, we generate a video with the frozen face expression by a semantic-guided expression reenactment network (D-Net in Sec. 3.1). The synthesized lips are the reference lips instead of the original ones. Then, the lower-half faces of the edited video will be used as a reference structure for our lip-synthesis network (L-Net). In L-Net, our method takes the audio as input and synthesizes the lip-sync results frame-wisely. Furthermore, we design an E-Net for the identity-aware face restoration in Sec. 3.3. Finally, we can paste the generated face back to the original video seamlessly through the post-processing in Sec. 3.4. Below, we give the details of each component.

3.1 Semantic-guided Reenactment Network

It is challenging to edit the lip-related motion in the video directly. Previous works often omit the original lip motion changes [Prajwal et al. 2020] or retiming the background [Song et al. 2022; Suwajanakorn et al. 2017] to avoid unnatural movements between the head pose and lip. Differently, we directly edit the whole lower-half face, including the facial movements with the help of a face reenactment method. Our key observation is that there is an information leak [KR et al. 2019; Prajwal et al. 2020] in conditional in-painting based methods if we use the original frame as the conditional image for lip synchronization. We give an example to show this phenomenon in Figure 4. Given the audio and the input frames, if we directly use the original frames as reference (w/o D-Net), the generated lips will be modified according to the original one. Thus, we aim at editing the expression of the whole lower-half face by the proposed semantic-guided reenactment network. Then, the frame with stable expression will serve as the reference for further lip synthesis.

As shown in Figure 3, after the face detection and crop, we extract the pose and expression coefficients from each frame using monocular face reconstruction [Deng et al. 2019b]. Then, we obtain the new driven signal by replacing the original expression coefficient with the pre-defined expression template. Thus, we can synthesize a video with the frozen expression via the produced dense warp fields of the network and the original frame. Similar to [Ren et al. 2021], the D-Net contains two encoder-decoder-like structures for coarse-to-fine training. After the expression editing, we get the stabilized expression across all the frames. Note that, since the quality of the face reenactment network is still limited, we use the edited face as the structure reference of our lip-sync network. To this end, we first detect facial landmarks, smooth them utilizing a temporal Savitzky–Golay filter, and then use the keypoint of the eye center and the nose as anchors for face alignment.

Interestingly, we can also utilize this information leak caused by the lip-sync reference frame through more expression templates (e.g., smile), resulting in an emotional talking-face video as shown in Figure 1. Since our expression reenactment network only edits the lower-half face of the original video, inspired by the facial action code system [Ekman and Friesen 1978], we can generate the talking faces in other emotions, i.e., anger and surprise, via the image-based expression editing network [Pumarola et al. 2018] on the upper face. We consider it as a plugin and show some results in Sec.5.

3.2 Lip-Sync Network

Our lip-sync network (L-Net) is inspired by a recent conditional inpainting-based framework [Prajwal et al. 2020], which edits the original video directly through new audio. Differently, we use the pre-processed frames from D-Net as the identity and structure reference, the audio and the masked original frames as the condition, to synthesize the lip-syncing video with respect to the input audio.

In Figure 3, we give a brief overview of L-Net, which contains two sub-networks, \( L_a \) and \( L_v \), for audio and video processing, respectively. Here, we give the detailed structure of L-Net in Figure 5.
For the audio processing, we firstly extract the mel-spectrograms from the raw audio and use a ResNet-based encoder [He et al. 2016] to extract the global audio vector \( F_{\text{audio}} \in \mathbb{R}^{256 \times 1 \times 1} \) of a time window. Following previous works, the time window is set to 0.2s per frame, causing the feature in the dimension of 80x16 to process. As for the image generation, we first extract the image features \( F_{\text{ref}}, F_{\text{orig}} \in \mathbb{R}^{256 \times H \times W} \) from the pre-processed referenced images and the original masked image by two different encoders respectively, then, these features are learned to model the relationship between pixels automatically via two cross-attention blocks [Vaswani et al. 2017a]. These cross-attention blocks will calculate the pixel-wise corresponding matrix of two features and enlarge the reception fields. After that, we use nine residual Fast Fourier Convolutional blocks [Chi et al. 2020] to refine the features inspired by recent general image inpainting framework [Suvorov et al. 2021], and we inject the audio features by the AdaIN blocks [Huang and Belongie 2017] which normalize visual features channel-wise after each FFC block. Finally, a series of the convolutional upsampling layers are used to generate the final results.

### 3.3 Identity-aware Enhancement Network

The result from \( L\)-Net is still unperfect since it is hard to train the model on high-resolution talking-head datasets. On the one hand, there is no public available large-scale high-resolution talking-head dataset. On the other hand, if we directly apply the GAN-prior based face restoration networks [Wang et al. 2021c; Yang et al. 2021] as the post-processing tools to improve the results, the results might not be perfect in terms of identity changes [Wang et al. 2021c] and blurry teeth and face [Yang et al. 2021] as shown in Figure 6.

To this end, we propose an identity-aware enhancement network inspired by recent image generation networks [Chan et al. 2021; Karras et al. 2020]. In detail, to acquire the high-resolution talking-head dataset and aligned domain for up-sampling, we enhance the low-resolution dataset firstly using a GAN prior-based face restoration network [Yang et al. 2021]. However, there is a domain gap between the enhanced high-resolution dataset during training and the blurry output of \( D\)-Net during testing. Then, to avoid this gap, we produce the low-resolution input of \( E\)-Net by feeding the enhanced frame and its corresponding audio to the \( L\)-Net. Ideally, \( L\)-Net should produce the same lip motions as the original frame using the conditional audio. Thus, we can use the high-resolution input as supervision directly. As for the architecture, we learn two style-based blocks [Karras et al. 2020] to up-sample the results four times and we design a ResBlock-based encoder \( E_i\)-Net to generate the identity-aware global modulation in each style block.

### 3.4 Post-processing

We also remove several artifacts when pasted back to the original video, including the artifacts of teeth generation and the synthesizing bounding box from the \( L\)-Net. Synthesizing the photo-realistic teeth for the face video is surprisingly hard [Suwajanakorn et al. 2017]. Unlike previous approach which uses the teeth proxy [Suwajanakorn et al. 2017], we seek help from the pre-trained face restoration network [Wang et al. 2021c] for teeth enhancement through face parsing [Yu et al. 2018]. As for the face bounding box caused by \( L\)-Net, we segment [Yu et al. 2018] the produced face and paste back to the original video using the Multi-band Laplacian Pyramids Blending [Burt and Adelson 1983].

### 4 TRAINING

Our framework is implemented using Pytorch [Paszke et al. 2019], and we train each module individually. After training, the whole framework can be tested in a sequence without manual intervention. Below, we give the dataset and training details of each module. More details can be found in supplementary material.

#### 4.1 Training for each module

**4.1.1 \( D\)-Net.** To perform semantic-guided expression reenactment, we train our network on the VoxCeleb [Nagrani et al. 2017] dataset with the pose and expression from [Deng et al. 2019b]. This dataset contains 22496 talking head videos with diverse identities and head poses. We resize the input frames to 256×256 and train the network on the cropped faces similar to [Siarohin et al. 2019]. We train the network in 400k iterations using a progressive training setting. As for the loss function, we calculate the pixel-wise differences between the predicted image and the ground truth using perception loss [Zhang et al. 2018] and gram matrix loss [Gatys et al. 2016].
Fig. 7. Qualitative comparison with LipGAN [KR et al. 2019], Wav2Lip [Prajwal et al. 2020], and PC-AVS [Zhou et al. 2021]. Above two rows show the edit audio and the input video frames, respectively. Note that, to visualize the input audio, we use the audio’s corresponding face to show their mouth shapes.

Natural face © ONU Brasil (CC BY).

|                      | LRS2 Dataset | HDTF Dataset |                      |                      |
|----------------------|--------------|--------------|----------------------|----------------------|
|                      | Visual Quality | Lip-Sync | Visual Quality | Lip-Sync |
|                      | FID↓ | CPBD↑ | LSE-D↓ | LSE-C↑ | FID↓ | CPBD↑ | LSE-D↓ | LSE-C↑ |
| LipGAN [KR et al. 2019] | 5.168 | 0.2615 | 9.609 | 3.062 | 7.684 | 0.2754 | 9.943 | 4.052 |
| Wav2Lip w/o GAN [Prajwal et al. 2020] | 5.069 | 0.2607 | 7.116 | 6.889 | 7.358 | 0.2764 | **8.689** | 5.427 |
| Wav2Lip [Prajwal et al. 2020] | **3.911** | 0.2714 | 7.191 | 6.870 | 5.632 | 0.2763 | 8.895 | 5.228 |
| PC-AVS [Zhou et al. 2021] | 12.800 | 0.2085 | 7.666 | 5.974 | - | - | - | - |
| Ours                 | 5.193 | **0.2809** | **6.519** | **7.089** | **4.504** | 0.2903 | **9.359** | 4.518 |

4.1.2 \( L^-\text{Net} \). We train the \( L^-\text{Net} \) on the LRS2 [Afouras et al. 2018] dataset. This lip-reading dataset contains large-scale 160p videos from BBC programs. We pre-process the dataset using face detection [Bulat and Tzimiropoulos 2017] and resize the input image to 96 x 96 following the previous method [Prajwal et al. 2020]. We train the \( L^-\text{Net} \) using perceptual loss and lip-sync discriminator for visual quality and audio-visual synchronization [Prajwal et al. 2020], respectively.

4.1.3 \( E^-\text{Net} \). The training process of \( E^-\text{Net} \) is based on \( L^-\text{Net} \). We enhance the LRS2 dataset in advance to get a high-resolution dataset, and train the \( E^-\text{Net} \) in 300k iterations. As for the loss function, \( E^-\text{Net} \) is trained on the hybrid losses of perceptual loss [Johnson et al. 2016], pixel-wise \( L_1 \) loss, adversarial loss [Isola et al. 2017], lip-sync discriminator [Prajwal et al. 2020] and identity-loss using a pre-trained face recognition network [Deng et al. 2019a].

4.2 Evaluation

We evaluate the proposed method in terms of visual quality and lip-synchronization. As for the visual quality, since the ground-truth talking video is unavailable, we choose Fréchet inception distance (FID) [Heusel et al. 2017] and cumulative probability blur detection (CPBD) [Narvekar and Karam 2009] to evaluate the visual quality of generated videos. A lower FID score means that the generated images are closer to the dataset distribution. The CPBD reflects the
sharpness of the results. Different from [Prajwal et al. 2020], we compute visual quality metrics on the full frames of the video instead of cropped faces since we focus on the quality of the whole video. We choose the LSE-C and LSE-D [Prajwal et al. 2020] to evaluate the quality of lip synchronization. As for the dataset choices, we evaluate our framework on both low-resolution dataset (LRS2) and high-resolution dataset (HDTF). HDTF dataset contains 720p or 1080p videos from YouTube. Following the unpaired evaluating settings as described in [Prajwal et al. 2020], we take a video and an audio clip from the other different video to synthesize the results. We create 14k and 100 twenty-second audio-video pairs for LRS2 and HDTF dataset evaluation respectively.

5 RESULTS

5.1 Comparison with state-of-the-art Methods

We compare our method with three state-of-the-art methods under the same settings, including LipGAN [KR et al. 2019], Wav2Lip [Prajwal et al. 2020] and PC-AVS [Zhou et al. 2021]. LipGAN and Wav2Lip share the similar network structures. Differently, Wav2Lip uses a pre-trained lip-sync discriminator as the lip-expert, yet a better lip-sync performance. PC-AVS is originally proposed for one-shot pose-controllable talking-head generation. We use the identity code of each original video frame to replace the original single image face animation settings. We compare the proposed method with these methods using their open-sourced codes.

As shown in Table 1, the proposed method achieves much better visual qualities according to CPBD and FID. Since the LRS2 dataset is low-resolution and our method produces high-resolution results, the FID of Wav2Lip on the LRS2 dataset is better. As for the accuracy of lip-sync, our method still gets much better and comparable performance on these two datasets. We also show some examples in Figure 7 to perform the visual comparison. From this figure, our method produces high-quality results with more accurate lip-sync than previous methods. Since visual dubbing is a video editing task, we highly recommend the reader to compare our methods with others refer to the accompanying video.

For the comparison of the lip-sync quality, human evaluation is required. We perform a user study to further evaluate the performance of the proposed method. In the user study, we generate ten talking videos with different audio and video sources of our method and two state-of-the-art methods (LipGAN and Wav2Lip) on the HDTF dataset. We let the users show their opinions about each video in terms of the visual and lip-sync qualities. We set five different scores (larger is better, ranging from 1 to 5) for each option. Our form is sent to 51 people in total, getting 510 opinions. As shown in Table 2, most users prefer to give higher scores to our method with respect to the visual and lip-sync quality.

5.2 Ablation Study

We mainly ablate three major components of our framework in Table 3. The first component is the cross-attention between two image encoders. L-Net w/o cross-attention in Table 3 means channel-wisely concatenating the features from the source and reference frames. We find cross-attention is helpful in terms of the lip-sync quality since it can capture the long-range dependencies. Besides the gains in numerical metrics, we also find it brings more vivid results (e.g. larger mouth). We then show the results of adding the E-Net in our framework. As we expected, the identity-aware face enhancement will hugely improve the visual quality. However, the additional artifacts will also influence the lip-sync quality. Finally, by using D-Net to stabilize reference frames, our framework generates better video in terms of visual and lip-sync quality.

5.3 Extensions to Emotional Talking Video

We have already shown that the proposed method can be used for emotional talking-head video editing in Figure 1. Since our method only modifies the lower-half face, we also get inspiration from the facial action unit system [Ekman and Friesen 1978] and edit the upper face of the images using [Pumarola et al. 2018], causing different combinations as shown in Figure 8.

5.4 Limitation

Although the proposed method can work for the videos in the wild, it still contains some noticeable artifacts in some cases. As shown in Figure 9, one noticeable difference of the proposed framework will cause a slightly identity change from the original video due to the dense warping of D-Net. However, it is only one module of our method and we will replace it with another face reenactment network [Wang et al. 2021] or 3D-based face reenactment method [Kim et al. 2018] directly. Our method also shows some artifacts in some extreme poses as shown in Figure 9. Since our
method edits the video in a frame-by-frame fashion, the results may show some small temporal jittering and flashing.

6 CONCLUSION
We present a generic system for audio-based talking-head video editing by removing the lip motion first and then performing editing. As demonstrated, our framework can work on in-the-wild videos without fine-tuning and produce high-quality results using the audio as the condition. Besides, our system has the potential on the emotional talking-head generation for the lower-half face of the video. We will explore in the future to support more emotions and connect the source audio and contexts to the emotions.

Ethical Considerations. Since our system can edit the talking content of the video in the wild, we also consider the misuse of the proposed method. We will add both robust video and audio watermark to the produced video, and develop the tools to identify the trustworthiness. On the other hand, we hope our method can also help the research in the DeepFake detection.

ACKNOWLEDGMENTS
This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0103202, in part by the National Natural Science Foundation of China under Grant 61922066, 61876142 and 62036007, in part by the Technology Innovation Leading Program of Shaanxi under Grant 2022QFY01-15, in part by Open Research Projects of Zhejiang Lab under Grant 2021KG0AB01.

REFERENCES
Triantafyllos Afromas, Joon Son Chung, Andrew Senior, Orios Vinyals, and Andrew Zisserman. 2018. Deep audio-visual speech recognition. IEEE transactions on pattern analysis and machine intelligence (2018).
Adrian Bulat and Georgios Tramontopulos. 2017. How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks). In International Conference on Computer Vision.
Peter J Burt and Edward H Adelson. 1983. A multiresolution spline with application to image mosaics. ACM Transactions on Graphics (TOG) 2, 4 (1983), 217–236.
Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. 2021. Efficient Geometry-aware 3D Generative Adversarial Networks. In arXiv.
Lele Chen, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. 2019. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 7852–7861.
Lu Chi, Borui Jiang, and Yadong Mu. 2020. Fast Fourier Convolution. In NeurIPS.
J. S. Chung, A. Jamaludin, and A. Zisserman. 2017. You said that?. In British Machine Vision Conference.
A STUDY ON DIFFERENT EXPRESSION TEMPLATES

We use a hand-crafted neutral expression template for all the experiments in our main paper. Here, we provide a study on how different expression templates influence the performance of D-Net. To achieve this goal, we interpolate the expression templates between the neutral and smile expression coefficients and evaluate the results on 10 videos from HDTF [Zhang et al. 2021a] dataset. In Table 4, the lip-sync metrics show minor changes when we edit the expression templates, indicating that D-Net is robust to the expression template.

C.1 Implementation Details of D-Net

C.1.1 Model Architecture. Our method utilizes the information leaks between models for expression editing. Interestingly, we find there is a trade-off between the identity preservation and expression editing when using D-Net. In detail, the expression normalization can be done in both one-shot face reenactment and video to video settings for better expression animation and identity preservation, respectively. In one-shot setting, the reenacted video is reconstructed by warping the first frame of the whole video using the original pose coefficients and template expression coefficients. In this setting, D-Net can generate more stable lip animation, yet a better lip-sync performance. However, since D-Net uses the dense flow for warping, there is a little identity modification. In video-to-video setting, we do not fix the reference frame to be warped, allowing D-Net to change the expression of each frame, which helps identity preservation but cause a little unstable on lip movement. We choose this setting as the default choice in our method.

C.1.2 Loss Functions. For the warping network, we calculate the perceptual loss [Johnson et al. 2016] between the warped image $I_{Dw}$ and ground truth:

$$L_{Dw} = L_{perceptual} = \sum_{l} || f_{vgg}^{l}(I_{gt}) - f_{vgg}^{l}(I_{Dw}) ||^2 $$  (1)

where $f_{vgg}$ is the pre-trained VGG-19 network [Simonyan and Zisserman 2014] and $l$ is the layer of the feature map.

For the editing network, we calculate the perceptual loss and gram matrix style loss [Gatys et al. 2016] between the generated image $I_D$ of the whole D-Net and ground truth:

$$L_c = \sum_{l} || f_{vgg}^{l}(I_{gt}) - f_{vgg}^{l}(I_{D}) ||^2 $$  (2)

$$L_s = \sum_{l} || G(f_{vgg}^{l}(I_{gt})) - G(f_{vgg}^{l}(I_{D})) ||^2 $$  (3)

where $G$ is the gram matrix constructed from activation map. The full optimization objective of editing network is:

$$L_{D_e} = \lambda_c L_c + \lambda_s L_s $$  (4)

where $\lambda_c = 1$ and $\lambda_s = 250$.

C.1.3 Training and Inference Details. We train D-Net on the VoxCeleb [Nagrani et al. 2017] dataset. We first pre-train the mapping network and the warping network for 200k iterations, and then train the whole network for another 200k iterations. We use Adam [Kingma and Ba 2014] optimizer and the learning rate is $1e^{-4}$. During the training phase, the network is doing a self-reconstruction task. We randomly choose two frames from the original video as the source and target image pairs, and we reconstruct the target frame using the source image and coefficients of the target frame. During the testing phase, we use the pose coefficients of the source image and expression coefficients of pre-defined template to normalize the lip shape while maintaining the original pose of the video.

C.2 Implementation Details of L-Net

C.2.1 Model Architecture. The L-Net consists of an audio encoder network $L_a$-Net and a visual encoder-decoder-based network $L_v$-Net, as shown in Figure 10. The audio encoder $L_a$-Net, which consists of several ResBlock-based down-sampling layers, are used to extract the high-level audio features. The encoder of $L_a$-Net is made up of three down-sampling layers which consist of 2D convolution, batch normalization and Leaky-ReLU activation function. Applying two separate encoder networks, the half-masked original frame $I_{lr}^{orig}$ and randomly selected reference frame $I_{lr}^{ref}$ from the same video are encoded to features $F_{orig}$ and $F_{ref}$, respectively, and then, these features are fused to $F_o$ using two cross-attention blocks [Vaswani et al. 2017b] for long-range dependencies and avoid the local information leak. For the cross-attention block, we use $F_{orig}$ to generate query $Q$.
and key $K$, and then use $F_{ref}$ to generate value $V$. We calculate the attention score between $Q$ and $K$ and the weighted sum of $V$ to obtain the $F_v$. The decoder of $L_v$-Net is made up of three up-sampling layers, each of which consist of a convolution-up block, nine Modulated Res-FFC (MR-FFC) blocks and skip connection from visual encoder. More details about the Fast Fourier Convolution (FFC) can be found in [Chi et al. 2020]. All the blocks used in $L$-Net are shown in Figure 11.
C.2.2 Loss Functions. For the visual quality, we calculate the pixel-wise $L_1$ loss in RGB space and perceptual loss in feature space between the generated results $O_{LR}$ of $L$-Net and low-resolution ground truth $I_{gt}$:

\[ L_1 = \|I_{gt} - O_{LR}\|_1, \]

\[ L_{perceptual} = \sum_{i} \|f^{384}_{vgg}(I_{gt}) - f^{384}_{vgg}(O_{LR})\|_2. \]

For the audio-visual synchronization, we use pre-trained SyncNet [Chung and Zisserman 2016; Prajwal et al. 2020] as sync discriminator to calculate sync loss between continuous five frames:

\[ L_{sync} = \frac{1}{N} \sum_{i=1}^{N} -\log(p_{\text{sync}}), \]

\[ p_{\text{sync}} = \frac{\sigma \cdot a}{\max ||e||_2 \cdot ||a||_2}, \]

where $p_{\text{sync}}$ indicates the probability that the input audio-video pair is in sync. $\sigma$ and $a$ are video and audio embeddings extracted from pre-trained SyncNet. The full optimization objective of $L$-Net is:

\[ L_L = \lambda_1 L_1 + \lambda_p L_{perceptual} + \lambda_{sync} L_{sync}, \]

where $\lambda_1 = 1$, $\lambda_p = 1$ and $\lambda_{sync} = 0.3$.

C.2.3 Training and Inference details. We train and inference the $L$-Net following the pipeline of Wav2Lip [Prajwal et al. 2020]. During the training phase, the input frames $I_{LR} \in \mathbb{R}^{256 \times 256 \times 3}$ are made up by five continuous frames with five randomly selected reference from the same video. We also send the corresponding audio window to the network for driving information. The audio features are mel-spectrograms conducted from 16kHz audio with FFT window size 800 and hop size 200. We train $L$-Net in 400k iterations on the LRS2 dataset using Adam optimizer. The learning rate of $L$-Net is $1e^{-4}$. During the testing phase, we use the whole input frames as the reference frame to preserve the pose and background information.

C.3 Implementation Details of E-Net

As discussed in the main paper, E-Net is used to upsample the generated videos by the enhanced LRS2 dataset. Below, we give the details of the implementation and training details.

C.3.1 Model Architecture. Figure 12 shows the architecture of $E$-Net, which contains an identity encoder $E_I$-Net and a super-resolution module $E_u$-Net. In $E_I$-Net, similar to $L$-Net, we feed the continuous five frames of the video frames $I_{HR}$ (concatenation of masked the lower-half face and itself) and the randomly picked references $I_{HR}^{ref}$ from the same video to the $E_I$-Net. Then, these images will be down-sampled with some augmentations (including the differential JPEG compression and bi-linear down-sampling) and send to the pre-trained $L$-Net for lip synchronization. After that, we get edited frames by the audio to further up-sampling. Inspired by StyleGAN [Karras et al. 2020], the super-resolution module $E_u$-Net uses the similar blocks to upsample the low-resolution results. Each style-based layer is built by a StyleConv block and a tRGB block to learn the high-resolution results. More details about the StyleConv and tRGB blocks can be found in [Karras et al. 2020]. In each StyleConv, we also use the features from the identity encoder $E_I$-Net as the modulation for identity preservation. $E_I$-Net is a res-block [He et al. 2016].
Sum
Bilinear Down-Sampling
Conv 3×3, s1p1
Leaky-ReLU
Conv 3×3, s1p1
Leaky-ReLU
Conv 3×3, s1p1
Leaky-ReLU
Bilinear Down-Sampling

Fig. 13. The architecture of ResBlockDown used in $E_I$-Net.

2016] based encoder which consists of six down-sampling layers and a linear layer. We first resize the high-resolution randomly selected reference frames $I_{HR}$ from $384 \times 384$ to $256 \times 256$. Then, the down-sampling layers will be used to extract the high-level feature of the $I_{HR}$ to a 512-dimension vector. The detailed architecture of the ResBlock-based down-sampling layer is shown in Figure 13.

C.3.2 Loss functions. We calculate the pixel-wise $L_1$ loss in RGB space and perceptual loss in feature space between the generated results $O_{HR}$ of $E$-Net and high-resolution ground truth $I_{GT}$:

$$L_1 = ||I_{GT} - O_{HR}||_1, \quad (10)$$

$$L_{perceptual} = \sum \| f_{log}^l(I_{GT}) - f_{log}^l(O_{HR}) \|_2. \quad (11)$$

For better identity preservation, we apply the identity loss. Specifically, we adopt the pre-trained face recognition network ArcFace [Deng et al. 2019a] and calculate this loss in feature space similar to perceptual loss:

$$L_{id} = || f_{arcface}(I_{GT}) - f_{arcface}(O_{HR}) ||_2. \quad (12)$$

To increase the realistic of the generated sample, we also use the adversarial loss:

$$L_{adv}(G_F, D) = \mathbb{E}_{I_{GT}} [ \log D(O_{HR}) ] + \mathbb{E}_{O_{HR}} [ \log (1 - D(G_F(O_{HR})) )]. \quad (13)$$

Finally, the full optimization objective of $E$-Net is:

$$(G_F^*, D^*) = \arg \min_{G_F} \max_D \lambda_1 L_1 + \lambda_p L_{perceptual} + \lambda_{adv} L_{adv} + \lambda_{id} L_{id}, \quad (14)$$

where $\lambda_1 = 0.2$, $\lambda_p = 1$, $\lambda_{adv}=100$ and $\lambda_{id} = 0.4$.

C.3.3 Training details. To train the $E$-Net, we first perform the face restoration network GPEN [Yang et al. 2021] to the LRS2 dataset to obtain the high-resolution training dataset. Then, we use a hybrid data augmentation method of differentiable JPEG and bi-linear down-sampling to get the low-resolution ($96 \times 96$) input $I_{LR}$ of $L$-Net. Notice that, we do not use the original image as the low-resolution sample since there is still a domain gap to avoid the temporal jitting. Driving by the driven audio, we can get the low-resolution lip-synced result for $E$-Net. Ideally, $L$-Net should produce the same lip motions as the original high-resolution input and we use it as the supervision for upsampling. This network is trained in 300k iterations. We use Adam optimizer and the learning rate is $1e^{-5}$. 

$1e^{-5}$.