# EVERYBODY'S TALKIN':LET ME TALK AS YOU WANT

**Anonymous authors**
Paper under double-blind review

Figure 1: **Audio-based video editing.** Within a set of speech videos, speech audio of a randomly chosen speaker, extracted from the video, can be used to drive any video featuring a random speaker.

## ABSTRACT

We present a method to edit a target portrait footage by taking a sequence of audio as input to synthesize a photo-realistic video. This method is unique because it is highly dynamic. It does not assume a person-specific rendering network yet capable of translating one source audio into one random chosen video output within a set of speech videos. Instead of learning a highly heterogeneous and nonlinear mapping from audio to the video directly, we first factorize each target video frame into orthogonal parameter spaces, *i.e.*, expression, geometry, and pose, via monocular 3D face reconstruction. Next, a recurrent network is introduced to translate source audio into expression parameters that are primarily related to the audio content. The audio-translated expression parameters are then used to synthesize a photo-realistic human subject in each video frame, with the movement of the mouth regions precisely mapped to the source audio. The geometry and pose parameters of the target human portrait are retained, therefore preserving the context of the original video footage. Finally, we introduce a novel video rendering network and a dynamic programming method to construct a temporally coherent and photo-realistic video. Extensive experiments demonstrate the superiority of our method over existing approaches. Our method is end-to-end learnable and robust to voice variations in the source audio. Some results are shown in Fig. 1.

## 1 INTRODUCTION

Video portrait editing is a highly sought-after technique in view of its wide applications, such as filmmaking, video production, and telepresence. Commercial video editing applications, such as Adobe Premiere and Apple iMovie, are resource-intensive tools. Indeed, editing audio-visual content would require one or more footage to be reshot. Moreover, the quality of the edited video is highly dependent on the prowess of editors.

Audio-based approach is an attractive technique for automatic video portrait editing. Several methods (Chen et al., 2018; Zhu et al., 2018) are proposed to animate the mouth region of a still image to follow an audio speech. The result is an animated static image rather than a video, hence sacrificing

realism. Audio-driven 3D head animation (Taylor et al., 2016) is an audio-based approach but aiming at a different goal, namely to drive stylized 3D computer graphic avatars, rather than to generate a photo-realistic video. Suwajanakorn et al. (2017) attempted to synthesize photo-realistic videos driven by audio. While impressive performance was achieved, the method assumes the source audio and target video to come from the same identity. The method is only demonstrated on the audio tracks and videos of Barack Obama. Besides, it requires long hours of single-identity data (up to 14 hours (Suwajanakorn et al., 2017)) for training using relatively controlled and high-quality shots.

In this paper, we investigate a learning-based framework that can perform many-to-many audio-to-video translation, *i.e.*, without assuming a single identity of source audio and the target video. We further assume a scarce number of target video available for training, *e.g.*, at most a 15-minute footage of a person is needed. Such assumptions make our problem *non-trivial*: 1) Without sufficient data, especially in the absence of source video, it is challenging to learn direct mapping from audio to video. 2) To apply the framework on randomly chosen source audios and target videos, our method needs to cope with large audio-video variations between different subjects. 3) Without explicitly specifying scene geometry, materials, lighting, and dynamics, as in the case of a standard graphics rendering engine, it is hard for a learning-based framework to generate photo-realistic yet temporally coherent videos.

To overcome the aforementioned challenges, we propose to use the expression parameter space, rather than the full pixels, as the target space for audio-to-video mapping. This facilitates the learning of more effective mapping, since the expression is semantically more relevant to the audio source, compared to other orthogonal spaces, such as geometry and pose. In particular, we manipulate the expression of a target face by generating a new set of parameters through a novel LSTM-based Audio-to-Expression Translation Network. The newly generated expression parameters, combined with geometry and pose parameters of the target human portrait, allow us to reconstruct a 3D face mesh with the same identity and head pose of the target but with new expression (*i.e.*, lip movements) that matches the phonemes of the source audio.

We further propose an Audio ID-Removing Network that keeps audio-to-expression translation agnostic to the identity of the source audio. Thus, the translation is robust to variations in the voices of different people in different source audio. Finally, we solve the difficult face generation problem as a face completion problem conditioned on facial landmarks. Specifically, after reconstructing a 3D face mesh with new expression parameters, we extract the associated 2D landmarks from the mouth region and represent them as heatmaps. These heatmaps are combined with target frames where the mouth region is masked. Taking the landmark heatmaps and the masked target frames as inputs, a video rendering network is then used to complete the mouth region of each frame guided by dynamics of the landmarks.

We summarize our contributions as follows: 1) We formulate an end-to-end learnable framework that supports audio-based video portrait editing. We demonstrate coherent and photo-realistic results by focusing specifically on expression parameter space as the target space, from which source audios can be effectively translated into target videos. 2) We present an Audio ID-Removing Network that encourages an identity-agnostic audio-to-expression translation. This network allows our framework to cope with large variations in voices that are present in the audio sources. 3) We propose a Neural Video Rendering Network based on the notion of face completion with a masked face as input and mesh landmarks as conditions. This approach facilitates the generation of photo-realistic video for multi-personal within one single network.

## 2 Related Work

**Audio-based Facial Animation.** Driving a 3D head model by input audio learns to associate speech features or phonemes with visemes. Taylor et al. (2017) propose to directly map speech phonemes to face rig control parameters. Different 3D head models are applied in speech-driven facial animation, *e.g.*, face rig (Zhou et al., 2018), face mesh (Karras et al., 2017), and expression blendshapes (Pham et al., 2017). Generating photo-realistic portrait videos is more challenging than driving 3D head model since speaker-specific appearance and head pose are crucial for the quality of generated videos. Many methods (Jamaludin et al., 2019; Vougioukas et al., 2018; Zhou et al., 2019; 2020) take a reference image for video generation, and the result is an animation of a still image rather than a natural video. Recently, Suwajanakorn et al. (2017) obtain the state-of-the-art

result in synthesizing the Obama video portrait. However, it assumes the source and target to have the same identity and requires long hour of training data (up to 14 hours). Thus, it is not applicable in audio-based video editing that need to cope with different sources of voice and target actors. In addition, the target video data is relatively scarce. Fried et al. (2019) proposed a time-consuming retrieve algorithm to edit a talking-head video based on its transcript to produce a realistic video. Thies et al. (2020) present a method to extract speech content by DeepSpeech Hannun et al. (2014) to drive frontal talking videos of different speakers. These methods Suwajanakorn et al. (2017); Fried et al. (2019); Kim et al. (2019); Thies et al. (2020) require a person-specific face rendering network for each target person. To generate lip-synced video with natural or speaker-specific head motion, recent methods try to learn head motions from short talkings videos (Chen et al., 2020) or speech audios (Yi et al., 2020).

**Video-based Facial Reenactment.** It is inherently difficult to synthesize mouth movements based solely on speech audio. Therefore, many methods turn to learning mouth movements from videos comprising the same/intended speech content. (Geng et al., 2018; Wiles et al., 2018; Thies et al., 2016; Kim et al., 2018; Nagano et al., 2018). From source portrait video, facial landmarks (Geng et al., 2018; Qian et al., 2019) or expression parameters (Wiles et al., 2018) are estimated to drive the target face image. In these methods, the generated portrait videos are frame-wise realistic but they suffer from poor temporal continuity. ReenactGAN (Wu et al., 2018) is the first end-to-end learnable video-based facial reenactment method. It introduces a notion of "boundary latent space" to perform many-to-one face reenactment. However, ReenactGAN needs person-specific transformers and decoders, which makes the model size increase linearly with person identities raising.

Many model-based methods (Thies et al., 2016; Kim et al., 2018; Nagano et al., 2018) leverage a 3D head model to disentangle facial geometry, expression, and pose. Face2Face (Thies et al., 2016) transfers expressions in parameter space from the source to the target actor. To synthesize a realistic target mouth region, the best mouth image of the target actor is retrieved and warped. Kim et al. (2018) present a method that transfers expression, pose parameters, and eye movement from the source to the target actor. These methods are person-specific therefore rigid in practice (Kim et al., 2018) and suffer audio-visual misalignment therefore creating artifacts leading to unrealistic results (Thies et al., 2016; Kim et al., 2018; Nagano et al., 2018).

**Deep Generative Models.** Inspired by the successful application of GAN (Goodfellow et al., 2014) in image generation (Radford et al., 2015; Mao et al., 2017; Zhu et al., 2017), many methods (Zhou et al., 2019; Thies et al., 2020; Kim et al., 2018) leverage GAN to generate photo-realistic talking face images conditioned on coarse rendering image (Kim et al., 2018), fused audio, and image features (Zhou et al., 2019). Generative inpainting networks (Iizuka et al., 2017; Liu et al., 2018; Yu et al., 2018) are capable of modifying image content by imposing guided object edges or semantic maps (Yu et al., 2018). We convert talking face generation into an inpainting problem of mouth region, since mouth movement is primarily induced by input speeches.

**Monocular 3D Face Reconstruction.** Reconstructing 3D face shape and texture from a single face image has extensive applications in face image manipulation and animation (Fyffe et al., 2014; Garrido et al., 2015; Thies et al., 2016; Roth et al., 2016; Suwajanakorn et al., 2017). In general, monocular 3D face reconstruction produces facial shape, expression, texture, and pose parameters by solving a non-linear optimization problem constrained by a statistical linear model of facial shape and texture, such as Basel face model (Paysan et al., 2009), FaceWarehouse model (Cao et al., 2014), and Flame (Li et al., 2017). Recently, different 3D head models have been increasingly applied in talking portrait video synthesis (Thies et al., 2016; Suwajanakorn et al., 2017; Kim et al., 2018; Nagano et al., 2018; Fried et al., 2019; Kim et al., 2019).

## 3 METHODOLOGY

The architecture of the proposed method is shown in Fig. 2. First, we register a parametric 3D face model (Cao et al., 2014) in the target video, for every portrait video frame to extract face geometry, pose, and expression parameters. Then, the Audio-to-Expression Translation Network learns the mapping from the source audio feature to face expression parameters. We design an Audio ID-Removing Network to alleviate the issues on large variations caused by multiple speakers. Lastly, we formulate the talking face generation problem as a face completion problem guided by mouth region landmarks, in which the landmarks are projected from the reconstructured 3D facial mesh.

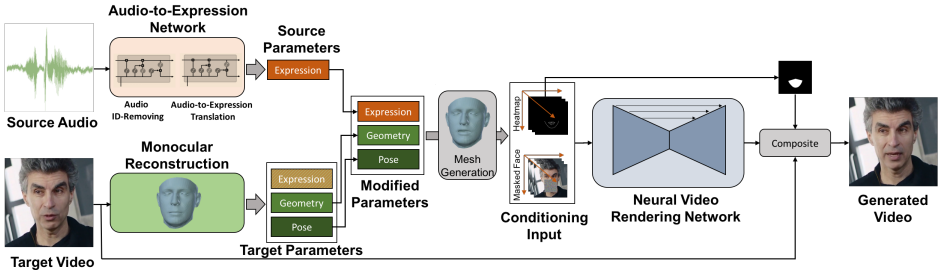



Figure 2: **Architecture.** Our network contains an Audio-to-Expression Translation Network that learns facial expression parameters from speech audio and a Neural Video Rendering Network that generates mouth region guided by projected mouth landmarks.

We propose a Neural Video Rendering network to complete the mouth region of each frame, guided by the dynamics of the landmarks to generate a photo-realistic portrait video.

### 3.1 3D Face Modeling

We leverage a parametric 3D face model (Cao et al., 2014) on portrait video frame to recover low dimensional geometry, expression, and pose parameters. To reduce parameter dimension, geometry and expression bases are computed based on high-quality head scans (Blanz & Vetter, 1999) and facial blendshapes (Cao et al., 2014; Alexander et al., 2010) via principal component analysis (PCA). The geometry parameters $s \in \mathbb{R}^{199}$ and the expression parameters $e \in \mathbb{R}^{29}$ are the coefficients of geometry and expression principal components in the PCA, respectively. The pose of the head $p \in \mathbb{R}^6$ which contains 3 head rotation coefficients, 2 translation coefficients ($x$ and $y$ directions on the screen surface), and 1 scaling coefficient. All the parameters are computed by solving a non-linear optimization problem, constrained by the statistical linear 3D face model (Blanz & Vetter, 1999). By optimizing the geometry, expression, and pose parameters of a given monocular face image based on its detected facial landmarks, portrait video frames will be automatically annotated with low dimensional vectors (Blanz & Vetter, 1999). The recovered expression parameters are used as the learning target in the Audio-to-Expression Translation Network. Then, the recovered geometry and pose parameters, together with the expression parameters inferred by the Audio-to-Expression Translation Network, are employed for reconstructing the 3D facial mesh.

### 3.2 Audio-to-Expression Translation

#### 3.2.1 Audio ID-Removing Network.

We empirically find that identity information embedded in the speech feature degrades the performance of mapping speech to mouth movement. Inspired by the speaker adaptation method in the literature of speech recognition (Visweswariah et al., 2002; Povey & Yao, 2012), we transfer the speech feature lies in different speaker domains onto a "global speaker" domain by applying a linear transformation, in the form (Visweswariah et al., 2002):

$$x' = W_i x + b_i = \bar{W}_i \bar{x}, \text{ where } \bar{W}_i = (W_i,\ b_i),\ \bar{x} = (x;\ 1),\ \bar{W}_i = I + \sum_{j=1}^{k} \lambda_j \bar{W}^j \quad (1)$$

Here, $x$ and $x'$ represent the raw and transferred speech feature respectively, $I$ means an identity matrix padding an all-zero column vector at the rightest side, while $\bar{W}_i = I + \sum_{j=1}^{k} \lambda_j \bar{W}^j$ represents the speaker-specific adaptation parameter that is factorized into a matrix $I$ plus weighted sum of $k$ components $\bar{W}^j$ (Povey & Yao, 2012). In speech recognition, these parameters are iteratively optimized by fMLLR (Digalakis et al., 1995; Gales, 1998) and EM algorithms. We formulate the above method into a neural network to be integrated with our end-to-end deep learning network.

From Eq.(1), the parameters $\lambda_j$ need to be learned from the input speech feature, while the matrix components $\bar{W}^j$ is general speech features of different speakers. Thus, we design an LSTM+FC network to infer $\lambda_j$ from the input and set the matrix components $\bar{W}^j$ as the optimizing parameter of the Audio ID-Removing Network. The matrix components $\bar{W}^j$ of the Audio ID-Removing

Network are updated by the gradient descent-based algorithm. The details of the network is depicted in Fig. 3. The output of the Audio ID-Removing Network is a new MFCC (Mel-frequency cepstral coefficients) spectrum. We apply a pre-trained speaker identity network VGGVox (Nagrani et al., 2017; Chung et al., 2018) on the new MFCC spectrum and constrain the Audio ID-Removing Network by the following cross-entropy loss function:

$$L_{norm} = -\sum_{c=1}^{N} \frac{1}{N} \log p(c|x'), \tag{2}$$

where $N$ is the number of speakers, $c$ is the speaker class label. The $p(c|x')$ is the probability of assigning MFCC $x'$ to speaker $c$, which is inferred from the pre-trained VGGVox. Eq. 2 enforces the Audio ID-Removing Network to produce an MFCC spectrum that is not distinguishable by the pre-trained VGGVox.

### 3.2.2 Audio-to-Expression Translation Network.

We formulate a simple but effective Audio-to-Expression Translation Network that learns the mapping from the ID-removed MFCC feature to the corresponding facial expression parameters. To infer the expression parameters at time $t$, the translation network observes a sliding window speech clip of 1 second, which spans 0.8 seconds before $t$ and 0.2 seconds after $t$.

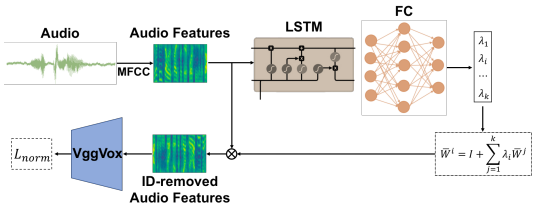



Figure 3: **Audio ID-Removing Network.** We formulate the speaker adaptation method from speech recognition (Visweswariah et al., 2002; Povey & Yao, 2012) as a neural network. The network removes identity in speech MFCC spectrum by transferring it to the "global speaker" domain.

We empirically find it challenging to train a network to solely regress the expression parameters (Sanyal et al., 2019; Tewari et al., 2017). The underlying reason could be that the expression parameters are defined and related to the 3DMM model that is hard to model by the network. To facilitate the learning, we introduce a shape constraint. In particular, with the predicted expression parameters from audio and the ground truth geometry/pose parameters of the video portrait, we can obtain a predicted reconstructed 3D facial mesh. Then, we project 3D points of mouth area to the 2D space to obtain the predicted 2D mouth landmarks. Using a similar method, we can obtain a set of ground-truth 2D mouth landmarks from the ground-truth expression parameters. The shape constraint can be introduced between the predicted 2D mouth landmarks and ground-truth 2D mouth landmarks. The whole process of generating mouth landmarks from expression parameters only involves linear operations and thus is differentiable. The loss function is written as follows:

$$L_{trans} = L_{exp} + L_{shape} = ||\hat{e} - e||_2 + ||\hat{l} - l||_2, \tag{3}$$

where $e$ and $l$ are the ground truth expression and landmark, respectively, and $\hat{e}$ and $\hat{l}$ are the output expression and landmark of the translation network, respectively. The Audio ID-Removing and Audio-to-Expression Translation Networks are trained jointly, whose objective function is weighted sum of $L_{norm}$ (Eq. 2) and $L_{trans}$ (Eq. 3).

## 3.3 Neural Video Rendering Network

### 3.3.1 Network Architecture.

Our final step is to generate photo-realistic talking face video that is conditioned on dynamic background portrait video and guided by the mouth region landmark heatmap sequence. We design a completion-based generation network that completes the mouth region guided by mouth landmarks. First, to obtain the masked face images, a tailored dynamic programming based on retiming algorithm inspired by (Suwajanakorn et al., 2017) is introduced to select frame sequence whose head shaking and blink of eyes look compatible with the source speech. Then, the mouth area that contains lip, jaw, and nasolabial folds are manually occluded by a square mask filled with random

noise. To make the conversion from the landmark coordinates to heatmap differentiable, we follow (Jakab et al., 2017) to generate heatmaps with Gaussian-like functions centered at landmark locations. We modify a U-Net-based (Ronneberger et al., 2015; Qian et al., 2019) network as our generation network. The employed skip-connection enables our network to transfer fine-scale structure information. In this way, the landmark heatmap at the input can directly guide the mouth region generation at the output, and the structure of the generated mouth obeys the heatmaps (Qian et al., 2019; Wang et al., 2019).

We composite the generated mouth region over the target face frame according to the input mouth region mask. To obtain the mouth region mask, we connect the outermost mouth landmarks as a polygon and fill it with white color, then we erode the binary mask and smooth its boundaries with a Gaussian filter (Kim et al., 2019). With the soft mask, we leverage Poisson blending (Pérez et al., 2003) to achieve seamless blending. To improve the temporal continuity of generated video, we apply a sliding window on the input masked video frames and heatmaps (Kim et al., 2018; 2019). The input of the Neural Video Rendering Network is a tensor stacked by seven RGB frames and seven heatmap gray images (Kim et al., 2019). It works well in most cases while a little lip motion jitters and appearance flicker might emerge in the final video. Then, a video temporal flicker removal algorithm improved from (Bonneel et al., 2015a) is applied to eliminate these artifacts. Please refer to *Appendix* for more details of the flicker removal algorithm.

#### 3.3.2 Loss Functions.

The loss function for training the Neural Video Rendering Network is written as follows:

$$L_{render} = L_{recon} + L_{adv} + L_{vgg} + L_{tv} + L_{gp}. \tag{4}$$

The reconstruction loss $L_{recon}$ is the pixel-wise L1 loss between the ground truth and generated images. To improve the realism of the generated video, we apply the LSGAN (Mao et al., 2017) adversarial loss $L_{adv}$ and add the gradient penalty term $L_{gp}$ (Gulrajani et al., 2017) for faster and more stable training. We also apply the perception loss $L_{vgg}$ (Johnson et al., 2016) to improve the quality of generated images by constraining the image features at different scales. The total variation regularization term $L_{tv}$ is used to reduce spike artifact that usually occurs when $L_{vgg}$ is applied (Johnson et al., 2016). The network is trained end-to-end with $L_{total} = L_{norm} + L_{trans} + L_{render}$ (Eq. 2,3, and 4) with different coefficients. Due to the limited space, we report the details of the loss function, network architecture, and experimental settings in our *Appendix*.

## 4 Results

We show qualitative and quantitative results on a variety of videos to demonstrate the superiority of our method over existing techniques and the effectiveness of proposed components. The details of datasets, evaluation metrics and training strategy are presented in Sec. A. The ablation study and user study are presented in the *Appendix*. For all results, we recommend viewing the *supplementary video* for subjective evaluation.

### 4.1 Audio-to-Video Translation

**Many-to-Many Results.** To prove that the audio-to-expression network is capable of handling various speakers and the face completion network can generalize on multiple speakers, we present one-to-many results and many-to-one results in Fig. 4. In the one-to-many results, we use the speech audio of one speaker to drive different speakers. Note that different speakers share a single generator instead of multiple person-specific generators. In the many-to-one results, we use the speech audio of different speakers to drive the same speaker. This is in contrast to recent methods, where the whole pipeline (Suwajanakorn et al., 2017) or part of components (Kim et al., 2019; Fried et al., 2019) is designed for a specific person, which prevents these methods in handling different voice timbres and facial appearances.

The many-to-many results demonstrate the better *generalization ability* of our method compared to existing methods (Kim et al., 2019; Fried et al., 2019; Suwajanakorn et al., 2017). Coping with multiple subjects only requires the algorithm to see 15 minutes video for training. This is more

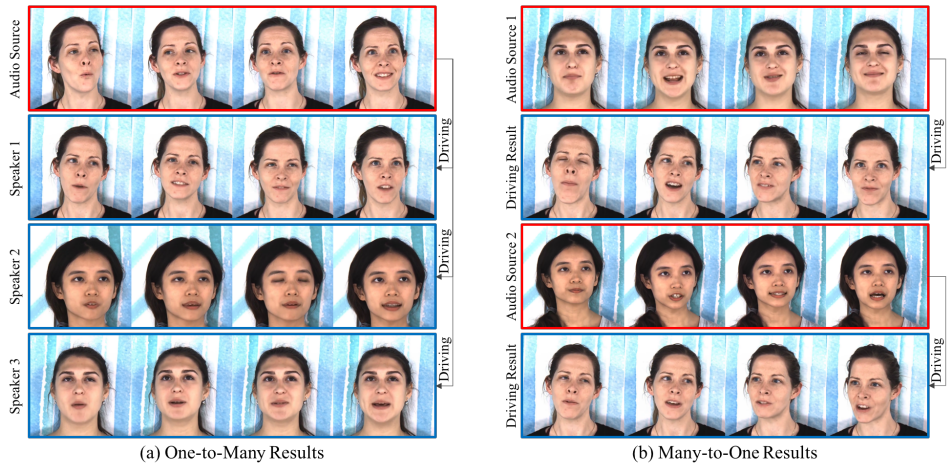



Figure 4: **Many-to-many results.** (a) One-to-many results: we use speech audio of one speaker to drive face of three different speakers. (b) Many-to-one results: we use speech audio of two different speakers to drive the same speaker.

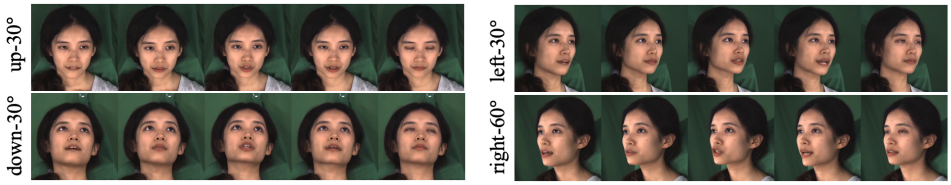



Figure 5: **Large pose results.** We demonstrate four head poses including up, down, and left. Video results of all of seven head poses can be viewed the *supplementary video*.

efficient than existing method, *e.g.*, (Suwajanakorn et al., 2017), which needs up to 17 hours of single identity. Interestingly, once scaling up the number of subjects in the training set further by an order of magnitude, our model can generalize to audio and subject that are totally unseen during training.

**Large Pose Results.** The main purpose of leveraging 3D face model is to handle head pose variations in generating talking face videos. As far as we know, a majority of recent audio-driving methods focus on generating frontal face video no matter whether a 3D head model is applied (Suwajanakorn et al., 2017; Kim et al., 2019; Thies et al., 2020) or not (Vondrick et al., 2016; Zhou et al., 2018; Chen et al., 2019). Our method, however, can generate portrait videos under various large poses driven by audio input, thanks to the decomposition that relates audio only to the expression parameters rather than shape or pose parameters. Many previous methods directly learn a mapping from audio to landmarks (Suwajanakorn et al., 2017; Chen et al., 2019) or frontalized landmrks (Suwajanakorn et al., 2017), which involves the shape and pose information that is actually independent to the input audio. Results are shown in Fig. 5.

**Audio Editing & Singing Results.** Our method can also be used to edit the speech contents of a pre-recorded video by splitting and recombining the words or sentences taken from any source audio. In addition, we also ask a person to record singing and the audio is fed into our network. The driving result can be viewed in Fig. 6. This demonstrates the generalization capability of our method and its potential in more complex audio-to-video tasks.

### 4.2 Comparison with State-of-the-Art Methods

We compare our method with the recent state-of-the-art portrait video generation methods, *e.g.*, Audio2Obama (Suwajanakorn et al., 2017), Face2Face (Thies et al., 2016), Deep Video Portrait (DVP) (Kim et al., 2018), Text-based Editing (TBE) (Fried et al., 2019) and Neural Voice Puppetry

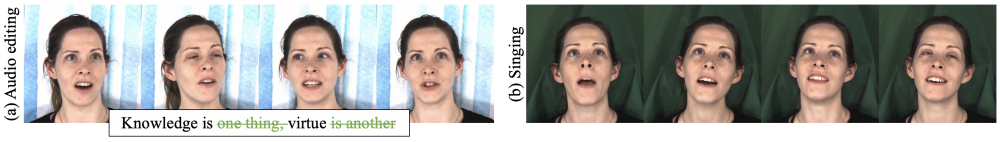



Figure 6: (a) **Audio editing.** We select "knowledge is" and "virtue" from "Knowledge is one thing, virtue is another" in the source audio, then recombine them as "Knowledge is virtue" as input. (b) **Singing.** We evaluate our network on the singing audio clips. Video result can be viewed in the *supplementary video*.

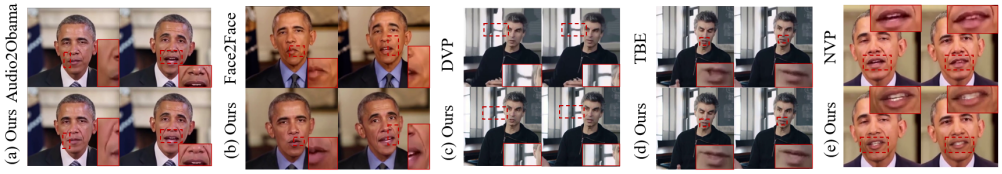



Figure 7: **Comparison to state-of-the-art.** Comparison between our method with Audio2Obama (Suwajanakorn et al., 2017), Face2Face (Thies et al., 2016), DVP (Kim et al., 2018), TBE (Fried et al., 2019) and NVP (Thies et al., 2020).

(NVP) (Thies et al., 2020). The comparison and user study results are demonstrated in Fig. 7 and Sec. B.2.

First, the Audio2Obama (Suwajanakorn et al., 2017) combines a weighted median texture for synthesizing lower face texture and a teeth proxy for capturing teeth sharp details. Our GAN-based rendering network generates better texture details compare to the weighted median texture synthesis (Suwajanakorn et al., 2017), *e.g.*, nasolabial folds (Fig. 7 (a)). Then, we compare our method to Face2Face (Thies et al., 2016) that supports talking face generation driving by source video in Fig. 7 (b). Face2Face (Thies et al., 2016) directly transfers facial expression of source video in the parameter space while our method infers facial expression from source audio. The similar lip movement of Face2Face and our method in Fig. 7 (b) suggests the effectiveness of our Audio-to-Expression Translation Network in learning accurate lip movement from speech audio. Moreover, our GAN-based rendering network generates better texture details, such as mouth corners and nasolabial folds. We also compare to another video-driving method DVP (Kim et al., 2018) that supports talking face generation (Fig. 7 (c)). In DVP, a rendering-to-video translation network is designed to synthesize the whole frame other than the face region. It avoids the blending of face region and background that might be easily detectable. DVP might fail in a complex and dynamic background as shown in Fig. 7 (c). In contrast, our method uses the original background and achieves seamless blending that is hard to distinguish. We compare our method with the contemporary text-based talking face editing method TBE (Fried et al., 2019) in Fig. 7 (d). In TBE, the mouth region is searched by phoneme and a semi-parametric inpainting network is proposed to inpaint the seam between the retrieved mouth and the original face background. This method requires training of a person-specific network per input video while our method can generalize on multiple speakers and head poses. Besides, our generation network produces competitive mouth details as shown in Fig. 7 (d). Finally, we compare our method with NVP (Thies et al., 2020) in Fig. 7 (c). Our method achieves competitive visual quality and lip-sync accuracy. The neural rendering network of NVP is person-specific while ours can be applied to multiple persons.

## 5 Conclusion

In this work, we present an end-to-end learnable audio-based video editing method. At the core of our approach is the learning from audio to expression space bypassing the ill-posed problem of directly mapping audio source to target video. Audio ID-Removing Network and Neural Video Rendering Network are introduced to enable generation of photo-realistic videos given randomly chosen targets and audio sources, within a set of speech videos. Extensive experiments demonstrate the robustness of our method and the effectiveness of each pivotal component. We believe our approach is a step forward towards solving the important problem of audio-based video editing and we hope it will inspire more explorations in this direction.

## REFERENCES


Oleg Alexander, M. Rogers, W. Lambeth, Jen-Yuan Chiang, Wan-Chun Ma, Chuan-Chang Wang, and Paul E. Debevec. The digital emily project: Achieving a photorealistic digital actor. *CG& A*, 30:20–31, 2010.

Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, 1999.

Nicolas Bonneel, James Tompkin, Kalyan Sunkavalli, Deqing Sun, Sylvain Paris, and Hanspeter Pfister. Blind video temporal consistency. *ACM TOG*, 34:196, 2015a.

Nicolas Bonneel, James Tompkin, Kalyan Sunkavalli, Deqing Sun, Sylvain Paris, and Hanspeter Pfister. Blind video temporal consistency. *TOG*, 34:196, 2015b.

Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *TVCG*, 20:413–425, 2014.

Lele Chen, Zhiheng Li, Ross K. Maddox, Zhiyao Duan, and Chenliang Xu. Lip movements generation at a glance. In *ECCV*, 2018.

Lele Chen, Ross K. Maddox, Zhiyao Duan, and Chenliang Xu. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In *CVPR*, 2019.

Lele Chen, Guofeng Cui, Celong Liu, Zhong Li, Ziyi Kou, Yi Xu, and Chenliang Xu. Talking-head generation with rhythmic head motion. *arXiv preprint arXiv:2007.08547*, 2020.

J. S. Chung, A. Nagrani, and A. Zisserman. Voxceleb2: Deep speaker recognition. In *Interspeech*, 2018.

Martin Cooke, Stuart Cunningham, and Xu Shao. An audio-visual corpus for speech perception and automatic speech recognition. *JASA*, 2006.

Vassilios V. Digalakis, Dimitry Rtischev, and L. G. Neumeyer. Speaker adaptation using constrained estimation of gaussian mixtures. *IEEE Transactions on Speech and Audio Processing*, 3:357–366, 1995.

Ohad Fried, Maneesh Agrawala, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, and Christian Theobalt. Text-based editing of talking-head video. *ACM TOG*, 38:68, 2019.

Graham Fyffe, Andrew Jones, Oleg Alexander, Ryosuke Ichikari, and Paul E. Debevec. Driving high-resolution facial scans with video performance capture. *ACM TOG*, 34:8, 2014.

Mark JF Gales. Maximum likelihood linear transformations for hmm-based speech recognition. *Computer Speech & Language*, 12(2):75–98, 1998.

Pablo Garrido, Levi Valgaerts, H. Sarmadi, Ingmar Steiner, Kiran Varanasi, Patrick Pérez, and Christian Theobalt. Vdub: Modifying face video of actors for plausible visual alignment to a dubbed audio track. *CGF*, 34:193–204, 2015.

Jiahao Geng, Tianjia Shao, Youyi Zheng, Yanlin Weng, and Kun Zhou. Warp-guided gans for single-photo facial animation. *ACM TOG*, 37:231, 2018.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *NeurIPS*, 2017.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM TOG*, 36:107, 2017.

Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Conditional image generation for learning the structure of visual objects. In *NeurIPS*, 2017.

Amir Jamaludin, Joon Son Chung, and Andrew Zisserman. You said that?: Synthesising talking faces from audio. *IJCV*, pp. 1–13, 2019.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

Tero Karras, Timo Aila, Samuli Laine, Antti Herva, and Jaakko Lehtinen. Audio-driven facial animation by joint end-to-end learning of pose and emotion. *ACM TOG*, 36:94, 2017.

Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Nießner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *ACM TOG*, 37:163, 2018.

Hyeongwoo Kim, Mohamed Elgharib, Michael Zollhöfer, Hans-Peter Seidel, Thabo Beeler, Christian Richardt, and Christian Theobalt. Neural style-preserving visual dubbing. *ACM TOG*, 38: 178, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM TOG*, 36:194, 2017.

Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, 2018.

Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *ICCV*, 2017.

Koki Nagano, Jaewoo Seo, Jun Xing, Lingyu Wei, Zimo Li, Shunsuke Saito, Aviral Agarwal, Jens Fursund, and Hao Li. pagan: real-time avatars using dynamic textures. *ACM TOG*, 37:258, 2018.

A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In *Interspeech*, 2017.

Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. *AVSS*, pp. 296–301, 2009.

Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *ACM TOG*, 22:313–318, 2003.

Hai X Pham, Samuel Cheung, and Vladimir Pavlovic. Speech-driven 3d facial animation with implicit emotional awareness: A deep learning approach. In *CVPRW*, 2017.

Daniel Povey and Kaisheng Yao. A basis representation of constrained mllr transforms for robust adaptation. *Computer Speech & Language*, 26:35–51, 2012.

Shengju Qian, Kwan-Yee Lin, Wayne Wu, Yangxiaokang Liu, Quan Wang, Fumin Shen, Chen Qian, and Ran He. Make a face: Towards arbitrary high fidelity face manipulation. In *ICCV*, 2019.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Joseph Roth, Yiying Tong, and Xiaoming Liu. Adaptive 3d face reconstruction from unconstrained photo collections. In *CVPR*, 2016.

Soubhik Sanyal, Timo Bolkart, Haiwen Feng, and Michael J. Black. Learning to regress 3d face shape and expression from an image without 3d supervision. *CVPR*, 2019.

Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, 2016.

Supasorn Suwajanakorn, Steven M. Seitz, and Ira Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM TOG*, 36:95, 2017.

Sarah Taylor, Akihiro Kato, Iain A. Matthews, and Ben P. Milner. Audio-to-visual speech conversion using deep neural networks. In *Interspeech*, 2016.

Sarah L. Taylor, Taehwan Kim, Yisong Yue, Moshe Mahler, James Krahe, Anastasio Garcia Rodriguez, Jessica K. Hodgins, and Iain A. Matthews. A deep learning approach for generalized speech animation. *ACM TOG*, 36:93, 2017.

Ayush Tewari, Michael Zollhöfer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Pérez, and Christian Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. *ICCVW*, 2017.

Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *CVPR*, 2016.

Justus Thies, Mohamed Elgharib, Ayush Tewari, Christian Theobalt, and Matthias Nießner. Neural voice puppetry: Audio-driven facial reenactment. In *ECCV*, 2020.

Laurens van der Maaten and Geoffrey E. Hinton. Visualizing data using t-sne. *JMLR*, 9:2579–2605, 2008.

Karthik Visweswariah, Vaibhava Goel, and Ramesh Gopinath. Structuring linear transforms for adaptation using training time information. In *ICASSP*, 2002.

Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *NeurIPS*, 2016.

Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. End-to-end speech-driven facial animation with temporal gans. In *BMVC*, 2018.

Miao Wang, Guo-Ye Yang, Ruilong Li, Run-Ze Liang, Song-Hai Zhang, Peter. M. Hall, and Shi-Min Hu. Example-guided style consistent image synthesis from semantic labeling. In *CVPR*, 2019.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *TIP*, 13(4):600–612, 2004.

Olivia Wiles, A. Sophia Koepke, and Andrew Zisserman. X2face: A network for controlling face generation using images, audio, and pose codes. In *ECCV*, 2018.

Wayne Wu, Yunxuan Zhang, Cheng Li, Chen Qian, and Chen Change Loy. Reenactgan: Learning to reenact faces via boundary transfer. In *ECCV*, 2018.

Ran Yi, Zipeng Ye, Juyong Zhang, Hujun Bao, and Yong-Jin Liu. Audio-driven talking face video generation with natural head pose. *arXiv preprint arXiv:2002.10137*, 2020.

Jiahui Yu, Zhe L. Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. *CoRR*, abs/1806.03589, 2018.

Hang Zhou, Yu Liu, Ziwei Liu, Ping Luo, and Xiaogang Wang. Talking face generation by adversarially disentangled audio-visual representation. In *AAAI*, 2019.

Yang Zhou, Zhan Xu, Chris Landreth, Evangelos Kalogerakis, Subhransu Maji, and Karan Singh. Visemenet: Audio-driven animator-centric speech animation. *ACM TOG*, 37:161, 2018.

Yang Zhou, DIngzeyu Li, Xintong Han, Evangelos Kalogerakis, Eli Shechtman, and Jose Echevarria. Makeittalk: Speaker-aware talking head animation. *arXiv preprint arXiv:2004.12992*, 2020.

Hao Zhu, Aihua Zheng, Huaibo Huang, and Ran He. High-resolution talking face generation via mutual information approximation. *CoRR*, abs/1812.06589, 2018.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

# APPENDIX

## A DATASETS, METRICS AND TRAINING DETAILS

**Datasets.** We evaluate our method on a talking face benchmark dataset GRID (Cooke et al., 2006) and a newly collected speech dataset with seven videos. The GRID dataset contains 1,000 sentences spoken by 18 males and 16 females. We follow the train/test splits of Chen et al. (2018). Since GRID only provides frontal face videos of minor head movement, we additionally collected seven videos for our evaluation. The first portion of these videos are collected from four different speakers who show multiple head poses and time-varying head motion. Example videos of one of the speakers are shown in Fig. 5. Each speaker contributes 15 minutes video for training and 2 minutes video for testing, all videos are captured from seven viewpoints to provide seven head poses. Resolution of each video is $1920 \times 1080$. The second portion of the new dataset contains videos downloaded from YouTube featuring three subjects (Barack Obama, Yoshua Bengio and Sara Seager). Videos of each subject are divided similarly to training (15 minutes) and testing (2 minutes). Different from GRID that is redundant of word contents in training and test sets, the samples of our collected dataset consist of totally different words. In the following evaluations, we use only two models, *i.e.*, Model_Pub trained on GRID dataset and Model_Pri trained on the collected dataset.

**Evaluation Metrics.** To evaluate the accuracy of the expression parameters and the projected landmarks under various head poses and motions, we apply the following distance metric:

$$E_{exp} = \frac{1}{N_{exp}} \sum_{i=1}^{N_{exp}} ||\widehat{e}(i) - e(i)||_2,\ E_{ldmk} = \frac{1}{N_{ldmk}} \sum_{i=1}^{N_{ldmk}} ||\widehat{l}(i) - l(i)||_2\ , \tag{5}$$

where $N_{ldmk}$ and $N_{exp}$ are the number of landmarks and expression parameters, respectively, and $i$ is the index of landmarks or expression parameters. To quantitatively evaluate the generated quality of portrait videos, we apply common image quality metrics like PSNR (Wang et al., 2004) and SSIM (Wang et al., 2004). To qualitatively evaluate the generated quality of portrait videos, we conduct a user study in Sec. B.2.

**Training Details.** We train the Audio-to-Expression Network and Neural Video Rendering Network together in an end-to-end way. All the preprocessings(*e.g.* frame selection) and post-processings(*e.g.* teeth proxy, temporal flicker removal) are nor applied during training. We use the Adam optimizer (Kingma & Ba, 2015) with $\beta_1 = 0.5, \beta_2 = 0.999$. We set the batch size as 1 since our network processes seven frames once, which cost vast GPU memory. In experiments, we find that the Neural Video Rendering Network has a more steep curve than the Audio-to-Expression Network. After hyper parameter tuning, we set the learning rate as $0.001$ and $0.0001$ for the Neural Video Rendering Network and the Audio-to-Expression Network respectively. The learning rate is decreased with scale factor $0.8$ every $10,000$ iterations. Both the Audio-to-Expression and Neural Video Rendering Network are trained for $100,000$ iterations.

## B MORE EXPERIMENTS

### B.1 ABLATION STUDY

**Evaluation of Parameter Regression.** To prove the superiority of incorporating the 3D face model, we compare our network with the one that replaces Audio-to-Expression Translation Network with an Audio-to-Landmark Translation Network as performed in (Suwajanakorn et al., 2017). The

Audio-to-Landmark Translation Network modifies the last fully connected layer of the Audio-to-Expression Translation Network so that its output dimension is the coordinate number of mouth region landmarks. The qualitative comparison can be viewed in Fig. 8 (a). We also compare the quantitative metric on GRID and collected dataset as shown in Tab. 1 (a). In the collected dataset that contains more head motion and poses, our method achieves better lip synchronization results while the mouth generated by the one that applies Audio-to-Landmark Translation Network might not even open (*supplementary video*).

Table 1: (a) **2D vs 3D quantitative comparison.** $E_{exp}$, $E_{ldmk}$, PSNR, and SSIM comparison of 2D and 3D parameter regression. (b) **ID-removing quantitative comparison.** "BL" is the 3D parameter regression baseline; "IR" is "id-removing".

| (a) 2D vs 3D | | $E_{exp}$ | $E_{ldmk}$ | PSNR | SSIM |
|---|---|---|---|---|---|
| GRID | 2D | - | 3.99 | 28.06 | 0.89 |
| | **3D** | **0.65** | **2.24** | **31.19** | **0.95** |
| Collected | 2D | - | 3.13 | 26.76 | 0.93 |
| | **3D** | **0.595** | **1.82** | **29.16** | **0.95** |

| (b) ID Removing | | $E_{exp}$ | $E_{ldmk}$ | PSNR | SSIM |
|---|---|---|---|---|---|
| GRID | BL | 0.84 | 3.09 | 30.88 | 0.94 |
| | +IR | **0.65** | **2.24** | **32.23** | **0.97** |
| Collected | BL | 0.68 | 1.92 | 26.86 | 0.95 |
| | +IR | **0.59** | **1.83** | **31.21** | **0.96** |


Figure 8: (a) **2D vs 3D qualitative comparison.** 3D parameter regression outperforms 2D parameter regression for head motion and poses. (b) **ID-removing qualitative comparison.** Improvement can be observed on a failure case caused by not applying Audio ID-Removing Network.

**Evaluation of ID-removing.** Our Audio ID-Removing Network transfers the speech feature of different speakers to a "global speaker". The tSNE (van der Maaten & Hinton, 2008) maps in Figure 9 demonstrate the 2D visualized the distribution of the input MFCC spectrum and the identity removed MFCC spectrum produced by our Audio ID-Removing Network. We can see that the speaker identity can not be distinguished after removing identity in the MFCC spectrum.

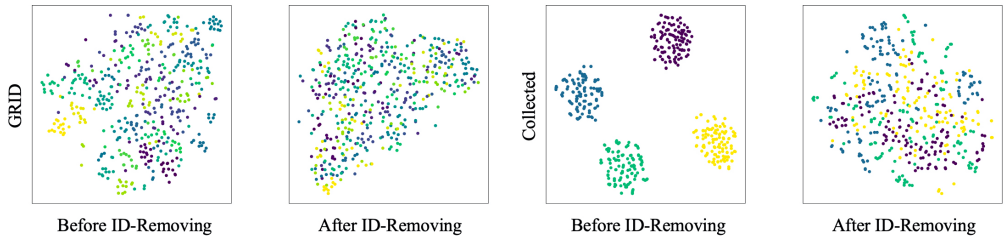



Figure 9: **tSNE before and after id-removing.** 2D visualized distributions of input MFCC and normalized MFCC. Different color represents different speaker. Our audio normalization erases the identity information embedded in the MFCC spectrum.

We also demonstrate the lip-sync improvement improvement in Fig. 8 (b) and *supplementary video*. Quantitative metrics on GRID and collected dataset also validate its effectiveness as shown in Tab. 1 (b) and Tab. 2.

**Evaluation of Temporal Flicker Removal Algorithm** The temporal flicker removal algorithm tries to smooth the output landmark coordinates and eliminate the appearance flicker. The quantitative improvement is slight as shown in Table 2 but the temporal continuity improvement is obvious as shown in the *supplementary video*, especially when the mouth does not open. We demonstrate the quantitative results of our 3D parameter regression baseline, Audio ID-Removing Network and Temporal Flicker Removal Algorithm in Table 2.

**Evaluation of Completion-Based Generation.** We evaluate the effects of the proposed completion-based generation that benefits from joint training on data of different people. As shown in Tab. 3, joint training outperforms training person-specific generators separately, enjoying much fewer net-

Table 2: **ID-removing & Deflicker quantitative comparison.** "BL" is the 3D parameter regression baseline; "IR" is "id-removing"; "DF" is "deflicker". The metrics validate the effectiveness of the proposed components except for the deflicker algorithm. The deflicker algorithm mainly focus on removing temporal discontinuity that can be viewed in the *supplementary video*.

| Method | | $E_{exp}$ | $E_{ldmk}$ | PSNR | SSIM |
|---|---|---|---|---|---|
| GRID | BL | 0.84 | 3.09 | 30.88 | 0.94 |
| | +IR | **0.65** | **2.24** | **32.23** | **0.97** |
| | +DF | 0.84 | 3.07 | 27.71 | 0.92 |
| | +IR+DF | **0.65** | **2.24** | 31.19 | 0.95 |
| Collected | BL | 0.68 | 1.92 | 26.86 | 0.95 |
| | +IR | **0.59** | 1.83 | **31.21** | **0.96** |
| | +DF | 0.68 | 1.92 | 27.46 | 0.93 |
| | +IR+DF | **0.59** | **1.82** | 29.16 | 0.95 |

Table 3: **Training data size and training scheme.** PSNR/SSIM of different amount of training data and training scheme. "Joint" means joint training one generator for all speakers and "Split" means training multiple person-specific generators separately. One generation network contains 75 million parameters and $N(N = 4$ in the table) is the speaker number.

| Time | 2 mins PSNR/SSIM | 5 mins PSNR/SSIM | 10 mins PSNR/SSIM | 15 mins PSNR/SSIM |
|---|---|---|---|---|
| Split ($N\times$ 75M) | 29.621/0.868 | 29.338/0.875 | 29.487/0.849 | 29.650/0.876 |
| Joint (75M) | **30.421/0.886** | **30.664/0.888** | **30.787/0.892** | **31.072/0.897** |

work parameters when the number of speakers increases, regardless of the time length of the training data.

### B.2 User Study

To quantitatively evaluate the visual quality of generated portrait videos, following (Fried et al., 2019), we conduct a web-based user study involving 100 participants on the collected dataset. The study includes three generated video clips for each of the seven cameras and for each of the four speakers, hence a total of 84 video clips. Similarly, we also collect $84$ ground truth video clips and mix them up with the generated video clips to perform the user study. We separately calculate the study results of the generated and ground truth video clips.

All the $84 \times 2 = 168$ video clips are randomly shown to the participants and they are asked to evaluate its realism by evaluating if the clips are real on a likert scale of 1-5 (5-absolutely real, 4-real, 3-hard to judge, 2-fake, 1-absolutely fake) (Fried et al., 2019). As shown in Tab. 4, the generated and the ground truth video clips are rated as "real"(score 4 and 5) in 55.0% and 70.1% cases, respectively. Since humans are highly tuned to the slight audio-video misalignment and generation flaws, the user study results demonstrate that our method can generate deceptive audio-video content under large poses in most cases.

Table 4: **User study compared with ground truth videos.** Results on generated and ground truth video clips for 7 poses.

| | Generated Videos | | | | | | Ground Truth Videos | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| score | 1 | 2 | 3 | 4 | 5 | "real" | 1 | 2 | 3 | 4 | 5 | "real" |
| front | 5.2 | 8.5 | 20.6 | 42.6 | 23.2 | 65.8% | 0.6 | 12.1 | 10.1 | 29.7 | 47.5 | 77.2% |
| up-30° | 4.6 | 25.0 | 14.2 | 36.9 | 19.3 | 56.3% | 0.8 | 13.5 | 13.2 | 29.1 | 43.4 | 72.5% |
| down-30° | 4.8 | 22.0 | 15.2 | 39.7 | 18.3 | 58.0% | 0.9 | 13.6 | 14.1 | 30.2 | 41.3 | 71.5% |
| right-30° | 3.9 | 22.9 | 15.8 | 42.1 | 15.3 | 57.4% | 1.3 | 15.6 | 14.4 | 29.2 | 39.6 | 68.8% |
| right-60° | 7.3 | 33.8 | 11.4 | 36.9 | 10.6 | 47.5% | 0.8 | 17.8 | 15.1 | 29.1 | 37.2 | 66.3% |
| left-30° | 3.2 | 20.9 | 21.9 | 40.1 | 13.9 | 54.0% | 1.1 | 12.8 | 16.1 | 31.5 | 38.6 | 70.1% |
| left-60° | 7.9 | 33.7 | 12.1 | 35.1 | 11.3 | 46.3% | 0.7 | 17.5 | 14.1 | 27.2 | 40.5 | 67.7% |
| all poses | 5.3 | 23.8 | 15.9 | 39.0 | 16.0 | 55.0% | 0.9 | 14.7 | 13.9 | 29.4 | 41.2 | 70.6% |

We also conduct a web-based user study to compare the visual quality, lip-sync accuracy and talking naturalness of our method and recent state-of-the-art methods (Suwajanakorn et al., 2017; Thies et al., 2016; Kim et al., 2018; Fried et al., 2019; Thies et al., 2020). To make fair comparisons with these methods, we use the same audio and background frames. We also asked the participants to evaluate the realisticness, lip-sync accuracy and talking naturalness on a likert scale of 1-5. We present the comparison results in Tab. 5. Even comparing with the perfect results released by these state-of-art methods. Our method still generates comparative results.

Table 5: **User study compared with state-of-the-art methods.** Comparsion of our methods and recent state-of-the-art methods on realisticness, lip-sync accuracy and talking naturalness.

| | Realisticness | | | | | | Lip-sync Accuracy | | | | | | Natural | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| score | 1 | 2 | 3 | 4 | 5 | "real" | 1 | 2 | 3 | 4 | 5 | "real" | 1 | 2 | 3 | 4 | 5 | "real" |
| Audio2Obama | 0 | 0 | 14 | 29 | 57 | 86% | 1 | 4 | 11 | 36 | 48 | 84% | 0 | 8 | 9 | 40 | 43 | 83% |
| Face2Face | 4 | 19 | 28 | 31 | 18 | 49% | 20 | 21 | 29 | 24 | 6 | 30% | 15 | 21 | 34 | 25 | 5 | 30% |
| DVP | 64 | 18 | 16 | 1 | 1 | 2% | 27 | 22 | 24 | 20 | 7 | 27% | 36 | 25 | 25 | 10 | 4 | 14% |
| TBE | 0 | 2 | 10 | 28 | 60 | 88% | 2 | 7 | 14 | 32 | 45 | 77% | 1 | 5 | 11 | 26 | 57 | 83% |
| NVP | 1 | 10 | 22 | 33 | 34 | 67% | 5 | 14 | 33 | 30 | 18 | 48% | 1 | 11 | 33 | 32 | 23 | 55% |
| Ours | 1 | 1 | 9 | 27 | 62 | 89% | 0 | 3 | 15 | 31 | 51 | 82% | 0 | 3 | 10 | 32 | 55 | 87% |

### B.3 Quantitative Comparison on GRID dataset

Our method mainly focuses on talking face video editing, which is different from the recent methods that generate full face from input audio and reference still face image (Vondrick et al., 2016; Jamaludin et al., 2019; Chen et al., 2018; Zhu et al., 2018). Here we quantitatively compare our method with these methods (Vondrick et al., 2016; Jamaludin et al., 2019; Chen et al., 2018; Zhu et al., 2018) on image generation metrics. For a fair comparison, in our method, we do not apply any post-process and we also modify the input of the inpainting network to generate the full face other than the mouth region. Specifically, the original network input tensor is stacked by 7 RGB frames and 7 heatmap gray images (from time $t-6$ to time $t$), we remove the RGB frame and heatmap gray image at time $t$ and require the network to generate the complete frame image at time $t$. Table 6 demonstrates that our approach outperforms these methods in PSNR, SSIM and achieves comparable performance in $E_{ldmk}$. Note that our $E_{ldmk}$ calculates error of 39 mouth region landmarks, the number is more than that of (Vondrick et al., 2016; Jamaludin et al., 2019; Chen et al., 2018).

Table 6: **Comparison on GRID dataset.** SSIM, PSNR and $E_{ldmk}$ results of recent state-of-the-art methods and ours. For a fair comparison, we generate the full face and do not apply any post-processing(*e.g.* temporal flicker removal, teeth proxy).

| Method | PSNR | SSIM | $E_{ldmk}$ |
|---|---|---|---|
| Chen et al. (2019) | 28.45 | 0.60 | 3.28 |
| Vondrick et al. (2016) | 28.45 | 0.60 | 3.28 |
| Jamaludin et al. (2019) | 29.36 | 0.74 | 2.23 |
| Chen et al. (2018) | 29.89 | 0.73 | 1.92 |
| Vougioukas et al. (2018) | 27.98 | 0.84 | - |
| Ours | 30.01 | 0.94 | 2.24 |

### B.4 Evaluation on Unseen Subjects and Sentences

In Figure 10, we present the results on GRID dataset, where we evaluate our method on unseen subjects and sentences. In concrete, we randomly select 80% of sentences and 31 subjects on GRID to form the training set and select 20% of sentences and 2 subjects to form the unseen test set. The results demonstrate the ability of our model to generalize to subject and sentence that are totally unseen during training, once scaling up the number of subjects in the training set further by an order of magnitude.

## C Details of Audio-to-Expression Translation Network

The network architecture of our Audio-to-Expression Translation Network can be viewed in Figure 11. In the training phase, we use paired audio and video frames from the training footage as

Figure 10: **Test on unseen subjects and sentences.** The testing subjects and sentences are unseen during training. For video results please refer to the *supplementary video*.

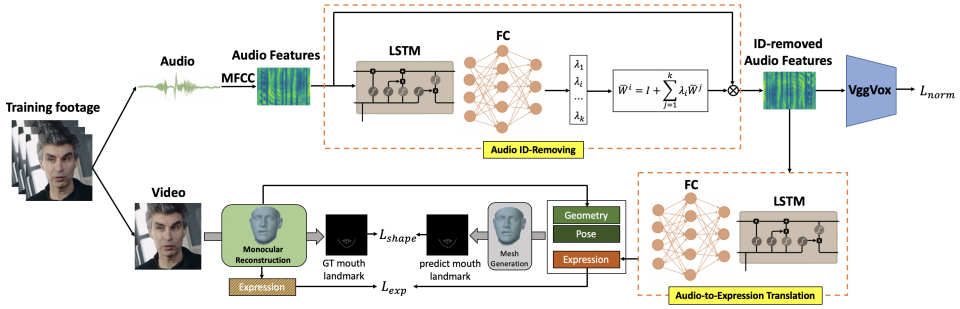



Figure 11: **Architecture of Audio-to-Expression Network.** The Audio ID-Removing Network eliminates identity information in speech audio. The Audio-to-Expression Translation Network estimate expression parameters from input audio. We constrain the predicted expression parameters and the projected 2D mouth landmark from the reconstructed facial mesh.

network input. Ground truth facial shape, expression and pose parameters are calculated from video frame by monocular reconstruction. From the input audio, our Audio-to-Expression Translation Network infers predicted expression parameters that are supervised by the ground truth expression parameters. The loss function is $L_{exp} = ||\hat{e} - e||_2$ in Eq. 6. We reconstruct facial 3D mesh by predicted expression parameters and ground truth shape parameters, then we use the ground truth pose parameters to project 2D mouth landmarks that are supervised by the ground truth 2D mouth landmarks. The loss function is $L_{shape} = ||\hat{l} - l||_2$ in Eq. 6. The ground truth 2D mouth landmarks are projected in a similar way where ground truth expression parameters are used. In the testing phase, the predicted expression parameters from the source audio together with ground truth shape and pose parameters from target video are used to estimate 2D mouth landmarks. The embedded mouth identity and head pose of estimated 2D mouth landmarks are the same as those of the target video while the mouth movement in accord with the source audio.

$$L_{trans} = L_{exp} + L_{shape} = ||\hat{e} - e||_2 + ||\hat{l} - l||_2 \tag{6}$$

## D Details of Neural Video Rendering Network

Our Neural Video Rendering Network modifies a U-Net based (Ronneberger et al., 2015; Qian et al., 2019) architecture and take sequences of masked face and landmark heatmap to generate mouth sequences. The network architecture of our Neural Video Rendering Network is listed in Table 7. In this table, the **Resolution** column indicates the spatial resolution of the output feature map. **EncBlock** means a convolutional layer where stride is 2, padding is 1 and kernel size is $4 \times 4$. **DecBlock** means a convolutional layer where stride is 2, padding is 1 and kernel size is $4 \times 4$, followed by a pixel shuffle (Shi et al., 2016) layer where upscale factor is 2. **Skip**($E$) means a skip connection that concatenates the encoding feature maps from $E$ and previous decoding feature maps. Discriminator architecture of the Neural Video Rendering Network is very similar to the

encoder (E1-E7) of the Neural Video Rendering Network. The architecture is listed in Tab. 8, where **AvgPool** means an average pooling layer and **FC** means a fully connected layer.

Table 7: **Architecture of the Neural Video Rendering Network**

| Layer Name | Resolution | Layer Structure |
|---|---|---|
| Input | $384 \times 384$ | Sequences of Maked Face and Heatmap |
| E1 | $192 \times 192$ | EncBlock $((3+1) \times 7) \rightarrow 64$ + LeakyReLU(0.2) |
| E2 | $96 \times 96$ | EncBlock $64 \rightarrow 128$ + LeakyReLU(0.2) |
| E3 | $48 \times 48$ | EncBlock $128 \rightarrow 256$ + LeakyReLU(0.2) |
| E4 | $24 \times 24$ | EncBlock $256 \rightarrow 512$ +LeakyReLU(0.2) |
| E5 | $12 \times 12$ | EncBlock $512 \rightarrow 512$ + LeakyReLU(0.2) |
| E6 | $6 \times 6$ | EncBlock $512 \rightarrow 512$ + ReLU |
| E7 | $3 \times 3$ | EncBlock $512 \rightarrow 512$ + ReLU |
| D7 | $6 \times 6$ | DecBlock $512 \rightarrow 512$ + ReLU |
| D6 | $12 \times 12$ | Skip(E6)+DecBlock $(512+512) \rightarrow 512$ + ReLU |
| D5 | $24 \times 24$ | Skip(E5)+DecBlock $(512+512) \rightarrow 512$ + ReLU |
| D4 | $48 \times 48$ | Skip(E4)+DecBlock $(512+512) \rightarrow 256$ + ReLU |
| D3 | $96 \times 96$ | Skip(E3)+DecBlock $(256+256) \rightarrow 128$ + ReLU |
| D2 | $192 \times 192$ | Skip(E2)+DecBlock $(128+128) \rightarrow 64$ + ReLU |
| D1 | $384 \times 384$ | Skip(E1)+DecBlock $(64+64) \rightarrow 3 \times 7$ |

Table 8: **Architecture of the Discriminator Network**

| Layer Name | Resolution | Layer Structure |
|---|---|---|
| Input | $384 \times 384$ | Sequences of Generated/Real Face Images |
| E1 | $192 \times 192$ | EncBlock $(3 \times 7) \rightarrow 64$ + LeakyReLU(0.2) |
| E2 | $96 \times 96$ | EncBlock $64 \rightarrow 128$ + LeakyReLU(0.2) |
| E3 | $48 \times 48$ | EncBlock $128 \rightarrow 256$ + LeakyReLU(0.2) |
| E4 | $24 \times 24$ | EncBlock $256 \rightarrow 512$ +LeakyReLU(0.2) |
| E5 | $12 \times 12$ | EncBlock $512 \rightarrow 512$ + LeakyReLU(0.2) |
| E6 | $6 \times 6$ | EncBlock $512 \rightarrow 512$ + ReLU |
| E7 | $3 \times 3$ | EncBlock $512 \rightarrow 512$ + ReLU |
| Output | $1 \times 1$ | AvgPool + FC $512 \rightarrow 1 \times 7$ |

## E Frame Selection Algorithm

In the generated video, our generated mouth region is composited into face frames of a long video footage. Our Frame Selection Algorithm aims at selecting face frames based on input audio to ensure the head motion looks natural with the input audio. For example, it prefers more head motion and eyes blink during utterance (Suwajanakorn et al., 2017). Our Frame Selection Algorithm modifies the Video Re-timing Algorithm of Audio2Obama (Suwajanakorn et al., 2017) to adopt for our setting of multiple speakers . For more algorithm design considerations, please refer to the Video Re-timing Algorithm of Audio2Obama (Suwajanakorn et al., 2017). Here we formulate the dynamic programming objective and present the pseudo-code of our Frame Selection Algorithm.

The Frame Selection Algorithm searches face frames from $M$ target video frames to composite $N$ generated mouth animation frames. To formulate the dynamic programming objective, we apply a threshold on the testing audio volume and compute a binary flag $A(i)$ to indicate slience or utterance, where $i$ denotes frame index of the audio. Like Audio2Obama (Suwajanakorn et al., 2017), we also apply dilation followed by erosion to remove the salt and pepper noise in the binary flag sequences. We use the first derivative of the nose tip landmark position to represent the head motion $V(j)$, where $j$ denotes the frame index of the target video. Here, we use the nose tip landmark position to replace the facial landmarks used in the Re-timing Algorithm (Suwajanakorn et al., 2017) since head motion represented by noise tip will not be influenced by non-rigid part like mouth and eyelid and it represents head motion better. We denote the eyes blink as a binary flag $B(j)$ by applying a threshold on the ratio of eyes height to eyes width. The eyes height are computed by the distance of the highest eye landmark between the lowest eye landmark, and the eyes width are computed similarly. The Re-timing Algorithm (Suwajanakorn et al., 2017) is designed for Obama only. It applies threshold to the size of eyes landmarks in $B(j)$, which is unsuitable for our multiple speaker setting. We also set

**Algorithm 1** Frame Selection Algorithm

**Require:** $A(i)$: binary flag of utterance and slience in source audio at frame $i$ ($i \in \{0, \cdots, N-1\}$);
$V(j)$: head motion and eyes blink representation in target video at frame $j$ ($j \in \{0, \cdots, M-1\}$);
$M, N$: number of target video frames, number of source audio frames;
**Ensure:** $O(i)$: target frame index for source audio at frame $i$ ($i \in \{0, \cdots, N-1\}$);
// Initialize $F$ for $F(0, j, 0),\ F(i, 0, 0),\ F(0, j, 1),\ F(i, j, 1)$.
$F(0, :, 0) = V(:);\ F(0, 0, 0) = 0;\ F(:, 0, 0) = F(:, 0, 1) = F(0, :, 1) = -\infty$;
Evaluate $G(i, j)$ ($i \in \{0, \cdots, N-1\},\ j \in \{0, \cdots, M-1\}$) following Audio2Obama;
// Back track table for $F$: $F_{bt}$.
**for** $i$ `in` $\{1, \cdots, N-1\}$ **do**
  **for** $j$ `in` $\{1, \cdots, M-1\}$ **do**
    $F(i, j, 0) = F(i-1, j-1, \text{argmin}(F(i-1, j-1, :))) + G(i, j)$;
    $F(i, j, 1) = F(i-1, j, 0) + \alpha_s V(j) + G(i, j)$;
    $F_{bt}(i, j, 0) = (i-1, j-1, \text{argmin}(F(i-1, j-1, :)))$;
    $F_{bt}(i, j, 1) = (i-1, j, 0)$;
  **end for**
**end for**
// By back tracking the optimal match score table $F$, we reversely get the target frame index that matches input audio.
$j,\ r = \text{argmin}(F(N-1, :, :))$; $idx = (N-1, j, r)$;
**for** $i \in \{N-1, \cdots, 0\}$ **do**
  $O(i) = idx(1)$;
  $idx = F_{bt}(idx(0), idx(1), idx(2))$;
**end for**
**return** $O(i)$ ($i \in \{0, \cdots, N-1\}$)

$V(j) \leftarrow V(j) + \alpha_B B(j)$ as the Re-timing Algorithm (Suwajanakorn et al., 2017). For the dynamic programming problem formulation, please refer to the Re-timing Algorithm (Suwajanakorn et al., 2017).

We demonstrate the pseudo code of our Frame Selection Algorithm as in Algorithm 1. It is a typical dynamic programming algorithm. In the pseudo code, $F$ stores the match score of generated mouth frame and target frame, and $G$ is an auxiliary function that penalizes large motion during silence and small motion during utterane. For more details about definition of $F$, $G$ and $\alpha_s$, please refer to the Re-timing Algorithm (Suwajanakorn et al., 2017). Following Audio2Obama (Suwajanakorn et al., 2017), each target frame is allowed to appear twice at most.

## F TEETH PROXY ALGORITHM

Our Neural Video Rendering Network implements an GAN-based architecture to generate mouth areas. The teeth area contains high frequency details and our GAN-based network might not produce satisfying teeth texture, especially the resolution of the target frame is high. Thus, we design the Teeth Proxy Algorithm to search teeth texture from target frames and transfer the high frequency information into our generated teeth texture. First, for each target frame, we connect the inner mouth landmarks to form the teeth mask. Then, we register teeth texture covered by the teeth mask. The teeth mask and texture pairs are stored as the *target teeth database*. During the testing phase, we apply the same method to get the generated teeth texture and its mask. We search the most similar teeth texture from the *target teeth database* by ranking IoU (Intersection over Union) value between mask of generated teeth and target teeth masks. Note that all the teeth masks are aligned by mouth center calculated by average value of 4 outmost mouth landmark positions (left, right, up and down). Since all the face images are corasely aligned and affined to a predefined *mean face*, we assume that all the mouth areas have the same scale. At last, we borrow the teeth enhancement method in Audio2Obama (Suwajanakorn et al., 2017) to transfer the high frequency details of teeth. Different from Audio2Obama (Suwajanakorn et al., 2017), we use the searched teeth image to replace *the teeth proxy reference frame*. Our method automatically search appropriate teeth texture while Audio2Obama needs to manually chose reference frame from target video. Note

that our teeth proxy algorithm is an optional post-processing since our GAN-based generator might produce satisfying teeth texture when the resolution of background face frames is not very high.

## G TEMPORAL FLICKER REMOVAL ALGORITHM

In our approach, talking face video is generated frame by frame and temporal information in video frames is only concerned in landmark estimation. During testing, we find the generated talking face videos demonstrate acceptable frame continuity even in the circumstance that the video temporal flicker removal algorithm is not applied. It is due to that audio series clips used to generate time-adjacent frames contain vast overlap in time. The remaining temporal flicker in the video can be attributed to two reasons: 1) The inferred mouth and jawline landmarks contain slight jitter. 2) Appearance flicker, especially color flicker exists in the video frames generated by the inpainting network. Based on the above analysis, our flicker removal algorithm contains two parts: mouth landmark motion smoothing and face appearance deflicker. Algorithm 2 demonstrates the mouth landmark motion smooth algorithm.

**Algorithm 2** Mouth Landmark Smoothing Algorithm

**Require:** $l_{t-1}$: mouth and jawline landmarks at time $t-1$;
$l_t$: mouth and jawline landmarks at time $t$;
$d_{th}$: mouth movement distance threshold, $s$: mouth movement smooth strength;
**Ensure:** $l'_t$: smoothed mouth and jawline landmarks at time $t$;
get mouth center position $c_t$ at time $t$ from $l_t$
get mouth center position $c_{t-1}$ at time $t-1$ from $l_{t-1}$
**if** $||c_t - c_{t-1}||_2 > d_{th}$ **then**
$l'_t = l_t$
**else**
$\alpha = \exp(-s||c_t - c_{t-1}||_2)$
$l'_t = \alpha l_{t-1} + (1-\alpha) l_t$
**end if**
**return** $l'_t$

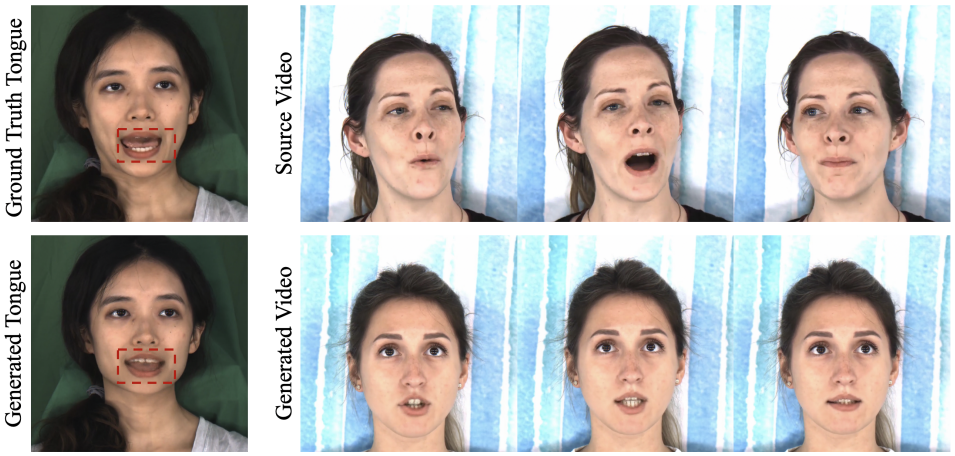
The appearance deflicker algorithm is modified from (Bonneel et al., 2015b). We take mouth movement into consideration. If the mouth does not move, then the color flicker is more obvious, and then we increase the deflicker strength. We denote the generated frame and processed frame at time $t$ as $P_t$ and $O_t$, respectively. The mouth center moving distance between time $t-1$ and $t$ is denoted as $d_t$. The processed frame at $t$ is written as:

$$\mathcal{F}(P_t) = \frac{4\pi^2 f^2 \mathcal{F}(P_t) + \lambda_t \mathcal{F}(\mathrm{warp}(O_{t-1}))}{4\pi^2 f^2 + \lambda_t} \tag{7}$$

where $\lambda_t = \exp(-d_t)$. Here, $\mathcal{F}$ is the Fourier transform and $f$ means frequency. Function $\mathrm{warp}(O_{t-1})$ uses optical flow from $P_{t-1}$ to $P_t$ to warp input frame $O_{t-1}$. Compared with (Bonneel et al., 2015b), the weight of previous frame $\lambda_t$ is measured by the strength of mouth motion instead of global frame consistency.

## H RUNTIME PERFORMANCE

We conduct the inference phrase on a commodity desktop computer with an NVIDIA GTX 1060 and an Intel Core i7-8700. The audio to expression network takes 17 ms per frame and the inpainting network takes 77 ms per frame. The post processes including deflicker and teeth proxy take 1.3s and 300 ms per frame respectively. The deflicker algorithm involves the calculation of optical flow that dominates the inference time. Thus, it takes about 1.7s/100ms to generate one video frame with/without the post-process on average.




(a) Tongue Generation (b) Generation Result with Russian Accent

Figure 12: **Failure cases.** (a) Poor tongue generation result on phoneme "Z" that require the use of tongue. (b) Poor lip-sync accuracy when we use normal speech audio to drive a speaker with strong Russian accent.

## I LIMITATIONS

**Robustness**: The accuracy of 3D face reconstruction is crucial for the quality of our generated videos. Currently, 3D face reconstruction is not very robust for extreme head pose (*e.g.* side view) and low quality face image (*e.g.* GRID dataset). Thus, the quality of our generated talking face varies. Generated videos of Obama are better than generated videos of GRID dataset on visual quality and lip-sync accuracy. Our user study indicates that the realisticness of generated videos decreases as the head pose increases.

**Speaking Style**: Each speaker has an unique speaking style in expression and timing (Kim et al., 2019). Our method assumes that the expression parameters of 3DMM do not contain any speaker information. The Audio-to-Expression Network applies an ID-removing Network to extract speaking content from audio and build a bijection between speaking content and expression parameters. We do not pay much attention to speaking styles in our method. However, such setting makes our method can be extended to generate text-driven talking videos with synchronous artificial voice.

**Emotion**: Our method does not explicitly model facial emotion or estimate the sentiment from the input speech audio. Thus, the generated video looks unnatural if the emotion of the driving audio is different from that of the source video. This problem also appears in (Suwajanakorn et al., 2017) and we leave this to future improvement.

**Tongue**: In our method, our Neural Video Rendering Network produces lip fiducials and the teeth proxy adds the teeth high-frequency details. Our method ignores the tongue movement when some phonemes (*e.g.* "Z" in the word "result") are pronounced. The tongue texture can not be well generated according to lip fiducials and teeth proxy as shown in Figure 12 (a).

**Accent**: Our method performs poorly when the driving speech audio contains an accent. For example, the generated results driven by a speech with a strong Russian accent do not achieve visually satisfactory lip-sync accuracy as shown in Figure 12 (b). We owe it to the fact that English speech with a strong accent is an outlier to our Audio-to-Expression Translation Network and we leave it to future research.

## J ETHICAL CONSIDERATIONS

Our method could contribute greatly towards advancement in video editing. We envisage relevant industries, such as filmmaking, video production, and telepresence to benefit immensely from this technique. We do acknowledge the potential of such forward-looking technology being misused or abused for various malevolent purposes, *e.g.*, defamation, media manipulation, or dissemination of malicious propaganda. Therefore, we strongly advocate and support all safeguarding measures

against such exploitative practices. We welcome enactment and enforcement of legislation to obligate all edited videos to be distinctly labeled as such, to mandate informed consent be collected from all subjects involved in the edited video, as well as to impose hefty levy on all law breakers. Being at the forefront of developing creative and innovative technologies, we strive to develop methodologies to detect edited video as a countermeasure. We also encourage the public to serve as sentinels in reporting any suspicious-looking videos to the authority. Working in concert, we shall be able to promote cutting-edge and innovative technologies without compromising the personal interest of the general public.