

# Live Speech Portraits: Real-Time Photorealistic Talking-Head Animation

YUANXUN LU, Nanjing University and Xmov, China

JINXIANG CHAI\*, Xmov, China

XUN CAO\*, Nanjing University, China

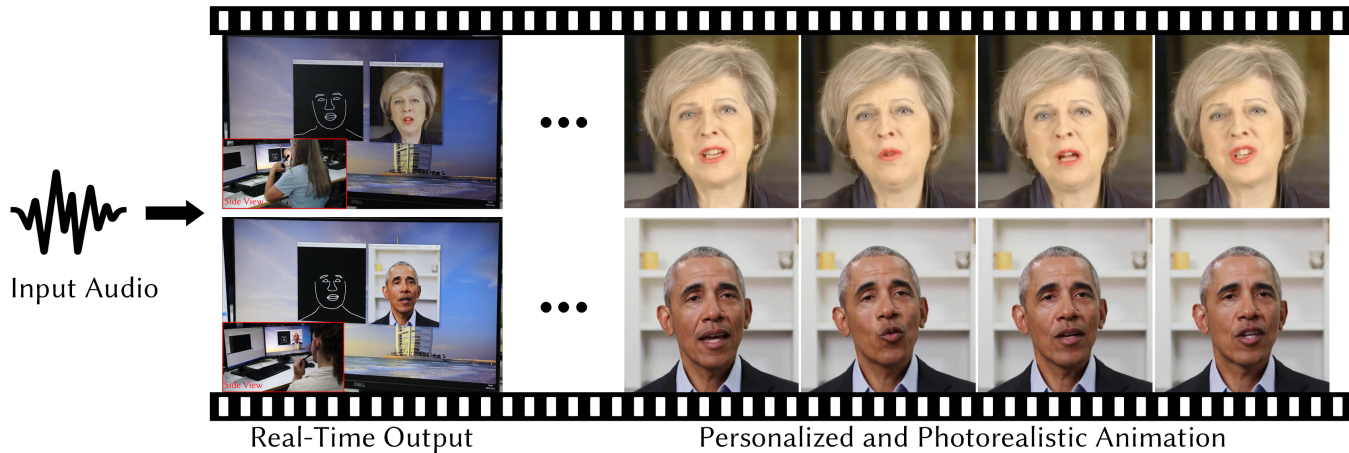


Fig. 1. Given an arbitrary input audio stream, our system generates personalized and photorealistic talking-head animation in real-time. Right: May and Obama are driven by the same utterance but present different speaking characteristics. Video May ©UK government (Open Government Licence). Video Obama ©Barack Obama Foundation (public domain).

To the best of our knowledge, we first present a live system that generates personalized photorealistic talking-head animation only driven by audio signals at over 30 fps. Our system contains three stages. The first stage is a deep neural network that extracts deep audio features along with a manifold projection to project the features to the target person's speech space. In the second stage, we learn facial dynamics and motions from the projected audio features. The predicted motions include head poses and upper body motions, where the former is generated by an autoregressive probabilistic model which models the head pose distribution of the target person. Upper body motions are deduced from head poses. In the final stage, we generate conditional feature maps from previous predictions and send them with a candidate image set to an image-to-image translation network to synthesize photorealistic renderings. Our method generalizes well to wild audio and successfully synthesizes high-fidelity personalized facial details, e.g., wrinkles, teeth. Our method also allows explicit control of head poses.

\*Corresponding authors

Authors' addresses: Yuanxun Lu, Nanjing University and Xmov, China, luyuanxun@smail.nju.edu.cn; Jinxiang Chai, Xmov, Shanghai, China, chajinxiang@xmov.ai; Xun Cao, Nanjing University, Nanjing, China, caoxun@nju.edu.cn.

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, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2021 Association for Computing Machinery.

0730-0301/2021/12-ART220 \$15.00

<https://doi.org/10.1145/3478513.3480484>

Extensive qualitative and quantitative evaluations, along with user studies, demonstrate the superiority of our method over state-of-the-art techniques.

CCS Concepts: • **Computing methodologies** → **Animation**; *Neural networks*; *Rendering*.

Additional Key Words and Phrases: Facial Animation, Neural Rendering

## ACM Reference Format:

Yuanxun Lu, Jinxiang Chai, and Xun Cao. 2021. Live Speech Portraits: Real-Time Photorealistic Talking-Head Animation. *ACM Trans. Graph.* 40, 6, Article 220 (December 2021), 17 pages. <https://doi.org/10.1145/3478513.3480484>

## 1 INTRODUCTION

Talking-head animation, i.e., synthesizing audio-synchronized video frames of a target person, is valuable to interactive applications like digital avatars, video conferencing, visual effects, virtual reality, visual dubbing and computer games. With recent advances in deep learning, people have made great progress in this long-standing problem. However, achieving a realistic and expressive talking-head animation remains an open challenge. Humans are extremely sensitive to any facial artifacts, leading to high requirements for the desired techniques.

Several factors contribute to the challenge. Firstly, attempts to generate lip-synchronized and personalized facial dynamics face a two-fold difficulty, due in part to the challenge of mapping from 1-D audio signals to facial movements which lie on high-dimensional

This project was performed during Yuanxun Lu's internship at Xmov. Project website: <https://yuanxunlu.github.io/projects/LiveSpeechPortraits/>

manifolds, but also due to the domain difference between wild audio and target speech space, which makes the system fail to preserve individual talking idiosyncrasies. Secondly, head and body motion, another critical component of lifelike animation, is not closely correlated to audio. For example, one can swing his head or be still when he says the same words, which depends on many factors - his mood, location, or history poses. Thirdly, synthesizing controllable photorealistic renderings of the target is non-trivial. Nowadays, traditional rendering engines are still far from desired, whose results can be recognized as fake at a glance. Neural renderer shows great power on photorealistic rendering but suffers from performance degradation if the predicted motion is far outside the span of the training corpus [Kim et al. 2018]. Last but not least, many interactive scenarios like video conferencing and digital avatars require the entire system to run in real-time, which makes high demands of the system efficiency without damaging the performance.

In this paper, we propose a deep learning architecture, called Live Speech Portraits (LSP), to address these challenges and step further to practical applications. Our system generates personalized talking-head animation stream, including facial expressions and motion dynamics (head pose and upper body motion) driven by audio and allows for photorealistic rendering in real-time.

First of all, we adopt the idea of self-supervised representation learning, which has shown great power in learning semantic or structural representations and benefits various downstream tasks [Chen et al. 2020b; He et al. 2020; Oord et al. 2018], to extract speaker-independent audio features. To achieve realistic and personalized animation on wild audio streams, we further project the wild features to the target feature space and reconstruct them using target features. This process can be seen as domain adaption from source to target. Subsequently, we are able to learn the mapping from reconstructed audio features to facial dynamics.

Another critical component that contributes to realistic talking-head animation is head and body motion. To generate personalized and time-coherent head poses from audio, we make assumptions that the current head pose is *partly* correlated to audio information and also *partly* to history poses. We propose a novel autoregressive probabilistic model to learn the head pose distribution of the target person based on these two conditions. Head poses are sampled from the estimated distribution, and upper body motions are further deduced from the sampled head poses.

To synthesize photorealistic renderings, we employ an image-to-image translation network conditioned on a feature map and candidate images. We apply the sampled rigid head poses on facial dynamics and project the transformed facial keypoints and upper body positions to the image plane, generating landmark images as our intermediate representations. Although our system comprises several modules, it is still compact enough to run in real-time at over 30 fps. In summary, we present the following contributions:

- To the best of our knowledge, we propose Live Speech Portraits (LSP) as the first audio-driven talking-head animation system with photorealistic renderings in real-time. A comprehensive evaluation demonstrates that our approach outperforms prior methods both qualitatively and quantitatively.
- A novel audio feature extraction module that generalizes our system to wild audio signals. The key component of this module is a manifold projection that reconstructs the deep speech representations using target speech features.
- An elaborately designed probabilistic autoregressive architecture that predicts personalized head pose distributions conditioned on audio signals and history motions. Our system also allows for user-controllable head poses generation.

## 2 RELATED WORK

Mathematically, audio-driven facial animation aims to generate a sequence of talking-head frames from an input audio stream. In the following, we generally review prior work on audio-driven facial animation, as well as related techniques on speech representation learning, head pose estimation and facial reenactment.

*Audio-driven Talking-head Animation.* Audio-driven talking-head animation is a cross-modal research topic with a long history in the computer graphics community. Prior approaches take two different roads depending on whether they aim for generating photorealistic videos. In the non-photorealistic case, these methods focus on learning a mapping from input waveforms to facial movements, e.g., 3D vertex coordinates [Cudeiro et al. 2019; Karras et al. 2017], reference facial model parameters [Taylor et al. 2017] or rigging parameters [Zhou et al. 2018]. These methods usually require high-quality 4D face capture data or rigging parameters with artist interventions. Here, we focus on the photorealistic case which our method belongs to. More than twenty years ago, people have made groundbreaking explorations in this field. Bregler et al. [1997] proposed *Video Rewrite* to create a new person-talking video using existing footage. Brand [1999] proposed *Voice Puppetry* to generate full facial animation from an audio track. These techniques can roughly be categorized into video-based editing methods and image-based generation methods. Video-based editing methods yield editing on a target video - usually synthesize a mouth-related region patch and blend it into the target frame while keeping other regions unaltered [Ezzat et al. 2002; Garrido et al. 2015; Thies et al. 2020]. Recently, Thies et al. [2020] proposed *Neural Voice Puppetry* as an upgrade to *Voice Puppetry*. They first learned a generalized 3D face model from audio sequences and then fine-tuned the model on the target clip via learning a person-specific blendshape basis, in which case the talking style of the target portrait can be preserved. Lower faces were synthesized via a neural rendering network finally. However, these approaches have several intrinsic limitations. First, animation length is restricted to target video length. A heuristic post-processing to select proper candidate frames is required for generating longer videos. Second, head poses and upper body motions are uncontrollable since they are directly copied from target videos, which may conflict with the audio track and introduce barriers to real-time applications. Notably, Suwajanakorn et al. [2017] employed a re-timing schedule to select target frames with natural and synchronized head motions. Last but not least, these methods rely on successful face tracking and tend to fail when faces are partly unseen or undetected, e.g., lower faces are obscured by hands or in a very dark environment. Skipping these bad frames leads to a

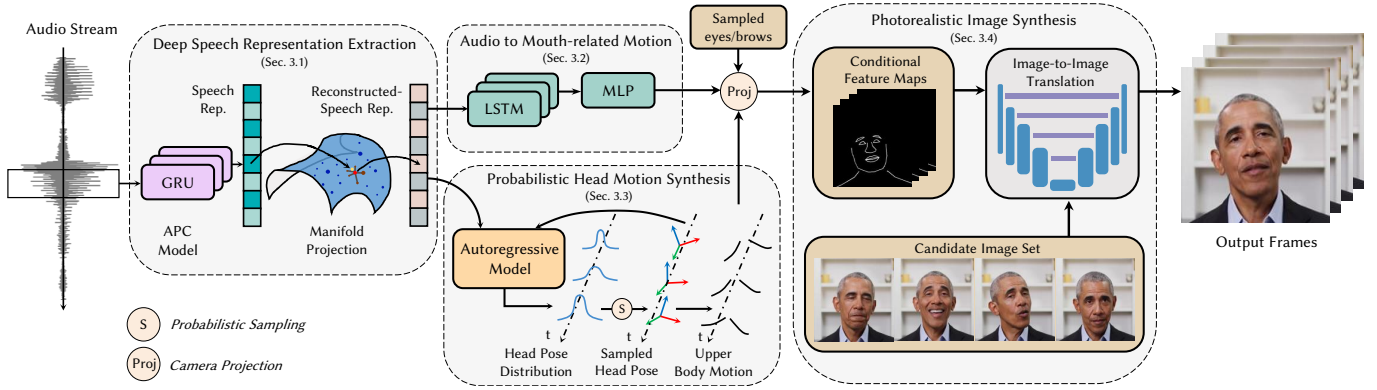


Fig. 2. An overview of our *Live Speech Portraits* method. Given an arbitrary audio stream, our method generates personalized and photorealistic talking animation of a target person in real-time. First, deep speech representations of the input audio are extracted and reconstructed using manifold projection. Then, mouth-related motions, head poses, and upper body motions are predicted from the reconstructed speech representations. We then generate conditional feature maps by projecting the predicted motions and other sampled facial components. Finally, we send conditional feature maps and a candidate image set to an image-to-image translation network to synthesize photorealistic talking portraits. Video Obama ©Barack Obama Foundation (public domain).

temporally inconsistent result. In contrast, our method synthesizes portraits directly. Obstructed frames could be dropped before training without affecting performance.

Image-based generation methods generate a talking-head video based on one or several cropped reference images. This kind of method avoids the pre-mentioned drawbacks but makes the task more challenging for the requirements of manipulating the entire image, including facial details, motion dynamics, and the background. End-to-end training [Chung et al. 2017; Wiles et al. 2018] is becoming a strong trend to generate videos with the rising of deep learning. Chuang et al. [2017] generated a talking face video from a still image and an audio sequence by utilizing a CNN model for the first time. Later, GANs are frequently adopted to generate high-fidelity facial images via adversarial learning [Vougioukas et al. 2018, 2019; Zhou et al. 2019]. Instead of directly synthesizing a talking face image, Chen et al. [2019] and Zhou et al. [2020] leveraged sparse facial landmarks as an intermediate representation. The landmark dynamics were first deduced from the audio input through an audio-to-landmark module and then worked as a condition of an image-to-image translation network to generate animated videos. One common problem shared by these methods is that they tend to learn average facial dynamics among the training corpus without *person-specific* talking styles. Note that Zhou et al. [2020] learned *speaker-aware* dynamics from speaker embedding vectors, but still fail to learn *target-aware* dynamics which may generate uncanny results. Our method focuses on capturing the *person-specific talking dynamics* using only a short target video (around 3 minutes). We utilize facial landmarks as an intermediate representation and generate controllable head poses and upper body motions, which makes the animated videos more impressive and realistic.

**Speech Representation Learning.** Speech signals contain rich high-level information, including content, timbre, and prosody. Much prior work require accurate phoneme labels within millisecond timestamps as input. These labels are often assembled into a sequence of

diphones or triphones to encode neighborhood information [Fan et al. 2015]. However, converting waveforms to phonemes leads to information compression, along with potential performance reduction introduced by error-prone automatic phoneme labeling tools. People also discovered different schemes to get rid of the dependence of phonemes using hand-crafted features [Suwajanakorn et al. 2017]. Recently, modeling these semantic and structural representations through deep neural networks has shown great success and outperforms the traditional hand-crafted features [Devlin et al. 2018; Peters et al. 2018]. Thies et al. [2020] employed a DeepSpeech [Hannun et al. 2014] network to extract speech features. Zhou et al. [2020] resorted to voice conversion community [Qian et al. 2019] to disentangle speech content and identity information. Similarly, Our system uses a self-supervised learning method [Chung and Glass 2020] to extract the high-level speech information. Moreover, manifold projection is applied to improve generalization.

**Head Pose Estimation from Audio.** Head pose, as a significant component of realistic animation, delivers rich information in talking-head videos. Greenwood et al. [2018] employed a Bi-directional LSTM model to predict character head animation from audio. Zhou et al. [2020] predicted speaker-aware head motion dynamics as 3D facial landmark displacements. They trained a transformer architecture [Vaswani et al. 2017] in an adversarial mechanism to capture long time dependencies and generate natural head dynamics. Recently, Chen et al. [2020a] proposed a 3D-aware generative network to learn target-aware head motion from a 3-second video clip. Different from most previous work which use deterministic models, we use an auto-regressive probabilistic model conditioned on history head poses and speech representations to predict the distribution parameters at the current timestamp. Head poses are sampled from the predicted probabilistic model. Besides, we further deduce upper body motions from the predicted head pose, which shows a great improvement on the animation quality.

**Video-based Facial Reenactment.** Video-based facial reenactment is another technique related to audio-driven animation. Thies et al. [2015] proposed the first real-time model-based reenactment system using two RGBD cameras. Face2Face [Thies et al. 2016] extended the boundary using only RGB cameras. Moreover, Liu et al. [2015] combined both audio and visual information as input and tackles the problem that tracking results prone to fail when face is occluded or head pose is extreme. Fried et al. [2019] proposed a method for text-based talking-head editing while is slow for viseme search (5 mins for three words). Yao et al. [2021] reduced the video generation time to 40 seconds for one video. Recently, GANs achieved great success in controllable high-fidelity face synthesis [Karras et al. 2019; Wang et al. 2018a,b]. Few shot or even one-shot facial animation methods are explored via landmarks pre-defined or learned in an unsupervised scheme [Siarohin et al. 2019; Sun et al. 2020; Zakharov et al. 2019]. Most methods rely on an image-to-image mechanism with semantic images as input. Kim et al. [2018] generated portrait videos including head, mouth and gaze from an input reference video. Kim et al. [2019] trained a recurrent GAN to synthesize style-preserving visual dubbing. Very recently, Elgharib et al. [2020] transferred the egocentric view videos to front facing videos using a position conditional GAN. Different from previous methods, our approach generates photorealistic talking head animation from speech only and runs in real-time.

### 3 METHOD

**Overview.** Given an arbitrary speech stream, our live speech portraits approach generates photorealistic talking-head animation of the *target person* in real-time (Figure 2). Our approach consists of three stages: deep speech representation extraction, audio-to-face prediction, and photorealistic face rendering. The first stage extracts the speech representation of the input audio (Section 3.1). The representation extractor learns the high-level speech representation and is trained in a self-supervised manner on an unlabelled speech corpus. We then project the representations to the target person’s speech space to improve generalization. The second stage predicts the full motion dynamics. Two elaborate designed neural networks predict the mouth-related motion (Section 3.2) and head pose (Section 3.3) from the speech representations, respectively. The mouth-related motions are represented as sparse 3D landmarks, and head poses are represented as rigid rotation and translation. Considering that head poses are less related to audio information than mouth-related motions, we employ a probabilistic autoregressive model to learn the poses conditioned on audio information and history poses. Other facial components which have nearly no correlation to audio (e.g., eyes, brows, noses, etc.) are sampled from the training set. We then compute the upper body motion from the predicted head pose. The final stage synthesizes the photorealistic video frames from the previous predictions and a candidate image set (Section 3.4) using a conditional image-to-image translation network. In the following, we introduce each module in detail.

#### 3.1 Deep Speech Representation Extraction

Input information, which are speech signals in our case, plays a crucial role because it powers the entire system. As illustrated in

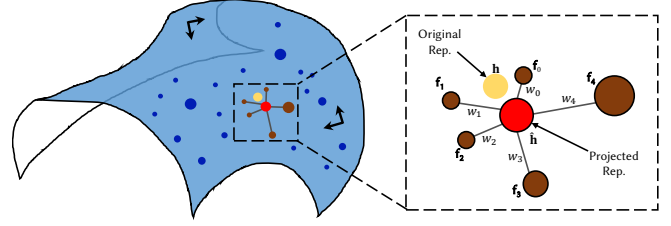


Fig. 3. Manifold projection. Left: For each original deep feature, we project it to the target feature space. Right: Zoom in of the original feature (yellow),  $k = 5$  nearest neighbours (brown), and the reconstructed feature (red).

Section 2, people have exploited deep learning approaches, commonly trained in a self-supervised mechanism, to learn high-level speaker-independent representations of speech from surface features. These methods greatly improve state-of-the-art performance of downstream tasks, e.g., auto speech recognition, phone classification, and speaker verification [Chorowski et al. 2019; Liu et al. 2020; Oord et al. 2018]. One of the reasons these methods succeed is that a huge amount of unlabelled data can be exploited freely, compared to the limited size of the human-labeled dataset.

Specifically, we use the autoregressive predictive coding (APC) model [Chung and Glass 2020] to extract structural speech representations. The APC model predicts future surface features given history information. In our case, we select 80-dimensional log Mel spectrograms as speech surface features. The model is a standard 3-layer unidirectional Gated Recurrent Units (GRUs):

$$\mathbf{h}_l = \text{GRU}^{(l)}(\mathbf{h}_{l-1}), \forall l \in [1, L], \quad (1)$$

where  $\mathbf{h}_l \in \mathbb{R}^{512}$  is the hidden states of each layer in GRUs. The hidden states in the final GRU layer are our desired deep speech representations. We add a linear layer to map the output to predict the future log Mel spectrograms during training, and the linear layer is dropped during testing time.

**3.1.1 Manifold Projection.** Different person owns diverse speaking styles, which are considered as personalized styles. For example, *May clip* exhibits large lip movements and always the ‘O’ style, *Ford clip* exhibits small lip movements like whispers, and *Nadella clip* exhibits adhesion of upper & lower lips like lisps. Directly applying the deep speech representation may lead to poor results when input speech representations locate far away from the target’s speech feature space (e.g., animate a woman by men’s voice, foreign languages, or even songs). We perform manifold projection after extracting the speech representations to improve generalization.

The manifold projection operation is inspired by the recent success in face synthesis from sketches [Chen et al. 2020c], which can be generalized to sketches far away from human faces. We apply the locally linear embedding (LLE) assumption on the speech representation manifold: each data point and its neighbors are locally linear on the high-dimensional manifold [Roweis and Saul 2000].

Given an extracted speech representation  $\mathbf{h} \in \mathbb{R}^{512}$ , we compute the LLE reconstructed representation  $\hat{\mathbf{h}} \in \mathbb{R}^{512}$  on each dimension. As illustrated in Figure 3, we first find the  $K$  nearest points of  $\mathbf{h}$



in a target speech representation database  $\mathcal{D} \in \mathbb{R}^{N_s \times 512}$  by computing the Euclidean distance.  $N_s$  is the number of training frames. Then, we seek a linear combination of the  $K$  nearest neighbours to best reconstruct  $\mathbf{h}$ , which is equivalent to compute the barycentric coordinates of  $\mathbf{h}$  based on its neighbors via solving the following minimization problem:

$$\min \|\mathbf{h} - \sum_{k=1}^K w_k \cdot \mathbf{f}_k\|_2^2, \quad s.t. \sum_{k=1}^K w_k = 1, \quad (2)$$

where  $w_k$  is the barycentric weight of  $k$ -nearest neighbor  $\mathbf{f}_k$ , which can be computed via solving a least-squares problem.  $K$  is chosen as 10 empirically in our experiment. At last, we obtain the projected speech representation  $\hat{\mathbf{h}}$ :

$$\hat{\mathbf{h}} = \sum_{k=1}^K w_k \cdot \mathbf{f}_k. \quad (3)$$

Subsequently,  $\hat{\mathbf{h}}$  is sent to the motion predictors in Section 3.2 and Section 3.3 as input deep speech representations. Our experiment results show that the manifold projection helps improve our system generalization ability. Non-linear projection hasn't been considered for the complexity.

### 3.2 Audio to Mouth-related Motion

Predicting mouth-related motion from audio has been widely researched in the past few years. People use deep learning architectures to learn a mapping from audio features to intermediate representations, e.g., lip-related landmarks [Greenwood et al. 2018; Zhou et al. 2020], parameters of a parametric model [Chen et al. 2019; Suwajanakorn et al. 2017; Taylor et al. 2017], 3D vertices [Cudeiro et al. 2019; Karras et al. 2017], or facial blendshapes [Thies et al. 2020; Zhou et al. 2018]. In our case, we use 3D displacements  $\Delta \mathbf{v}_m \in \mathbb{R}^{25 \times 3}$  with respect to mean positions of the target person in object coordinates as our intermediate representations.

To model sequence dependencies, we use Long Short Term Memory (LSTM) models to learn the mapping from speech representations to mouth-related motions. Similar to [Suwajanakorn et al. 2017], we add a  $d$  frames delay to make the model accessible to a short future, leading to significant improvement on the quality. We later feed the output of the LSTM network to a Multi-Layer Perception (MLP) and finally predict the 3D displacements  $\Delta \mathbf{v}_m$ . In summary, our mouth-related prediction module works as follows:

$$\mathbf{m}_0, \mathbf{m}_1, \dots, \mathbf{m}_t = \text{LSTM}(\hat{\mathbf{h}}_0, \hat{\mathbf{h}}_1, \dots, \hat{\mathbf{h}}_{t+d}), \quad (4)$$

$$\Delta \mathbf{v}_{m,t} = \text{MLP}(\mathbf{m}_t), \quad (5)$$

where time delay  $d$  is set to 18 frames, equal to 300 ms delay in our experiments (60 FPS). The LSTM is stacked into three layers, and each layer has a hidden state of size 256. The MLP decoder network has three layers with hidden state size of 256, 512, and 75.

### 3.3 Probabilistic Head and Upper Body Motion Synthesis

Head pose and upper body motion are another two components that contribute to vivid talking-head animation. For example, people naturally swing their heads and move their bodies when talking, aiming to express emotion and deliver attitudes to the audience. We

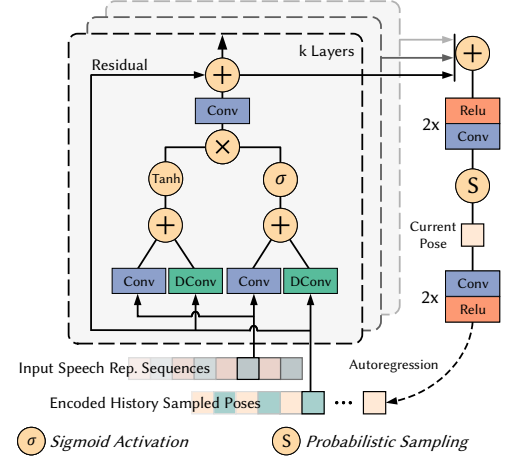


Fig. 4. Illustration of our probabilistic head pose estimation network. This figure demonstrates an example architecture composed of one residual block with three layers.

first describe the method to estimate head pose and then the upper body motion.

Head pose estimation from audio is non-trivial since little relationship exists between them. Considering the intrinsic difficulties, which is a one-to-many mapping from audio to head pose (one can say the same sentence in arbitrary poses), we make two assumptions as prior knowledge.

**ASSUMPTION 1.** *Head poses are partly related to audio information, like expression and intonation. For example, people tend to nod heads when expressing agreement and look up when speaking in rising intonation and vice versa.*

**ASSUMPTION 2.** *Current head pose partly depends on history head poses. For example, there is a large probability that people will turn heads back if they have turned a large angle before.*

These two assumptions simplify the problem and motivate our architecture design. To satisfy the requirements, the proposed network  $\Phi$  should have the ability to see the history head poses and current audio information as conditions. Besides, instead of considering it as a regression problem and train it using Euclidean distance loss [Zhou et al. 2020], we should model this mapping as a probabilistic distribution. Recently, probabilistic models are successfully used in motion synthesis [Henter et al. 2020] and outperform the deterministic models. The joint probability of the head motion can be described as follows:

$$p(\mathbf{x}|\hat{\mathbf{h}}) = \prod_{t=1}^T p(x_t|x_1, x_2, \dots, x_{t-1}, \hat{\mathbf{h}}_t) \quad (6)$$

where  $x$  is the head motion and  $\hat{\mathbf{h}}$  is the speech representation.

The probabilistic model we use is a multi-dimensional Gaussian distribution. The network architecture is inspired by recent success in conditional probabilistic generative modeling [Oord et al.

2016a,b]. The detailed design of the probabilistic model is illustrated in Figure 4. The model is a stack of two residual blocks with seven layers each. Considering the long-time dependencies required to produce natural head motion (one swing his head from left to right may lasts for several seconds), these residual blocks use dilation convolutions layers to capture the dependencies instead of normal convolutions with much fewer parameters. The dilation is doubled seven times for each layer in our architecture and then repeated twice: 1, 2, 4, 8, 16, 32, 64, 1, 2, 4, 8, 16, 32, 64. As a result, the history receptive field size  $F$  of our model is 255 frames, equal to 4.25 seconds in our experiments. The output of each layer is summed up and processed by a post-processing network (a stack of 2 relu-conv layers) to generate the current distribution. In particular, the model outputs the mean values  $\mu$  and the standard deviations  $\sigma$  of the estimated Gaussian. Then we sample in the distribution to get final rigid head pose  $P \in \mathbb{R}^6$ , composed of 3D rotation  $R \in \mathbb{R}^3$  and translation  $T \in \mathbb{R}^3$ . We also tried with a Gaussian Mixture Model but found no obvious improvement. After sampling, we encode the current pose as input pose information for the next time step, forming an autoregressive mechanism. In summary, the head pose estimation can be illustrated as follows:

$$\mathbf{P}_{para,t} = \Phi(\mathbf{P}_{t-F}, \dots, \mathbf{P}_{t-1}, \hat{\mathbf{h}}_t), \quad (7)$$

$$\mathbf{P}_t = \text{Sample}(\mathbf{P}_{para,t}). \quad (8)$$

**Upper Body Motion.** For upper body motion estimation, an ideal method is to build a body model and estimate the parameters [Mehta et al. 2020]. To avoid making the algorithm too complex (upper body always lies only at the bottom part of the image), we assign the upper body part as a billboard [Cao et al. 2016] which is shaped by several shoulder landmarks manually defined. The initial depth of the billboard is set as the average depth of the landmarks in full training sequences and same for all. We translate the billboard model with the 50% translation part  $T$  in the predicted head motion  $P$  in most cases as the results.

### 3.4 Photorealistic Image Synthesis

The last stage of our approach is to generate the photorealistic facial renderings from previous predictions, as illustrated in Figure 2. Our rendering network is inspired by the recent advances in synthesizing photorealistic and controllable facial videos [Elgharib et al. 2020; Isola et al. 2017; Kim et al. 2018; Thies et al. 2019]. We use a conditional image-to-image translation network as our backbone along with adversarial training. The network takes a channel-wise concatenation of a conditional feature map and  $N = 4$  candidate images of the target person to produce photorealistic renderings.

**Conditional Feature Maps.** To provide the clues of the face and the upper body, we draw a conditional feature map for each frame from the above predictions. An example of the conditional map is shown in Figure 5. The feature map consists of a facial part and an upper body part. Drawing semantic regions with colors, or even further one region, one channel brings richer information and more drawing time. We didn't find obvious improvements on these two alternatives. Note that both the sparse facial landmarks and the upper body billboard we predict locate in object coordinates. Therefore,

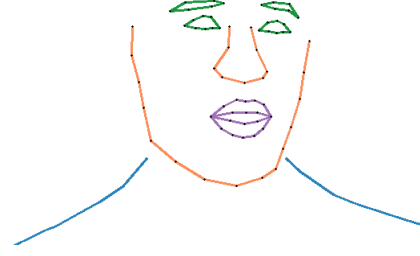


Fig. 5. An example of our conditional feature map. Different colors are used to illustrate different semantic subsets. For example, green denotes the eyes and brows and blue denotes the upper body line. In practical experiments, the conditional feature map is monochrome, as shown in Figure 2.

we need to project these 3D positions to the 2D image plane via pre-computed camera intrinsic parameters  $K$ . The camera model we use is a pinhole camera model and  $K = [f \ 0 \ c_x; 0 \ f \ c_y; 0 \ 0 \ 1]$ , where  $f$  is the focal length and  $(c_x, c_y)$  is the principal point. The consecutive 2D projected components are line-connected in a pre-defined semantic sequence, resulting in the conditional feature map of size  $1 \times 512 \times 512$ .

**Candidate Image set.** Besides the conditional feature map, we additionally input a candidate image set of the target person to provide detailed scene and texture clues. We found that adding such a candidate set helps the network generate consistent backgrounds considering the changing camera motions in the training set and relieves the pressure of the network to synthesize subtle details, like teeth and pores. These images are automatically selected. For the first two, we choose the 100th minimum/maximum mouth area. For the rest, we sample x- and y-axis rotation by uniform intervals and choose the nearest samples from intervals. Therefore, the size of the final concatenated input image becomes  $13(1 + 3 \times 4) \times 512 \times 512$ .

The network is a 8-layer UNet-like [Esser et al. 2018; Han et al. 2019; Ronneberger et al. 2015] convolutional neural network with skip connections in each resolution layer. The resolution of each layer are  $(256^2, 128^2, 64^2, 32^2, 16^2, 8^2, 4^2, 2^2)$  and the corresponding numbers of feature channels are (64, 128, 256, 512, 512, 512, 512, 512). Each encoder layer consists of one convolution (stride 2) and one residual block. The symmetric decoder layer is almost the same, except the first convolution is replaced by a nearest up-sample operation with a scale factor of 2. Examples of our photorealistic renderings are shown in Figure 6.

## 4 IMPLEMENTATION DETAILS

In this section, we describe the aspects relevant to the implementation of our approach: dataset acquisition and pre-processing (Section 4.1), loss functions (Section 4.2), training setup (Section 4.3), and real-time animation settings (Section 4.4).

### 4.1 Dataset Acquisition and Pre-processing

We apply our approach to 8 different target sequences of 7 different subjects for training and testing. These sequences span a range of 3-5 minutes. All videos are extracted at 60 frames per second

(FPS) and the synchronized audio waves are sampled at 16K Hz frequency. We first crop the video to keep the face at the center and then resize to  $512 \times 512$ . All input images and output images share the same resolution. We split videos as 80% / 20% for training and validation. Please refer to Appendix A for more details.

We detect the 73 pre-defined facial landmarks for all videos using an off-the-shelf tool. To provide the groundtruth of the 3D mouth shape and head pose, we employ an optimization-based 3D face tracking algorithm similar to [Shi et al. 2014; Thies et al. 2016]. For camera calibration, we use a binary search to compute the focal length  $f$  as demonstrated in [Cao et al. 2013]. The principle point  $(c_x, c_y)$  is set as the center of the image. Note that we do camera calibration and 3D face tracking on the original image and compute a transformation matrix according to the crop and resize parameters. The upper body motion feature points are manually selected once for first frame of each sequence and tracked for the rest frames using LK optical flow [Bouguet et al. 2001], and the OpenCV implementation [Bradski 2000]. For more details about monocular 3D face tracking, we recommend readers to refer to the summary paper [Zollhöfer et al. 2018].

To train the APC speech representation extractor, we use the Mandarin Chinese part of the Common Voice dataset [Ardila et al. 2020] that provides unlabelled wild utterances. Specifically, The subset contains 889 different speakers with various accents. In total, there are about 26 hours of unlabelled utterances. We use 80-dimensional log Mel spectrograms as surface features. The log mel spectrograms are computed with 1/60 second frame length, 1/120 second frame-shift, and 512-point Short-Time Fourier Transform (STFT). Although our APC model is trained in Mandarin, we find that our system still works well in other languages because the model learns the high-level and semantic information. Also the manifold projection improves the generalization ability.

## 4.2 Loss Functions

**4.2.1 Deep Speech Representation Extraction.** The training of the APC model is fully self-supervised via predicting the surface features  $n$  frames ahead. Given a sequence of log mel spectrograms  $(x_1, x_2, \dots, x_T)$ , the APC model processes each element  $x_t$  at time step  $t$  and outputs a prediction  $y_t$ , generating a predicted sequence  $(y_1, y_2, \dots, y_T)$ . We optimize the model by minimizing the L1 loss between the input sequence and the predicted as follows:

$$\sum_{i=1}^{T-n} |x_{i+n} - y_i|, \quad (9)$$

where  $n = 3$  following the setting in [Chung and Glass 2020].

**4.2.2 Audio to Mouth-related Motion.** To learn the mapping from audio to mouth-related motion, we minimize the  $L_2$  distance between the groundtruth mouth displacements and the predicted displacements. Specifically, the loss can be written as:

$$\sum_{t=1}^T \sum_{i=1}^N \|\Delta \mathbf{v}_{m,t} - \hat{\Delta \mathbf{v}}_{m,t}\|_2^2, \quad (10)$$

where  $T = 240$  represents the number of consecutive frames sent to the model at each iteration.  $N = 25$  is the number of the pre-defined sparse mouth-related 3D points in our experiments.

**4.2.3 Probabilistic Head Motion Synthesis.** Apart from learning the mapping from audio to mouth-related motion, we also aim to estimate the target's head pose during training. The upper body motion can be deduced from the head pose as mentioned in Section 3.3. Specifically, we employ an autoregressive probabilistic model to model the head pose distribution. We train the model by minimizing the negative log-likelihood of the pose distribution. Given a sequence of history head pose  $(x_{t-F}, \dots, x_t)$  and speech representation  $\mathbf{h}_t$ , the probabilistic loss is:

$$-\ln(\mathcal{N}(\mathbf{x}_t, \mathbf{h}_t | \hat{\mu}_n, \hat{\sigma}_n)), \quad (11)$$

where  $\mathbf{x}_t, \mathbf{h}_t$  is the input head pose and speech representation at time  $t$ . This loss term forces the model to output the mean values  $\hat{\mu}_n$  and standard deviations  $\hat{\sigma}_n$  of the Gaussian distribution. To increase numerical stability, we output the negative log sigma instead of the sigma directly. Each element  $\mathbf{x}_t \in \mathbb{R}^{12}$  in pose sequence are composed of the current pose  $\mathbf{p}_t \in \mathbb{R}^6$  and a linear velocity term  $\Delta \mathbf{p}_t \in \mathbb{R}^6$ . Although we only use the first six dimensions of rotation and translation after sampling in the distribution, we find that adding such a velocity term could implicitly force the model to focus on the motion speed, leading to smoother results.

**4.2.4 Photorealistic Image Synthesis.** Finally, we train a neural renderer to synthesis photorealistic talking-head images. The training procedure follows the adversarial training mechanism [Goodfellow et al. 2014]. We adopt the multi-scale PatchGAN architecture [Isola et al. 2017; Wang et al. 2018b] as the backbone of discriminator  $\mathbf{D}$ . The image-to-image translation network  $\mathbf{G}$  is trained to generate "realistic" images to fool the discriminator  $\mathbf{D}$ , while the discriminator  $\mathbf{D}$  is trained to tell the generated images from groundtruth images. Specifically, we employ LSGAN loss [Mao et al. 2017] as the adversarial loss to optimize the discriminator  $\mathbf{D}$ :

$$\mathcal{L}_{GAN}(\mathbf{D}) = (\hat{r} - 1)^2 + r^2, \quad (12)$$

where  $\hat{r}, r$  is the discriminator classification output when input the groundtruth image  $\hat{y}$  and the generated rendering  $y$ , respectively. We additionally use a color loss, a perceptual loss [Johnson et al. 2016] and a feature matching loss [Wang et al. 2018b]:

$$\mathcal{L}_G = \mathcal{L}_{GAN}(\mathbf{G}) + \lambda_C \mathcal{L}_C + \lambda_P \mathcal{L}_P + \lambda_{FM} \mathcal{L}_{FM}, \quad (13)$$

where  $\mathcal{L}_{GAN}(\mathbf{G}) = (r - 1)^2$  is the adversarial loss,  $\mathcal{L}_C$  is the color loss,  $\mathcal{L}_P$  is the perceptual loss and  $\mathcal{L}_{FM}$  is the feature matching loss. The weights  $\lambda_C, \lambda_P, \lambda_{FM}$  of each loss are set to (100, 10, 1) in all our experiments empirically. The color loss is a  $L_1$  per-pixel loss to minimize the difference from generated images  $y$  and groundtruth images  $\hat{y}$ :  $\mathcal{L}_C = \|y - \hat{y}\|_1$ . We tried higher weights (x10) on mouth, yep mouth-related errors drop, but full-image errors rise. Considering the full-image generation task, we choose equal weight.

For perceptual loss, we adopt a VGG19 network [Simonyan and Zisserman 2014] to extract the perceptual features from  $\hat{y}$  and  $y$  and minimize their  $L_1$  distance:

$$\mathcal{L}_P = \sum_{i \in \mathcal{S}} \|\phi^{(i)}(y) - \phi^{(i)}(\hat{y})\|_1, \quad (14)$$

where  $\mathcal{S} = \{1, 6, 11, 20, 29\}$  denotes the layers we use and  $\phi^{(i)}$  denotes the  $i$ -th layer. Finally, to improve the training speed and



Fig. 6. A gallery of our audio-driven talking-head animation results. Given an arbitrary audio stream, our method first generates personalized facial dynamics, head poses, and upper body motions and then synthesizes photorealistic renderings from these predictions. Please refer to the supplementary video for full sequences. Video (at upper left corner) *Obama* ©Barack Obama Foundation (public domain). Video *May* ©UK government (Open Government Licence). Video *Nadella* ©IEEE Computer Society (public domain). Video *Trump* ©White House (public domain). Video (at bottom right corner) *Obama* ©White House (public domain).

stability, we adopt a feature matching loss:

$$\mathcal{L}_{FM} = \sum_{i=1}^L ||r - \hat{r}||_1 \quad (15)$$

where  $L$  is the number of spatial layers in discriminator  $\mathbf{D}$ . The  $L_1$ -based feature matching loss is designed to match the statistics of features extracted by the discriminator from  $\hat{y}$  and  $y$ .

### 4.3 Training Setup and Parameters

All our models are trained on PyTorch (Python) [Paszke et al. 2019] using Adam optimizer with hyper-parameters  $(\beta_1, \beta_2) = (0.9, 0.999)$ . The learning rate is set to  $10^{-4}$  and linearly decay to  $10^{-5}$  in all experiments. The APC model contains 4.064M parameters, the mouth-related position predictor contains 3.064M parameters, the head pose estimator contains 4.267M parameters and the renderer contains 76.204M parameters.

We train the first three models on a Nvidia 1080Ti GPU and takes about (11, 0.5, 5) hours in total (200, 200, 200) epochs with a batch size of 32, respectively. The photorealistic images renderer is trained on 4 Nvidia 1080Ti GPUs for an average of 22 hours in 60 epochs and a batch size of 8. During testing, we select all models with minimized validation loss.

### 4.4 Real-Time Animation

We implement and test our real-time animation system using C++ on a desktop PC with an Intel Core i7-9700K CPU (32 GB RAM) and an NVIDIA GeForce RTX 2080 (8 GB RAM). The deep speech representation extraction module takes around 2.9 ms for inference (1.4 ms for APC model forward pass and 1.5 ms for manifold projection). Learning facial dynamics from audio representations via

3-layer LSTM and MLP networks takes around 2.5 ms. Besides, the Audio2Mouth module leads to around 300 ms latency for obtaining 18 frames of future audio information. We then use Tensorrt to accelerate the last two models. Specifically, the head poses estimation model takes 4.4 ms and the photorealistic renderer takes 20.1 ms after acceleration. Time of memory copying between CPU and GPU has already been included here. Therefore, the entire system takes about 27.4 ms for inference at over 30 FPS with 300 ms latency.

*Discussion.* Here we discuss the run time with related work Zhou et al. [2020] and Thies et al. [2020]. We emphasize that our system is the first *realization* of an end-to-end live system for photorealistic audio-driven talking-head animation and step further to practical applications, considering that these papers haven't shown an actual live demo. Besides, Zhou et al. [2020] is not designed for live streaming generation. The self-attention network [Vaswani et al. 2017] in speaker-aware animation works as a post weighted combination of previous landmarks predictions and is inappropriate for live applications like video conferencing, requiring a low latency. Thies et al. [2020] will meet more difficulties. Their method is restricted to the target video length, and therefore a sufficiently long target video without obstruction is required, which is hard to acquire. To generate longer frames, an additional heuristic schedule to select proper candidate frames is needed. Also, pose-audio inconsistency appears for the lacking control on head motion (Section 2). These factors result in obstacles in the live implementation.

## 5 RESULTS

Our live speech portraits method generates personalized and photorealistic talking-head animation from audio input in real-time. We recommend readers watch the supplementary video.





Fig. 7. Our method allows generating pose controllable results. We demonstrate results with different poses here. The number at the left of the image denotes the frame index of the video. Pose 1 is generated by our model, and Pose 3 is sampled from the training set. To further evaluate the head poses controllability, we mirror the head poses 1 and 3 to generate head pose 2 and 4. Pose 5 is self-defined. Please refer to the supplementary video for full sequences. Video *May* ©UK government (Open Government Licence).

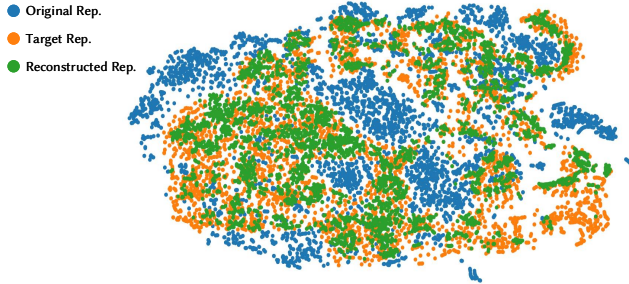


Fig. 8. t-SNE visualization of Manifold Projection. As shown in the legend, original, target and reconstructed representations are marked using different colors.

In the following, we present the results of our approach, evaluate the design of our method both qualitatively and quantitatively, compare to the state-of-the-art techniques, and show results of a user study. We further demonstrate the potentials on several applications, e.g., dubbing, video conferencing and virtual avatars.

### 5.1 Qualitative Evaluation

Figure 6 shows a gallery of our results. Our approach allows animating target portraits driven by audio sequences while preserving the personal talking styles. Our method produces facial dynamics, natural head poses, and upper body motions and synthesizes temporally coherent renderings in high-fidelity, e.g., clear teeth textures. It also successfully works on subjects with long hair, glasses, and earrings.

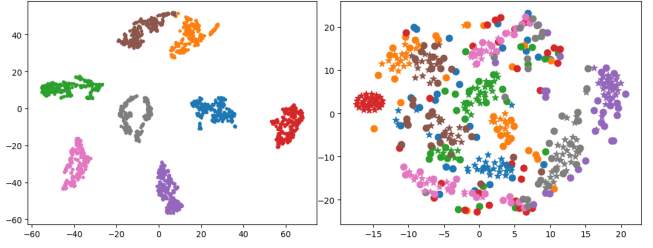


Fig. 9. t-SNE visualization of head pose generation. Different targets are marked using different colors. Left: visualization of generated poses. Right: visualization of generated poses and head poses from training corpus. Marker *solid dot* denotes training corpus poses and marker *star* denotes generated poses.

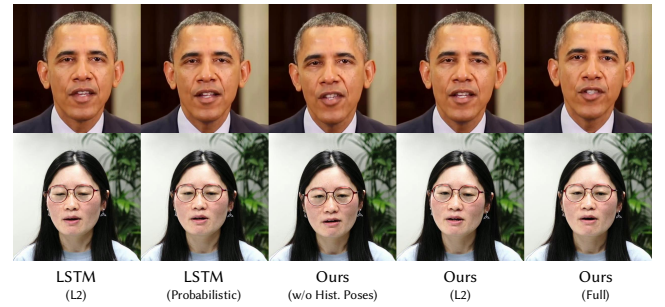


Fig. 10. Qualitatively evaluation of head pose estimation. We compare our model design with several alternative variants. Please refer to the supplementary video for full sequences. Video *Obama* ©White House (public domain).

Figure 1 and the supplementary video show examples of our real-time animation system and results of different people driven by the same utterance. Results prove that we preserve their own talking styles in the training video.

Figure 7 demonstrates the pose controllability of our method. We synthesize results with different head poses, either generated by our system or sampled from the training set (Columns 1 and 3). We further evaluate our model with mirrors of the previous two poses and a self-defined pose (Columns 2, 4, and 5). Even though these challenging poses partly lay outside the training corpus and therefore may lead to artifacts, our system still generates correct lip motions and temporally smooth renderings.

We analyze the effectiveness of manifold projection in Figure 8. The operation is designed to project the original speech representations to the target speech space by minimizing the reconstruction loss. As shown in the figure, the reconstructed features (green) are much closer to the target speech space than the original wild features (blue). More quantitative comparison results of the manifold projection can be seen in our supplementary video. We found that using manifold projection generates more accurate lip synchronization than not using it, especially when encountering the audio of different genders or foreign languages.

Figure 9 depicts a qualitative evaluation of the generated head poses using t-SNE visualization. We select 8 different targets and

Table 1. Quantitative evaluation of input time delays. Validation losses are computed as Euclidean distance of landmarks.

Time Delay	0ms	50ms	100ms	150ms
Val Loss (mm)	5.309	5.280	5.117	5.335
Time Delay	200ms	300ms	400ms	500ms
Val Loss (mm)	5.160	<b>4.916</b>	5.248	5.539

denote them with different colors. In Figure 9 (left), we visualize the generated head poses of these targets from the same input audio. It can be seen that our predictions for one person lie in a nearby region and are far away from different targets. We further analyze the connections between generated poses and training poses in Figure 9 (right). *Solid dot* denotes training corpus poses and *star* denote our predictions. We downsample the training poses using K-means to 30 to reduce data size. The visualization demonstrates that our predictions locate closely to training corpus poses.

We also qualitatively evaluate the design of the head pose estimation module from audio. Results are shown in Figure 10. We recommend readers to watch the supplementary video for better visual comparisons. To confirm our proposed two prior assumptions, here we perform an ablative study of the input, architecture, and loss design by training and testing four alternative variants: "*LSTM (L2)*" (LSTM network trained using L2 loss), "*LSTM (Probabilistic)*" (LSTM network trained using probabilistic loss), "*Ours (w/o Hist. Poses)*" (Our architecture without history poses input), "*Ours (L2)*" (Our architecture trained using L2 loss). A quantitative evaluation of head pose estimation can be found in Section 5.2.

The variants "*LSTM (L2)*" and "*LSTM (Probabilistic)*" generate more temporal jitters than other variants, which indicates that directly using LSTM architecture may not be a good choice in our task. A reasonable explanation is that RNN is easy to overfit on a small dataset, which is an around 3-minute length video in our case. On the other hand, LSTMs have a theoretically infinite receptive field of the history information depending on the forget and memory mechanism. Our architecture has a fixed receptive field which intuitively becomes more robust to long history information and will not easily overfit.

The variant "*Ours (w/o Hist. Poses)*" also performs worse than our full model. It tends to generate unchanged poses in the *Obama* video and struggled poses in the woman video (see the supplementary video). Without modeling the history poses, the model is trained to learn a mapping from audio to the head poses solely. Therefore, the network generates pose with the most probability of the current audio clip, which may be far away from adjacent ones, leading to time-incoherent results.

The variant "*Ours (L2)*" generates best results among all variants but there is still room for improvement in the term of time consistency and motion variation. The only difference is that this variant replaces the probabilistic loss with the L2 distance loss. It makes the problem a regression problem, which means the model needs to find the deterministic position given history poses and audio information. Therefore, the model struggles to find the best balance point between audio and history poses, resulting in time-inconsistency and minor motion variations. A probabilistic model

Table 2. Quantitative evaluation of head pose prediction. We compare our approach with alternative methods in Section 5.1 and Zhou et al. [2020]. ↓ denotes lower is better.

Methods	D-L ↓	D-V ↓	D-Rot/Pos ↓
LSTM (L2)	4.9%	1.1%	6.9/12.2%
LSTM (Probabilistic)	4.9%	1.0%	6.7/11.6%
Ours (w/o Hist. Poses)	3.9%	0.9%	4.2/8.9%
Ours (L2)	4.5%	1.1%	3.7/9.2%
[Zhou et al. 2020]	4.6%	0.9%	6.1/10.1%
Ours (Full)	<b>3.6%</b>	<b>0.8%</b>	<b>3.6/8.9%</b>

is suitable to handle these ambiguities and finally achieves the best results both in time consistency and motion richness.

## 5.2 Quantitative Evaluation

Here, we perform a thorough quantitative evaluation of the proposed method. Table 1 evaluates the importance of time latency of input audio. Choosing a proper latency is important for a real-time system because it greatly affects the user experience. In this experiment, we train the audio-to-mouth network with different time delays on a 30-minutes subset of the *Obama Weekly Address* dataset [Suwajanakorn et al. 2017]. The subset was split into a training dataset (80%) and a validation dataset (20%). Validation losses are computed as Euclidean distance between predicted mouth-related landmarks positions and the tracked groundtruth. As can be seen, the time delay of 300 ms (18 frames) gives minimized validation loss. Too short or too long delays both lead to performance reduction. When the network fails to have access to a certain range of future information (smaller than 300 ms), it cannot model the coarticulation. Too long delays (longer than 300 ms) always cover several phonemes and introduce redundant information. In most real-time scenarios, e.g., video conferencing, 300 ms is a tolerable latency. Therefore, we use this latency in all experiments.

Now we perform quantitative evaluation of our head pose estimation module. We compare our method with using four alternative variants pre-mentioned in Section 5.1 as well as Zhou et al. [2020]. We created a test set composed of 6 different videos (average lasts for 45 seconds each clip) and the speeches and videos are unseen during training. Specifically, We evaluate these methods on the testing set by computing the metrics **D-L**, **D-V** and **D-Rot/Pos** defined in [Zhou et al. 2020]. **D-L** and **D-V** denote the normalized Euclidean position and velocity difference between predictions and the groundtruth. **D-Rot/Pos** denotes the rotation angle differences and normalized translation distances. Lower is better for all these metrics.

Table 2 reports the evaluation results. Our model obviously outperforms other alternative variants, tallying with the subjective evaluation in Section 5.1. In particular, we observe that the LSTM-based variants produce higher error than other variants ((see "*LSTM (L2)*" and "*LSTM (Probabilistic)*"). This results confirms that LSTM is prone to overfit in the training set. Replacing the probabilistic modeling with L2-regression generates slightly worse results (see "*Ours (L2)*"). Our full method generates similar head motion with

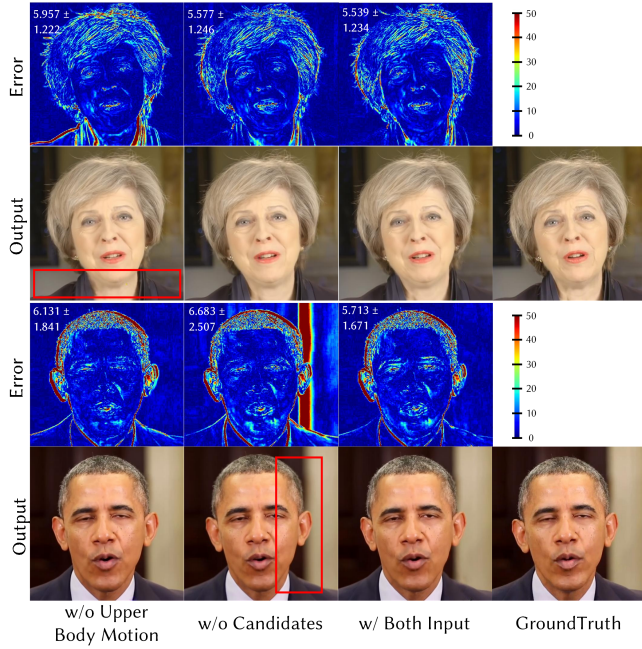


Fig. 11. Quantitative evaluation of the conditional input of the renderer. Video *May* ©UK government (Open Government Licence). Video *Obama* ©White House (public domain).

respect to the groundtruth, confirming that we learn the distribution of the target person. We also compare with Zhou et al. [2020] which learn speaker-aware head motions. Results demonstrate that our method outperforms their method.

Here, we quantitatively evaluate the photorealistic renderer. Results can be found in Figure 11-13. We evaluate the model on three videos. The first three minutes of these videos are used as training data while the rest work as the testing set unless stated otherwise. Note that we test models with groundtruth head poses for numerical evaluation. We report the numerical errors and corresponding standard deviation of average L1-photometric loss (range of 0-255), Peak Signal to Noise Ratio (PSNR), Structural Similarity (SSIM), and deep perceptual distance (LPIPS) [Zhang et al. 2018] in Table 3. The standard deviation indicates the temporal stability around the mean loss - higher variance means stronger incoherence. The L1 loss and its standard deviation are shown directly on the top-left corner of error heat-maps. We refer readers to the supplementary video for better visualization. A large-version generator works as a baseline design to test with different ablative conditions.

Figure 11 and Table 3 (top) show the importance of our two conditional input. Removing upper body motions leads to strong shakiness especially around the neck and shoulders. It is hard for the network to synthesize time-consistent upper body motions without this strong location condition. Removing the candidate image set also leads to significant performance decrease on backgrounds and facial contours, especially when the training videos includes camera motions. For example, Obama video in Figure 11 consists of several different camera motions and confuses the model to learn

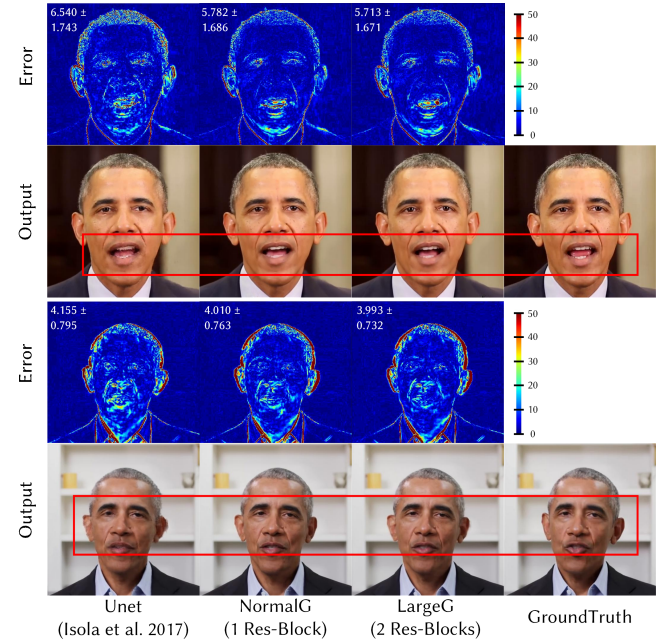


Fig. 12. Quantitative evaluation of the architecture design of the renderer. Video (Upper) *Obama* ©White House (public domain). Video (Lower) *Obama* ©Barack Obama Foundation (public domain).

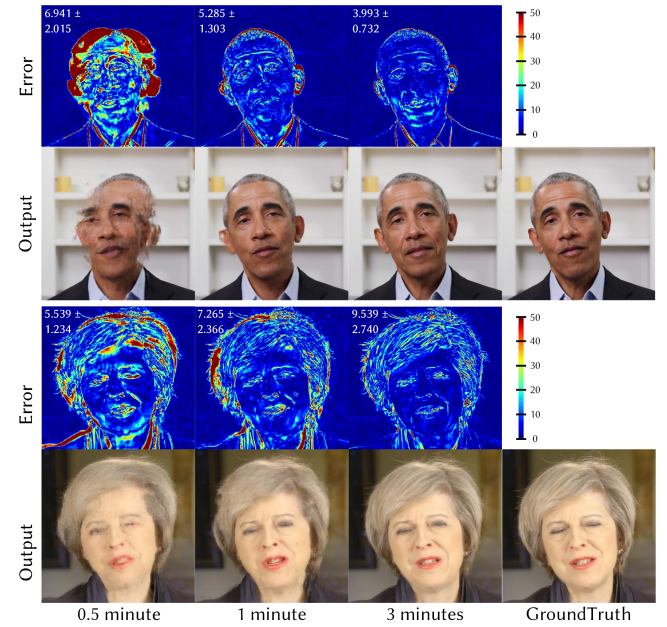


Fig. 13. Quantitative evaluation of training dataset size. Video *Obama* ©Barack Obama Foundation (public domain). Video *May* ©UK government (Open Government Licence).

the one-to-many mappings. The candidate image set works as a clue to tell the model what the scene is and guides the network



Table 3. Quantitative evaluation of photorealistic renderer design. We evaluate the results of input condition (top), architecture (middle), and training dataset size (bottom). ↓ denotes lower is better and ↑ denotes higher is better.

Target/Term/Method		L1 ↓	PSNR ↑	SSIM ↑	LPIPS ↓ × 10
Obama1	w/o U.B.M.	6.131 ± 1.841	25.364 ± 2.259	0.850 ± 0.043	0.704 ± 0.407
	w/o Cand.	6.683 ± 2.507	24.964 ± 2.483	0.845 ± 0.048	0.746 ± 0.422
	w/ Both	<b>5.713</b> ± 1.671	<b>26.006</b> ± 2.275	<b>0.862</b> ± 0.038	<b>0.698</b> ± 0.348
Obama2	w/o U.B.M.	7.634 ± 5.105	23.143 ± 2.758	0.843 ± 0.102	1.136 ± 1.279
	w/o Cand.	4.006 ± 0.754	<b>27.927</b> ± 2.228	0.917 ± 0.020	<b>0.366</b> ± 0.111
	w/ Both	<b>3.993</b> ± 0.732	27.576 ± 2.174	<b>0.926</b> ± 0.020	0.370 ± 0.103
May	w/o U.B.M.	5.957 ± 1.322	27.171 ± 1.820	0.807 ± 0.037	0.790 ± 0.260
	w/o Cand.	5.577 ± 1.246	27.979 ± 1.683	0.823 ± 0.033	0.747 ± 0.280
	w/ Both	<b>5.539</b> ± 1.234	<b>28.044</b> ± 1.782	<b>0.823</b> ± 0.035	<b>0.746</b> ± 0.209
Target/Term/Method		L1 ↓	PSNR ↑	SSIM ↑	LPIPS ↓ × 10
Obama1	Unet	6.540 ± 1.743	25.087 ± 2.046	0.832 ± 0.035	0.875 ± 0.395
	Normal	5.782 ± 1.687	25.920 ± 2.229	0.861 ± 0.038	0.727 ± 0.346
	Large	<b>5.713</b> ± 1.671	<b>26.006</b> ± 2.275	<b>0.862</b> ± 0.038	<b>0.698</b> ± 0.348
Obama2	Unet	4.155 ± 0.794	27.414 ± 2.182	0.907 ± 0.022	0.479 ± 0.124
	Normal	4.010 ± 0.763	27.406 ± 2.211	0.912 ± 0.021	0.376 ± 0.106
	Large	<b>3.993</b> ± 0.732	<b>27.576</b> ± 2.174	<b>0.926</b> ± 0.020	<b>0.370</b> ± 0.103
May	Unet	6.007 ± 1.709	27.528 ± 1.835	0.806 ± 0.037	1.018 ± 0.389
	Normal	5.578 ± 1.694	27.924 ± 1.882	0.818 ± 0.049	0.828 ± 0.449
	Large	<b>5.539</b> ± 1.234	<b>28.044</b> ± 1.782	<b>0.823</b> ± 0.035	<b>0.746</b> ± 0.209
Target/Term/Method		L1 ↓	PSNR ↑	SSIM ↑	LPIPS ↓ × 10
Obama1	0.5 mins	9.403 ± 3.068	22.481 ± 2.166	0.788 ± 0.042	1.421 ± 0.533
	1 min	8.259 ± 2.552	23.939 ± 2.427	0.812 ± 0.046	1.233 ± 0.523
	3 mins	<b>5.713</b> ± 1.671	<b>26.006</b> ± 2.275	<b>0.862</b> ± 0.038	<b>0.698</b> ± 0.348
Obama2	0.5 mins	6.941 ± 2.015	22.960 ± 2.195	0.858 ± 0.030	0.945 ± 0.382
	1 min	5.285 ± 1.303	24.873 ± 2.010	0.888 ± 0.024	0.585 ± 0.239
	3 mins	<b>3.993</b> ± 0.732	<b>27.576</b> ± 2.174	<b>0.926</b> ± 0.020	<b>0.370</b> ± 0.103
May	0.5 mins	9.539 ± 2.740	23.432 ± 2.087	0.739 ± 0.033	1.731 ± 0.582
	1 min	7.265 ± 2.366	25.875 ± 2.086	0.739 ± 0.050	1.175 ± 0.685
	3 mins	<b>5.539</b> ± 1.234	<b>28.044</b> ± 1.782	<b>0.823</b> ± 0.035	<b>0.746</b> ± 0.209

synthesize the right and consistent background. Besides, it also relieve the network pressure to synthesize high-fidelity details since it has included details as input.

Figure 12 and Table 3 (middle) evaluate the generator architecture design. We compare our model with a baseline model Unet [Isola et al. 2017] and a larger model with 2 residual blocks each layer. We found significant performance degradation using Unet (see red boxes in Figure 12). They fail to synthesize clear teeth, ears and other facial details compared to other architectures. Increasing the res-block in each layer takes over 59% more parameters (121.790M vs 76.204M) but obtains no significant improvements on image quality. Considering the balance of performance and efficiency, we use normal architecture (1 Res-block) as our default architecture.

Finally we evaluate the importance of training dataset size. In this experiment, we train the model using 0.5/1/3 minutes frames (see Figure 13). Larger training set generates better results since it covers more pose and expression variations (see Table 3 (bottom)), and the best results are achieved using the full training set.

### 5.3 Comparisons to the State-of-the-Art

Now we compare our method with state-of-the-art audio-driven talking-head animation techniques. All the test input audio sequences are unseen during training. We strongly recommend readers watch the supplementary video for better comparisons.

*Comparisons to image-based generation methods.* We first compare our approach with image-based generation methods for synthesizing talking-head videos. Specially, we compare with Chen et al. [2019], Vougioukas et al. [2019] and Zhou et al. [2020]. Figure 14 (upper the solid line) shows results driven by wild audios, and Figure 14 (lower the solid line) shows the results driven by voices of the target person in images. These methods are trained to generalize to unseen faces and therefore lack personal talking styles (they tend to generate the same lip motions for everyone) and facial details. Chen et al. [2019] and Vougioukas et al. [2019] generate the talking videos in a cropped and normalized face region and therefore fail to handle head poses. Zhou et al. [2020] generate speaker-aware talking-head videos but not the target person style. They warp both the background and the talking-head, giving the



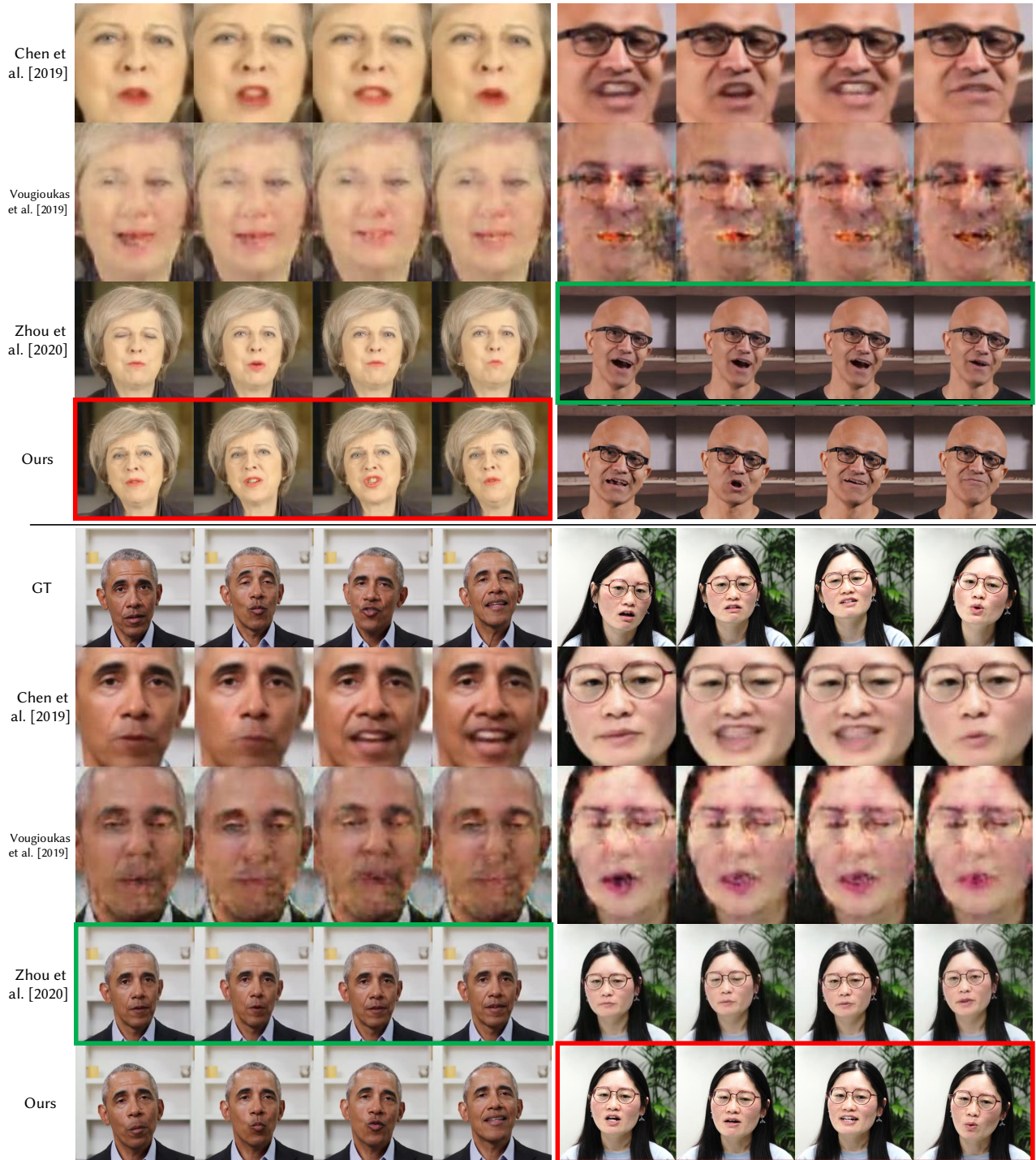


Fig. 14. Comparison with state-of-the-art image-based generation methods. Best viewed in the supplementary video. Upper the solid line: test using wild audio streams. Lower the solid line: test using groundtruth audio streams in the validation set. Video May ©UK government (Open Government Licence). Video Nadella ©IEEE Computer Society (public domain). Video Obama ©Barack Obama Foundation (public domain).

impression that the foreground head carries the background and moves together (see the green boxes). Moreover, the mouth of the portrait tends to twist and the synthesized region is blurred. Compared against these methods, our method successfully captures the talking style of the target personal and synthesizes sharper images with higher fidelity, e.g., wrinkles and teeth (see the red boxes). We also note that the head pose generated by our method is different from the groundtruth but exhibits a similar distribution in the training set.

We further compare with Chen et al. [2020a] in the supplementary video, which generate talking faces with rhythmic head movements. We notice that Chen et al. [2020a] do not disentangle the head motion and the background, resulting in an associated movement of the both. Besides, our method keeps the facial details and generates facial images with higher quality.

**Comparisons to video-based editing methods.** We compare our approach against video-based editing methods. In particular, we compare against Suwajanakorn et al. [2017] and Thies et al. [2020]. Please watch the supplementary video for the results. These two methods both synthesize a lower face patch and blend it into the target frame. They rely on additional facial tracking algorithm to provide accurate and stable rigid head motions and 3D mouth locations. Our method directly generates full head renderings as well as backgrounds. Suwajanakorn et al. [2017] synthesize high-quality talking videos of Obama, which is trained on a huge amount of his weekly address videos (17 hours). *Neural Voice Puppetry* [Thies et al. 2020] additionally trains person-specific blendshape basis using 2-3 minute videos aside from training a general model basis on around 3 hours videos. Our method generates visually comparable and controllable results while using only around 3 minutes videos for training. Considering the intrinsic features of video-based editing methods, which limits the application scenarios, our approach is more applicable to other target person. Last but not least, our system runs in real-time.

**Comparisons to model-based methods.** We also compare our approach with model-based methods [Cudeiro et al. 2019; Karras et al. 2017; Taylor et al. 2017; Zhou et al. 2018]. These methods focus on learning a 3D face mesh or rigging parameters from audio. They usually require a 4D training corpus obtained by a high-cost vision-based capture system or rigging parameters with artist interventions, while our method uses sparse 3D landmarks as an intermediate representation and works on internet videos. Also, our method generates photorealistic results.

## 5.4 User Study

We finally conduct three user studies to quantitatively evaluate the quality of our method. We compare our results with state-of-the-art open-source methods [Chen et al. 2019; Vougioukas et al. 2019; Zhou et al. 2020]. We prepare 20 audio clips for each method and generate 80 video clips in total. All audio clips are wild and unseen in the training set. These user studies are web-based, and 48 participants with computer science backgrounds finished our questionnaire. During the study, the web page shows one video at a time

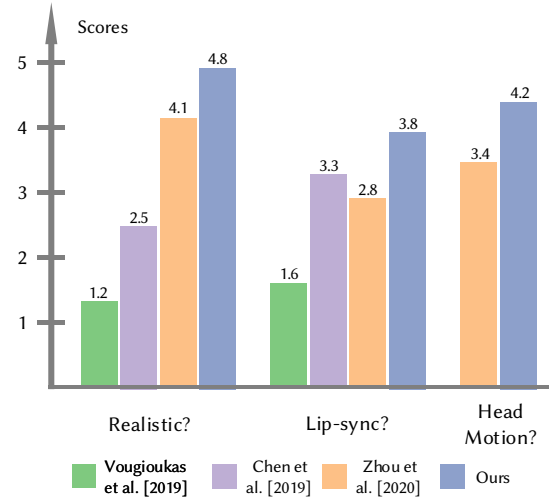


Fig. 15. User study results for three different tasks.



(a) Dubbing. The dubber (left) is dubbing for the targets (right).



(b) Video Conferencing.



(c) Virtual Avatars.

Fig. 16. Applications. Our system can be applied on many applications, e.g., dubbing, video conferencing and virtual avatars. Please refer to the supplementary video for full sequences. Video *May* ©UK government (Open Government Licence). Video *Trump* ©White House (public domain). Video *Nadella* ©IEEE Computer Society (public domain). Video *Obama* ©White House (public domain).

in randomized order, and the participant is asked to evaluate the video w.r.t three statements: 'The video looks realistic to me.', 'The mouth motion is sync with the audio.' and 'The head motion of the portrait is natural.' on a scale between 1 to 5 (5-strongly agree, 4-agree, 3-neither agree nor disagree, 2-disagree, 1-strongly disagree). Figure 15 shows the average scores of different methods on three statements. The head pose evaluation is only conducted on Zhou et al. [2020] since the other two methods animate the cropped face and hardly have any pose motions.

As can be seen, our method achieves the highest scores over three tasks. For the first task, our method has the highest score of 4.8, which means our results are the most photorealistic. Meanwhile, we have the best results in terms of lip-synchronization. We believe that is because our manifold projection works which improves the generalization. Finally, we compare the head motion against Zhou et al. [2020]. Our method generates more natural head motions because we model the target-aware head motion.

### 5.5 Applications

Our method synthesizes photorealistic talking-head animation from audio streams in real-time, thus having a wide range of applications, e.g., dubbing, video conferencing, and virtual avatars. We refer readers to our supplementary video. Figure 16 demonstrates the potential applications. On the top of the figure, we show the audio-driven dubbing results of the target person. Compared to video-based dubbing methods [Kim et al. 2019], our method avoids generating implausible facial dynamics of the target person because we model the personal characteristics.

Video conferencing is another application (see Figure 16 (b)). In scenarios that people cannot deliver visual signals, e.g., they are outdoor or have limited bandwidth, our method can generate high-fidelity video frames only driven by audio in real-time.

We finally demonstrate our potentials in virtual avatars such as virtual anchors, assistants. Our supplementary video shows a real-time demo of virtual avatars, e.g., the portrait of Theresa May is animated to sing a song driven by the actor's voice. Figure 16 (c) shows results driven by the Text-to-Speech (TTS) system. The supplementary video also includes a comparison with Zhou et al. [2020] and Thies et al. [2020]. Our method generates more realistic frames and more accurate lip synchronization.

## 6 CONCLUSION

We presented a deep learning approach for generating photorealistic talking-head animation of the target person in real-time. Our method can handle new audio clips not seen during training and still synthesize personalized video frames. The full system is only needed to be trained on a several-minute length video. Our pipeline contains three stages: deep audio features extraction, facial dynamics and motion generation, and photorealistic image synthesis. The first stage includes a manifold projection on deep audio features, which helps generalize to wild audio. In the second stage, facial dynamics, head poses, and upper body motions are generated. An autoregressive probabilistic head pose estimation network is trained to learn the target actor's pose distribution. This network led to personalized head pose generation and avoided the potential performance degradation of the subsequent neural renderer. Finally, we generated intermediate feature maps from these predictions and sent them with a candidate image set to an image-to-image translation network to synthesize video frames. Thorough experiments and a user study show that our method outperforms the state-of-the-art techniques both qualitatively and quantitatively. Our method can be applied in many scenarios, especially the ones required to run in real-time, like dubbing, video conferencing, and virtual avatars. We hope this work could open a new avenue for future researches

in this field.

*Limitations and Future Work.* While we have demonstrated impressive results of our method in a wide variety of scenarios, there are still several limitations to our approach. Our real-time system does not always capture well with the plosive and nasal consonants, e.g., /p/b/m/. The reasons behind are various. First, /p/b/m/ usually sounds in low volumes and may be ignored by the fore-end as environmental noise. Second, our live system runs at over 30 FPS, and it may miss these short sounds. It also fails to capture the speakings at a very fast speed, like a quarrel scenario. Our offline results (60FPS) are better, which partly verifies our supposition. Applying model pruning is a promising solution to decrease the parameters and increase the running speed. Besides, the spectrum construction we use tends to miss those short phonemes, which can be tackled by using pure deep features, e.g., wav2vec [Schneider et al. 2019]. The face tracking algorithm we used is not state-of-the-art, we believe that better reconstruction leads to better lip-sync results.

Similar to most learning-based methods, the style of the generated videos is restricted to the training corpus. Our method preserves the talking styles in training sequences (3-5 minutes) via manifold projection, a domain transfer method to find the most similar samples. This mechanism alleviates this problem to some extent. We believe that a complete solution is to apply a perfect audio disentanglement algorithm like [Qian et al. 2020] to split each components, i.e., content, pitch, timbre and rhythm, and find the best mapping of these components.

Emotional audio may generate unsatisfied results when the model was trained on a neutral-style video. Our method cannot directly control the emotion of the generated videos. Recent work [Ji et al. 2021] shows promising emotion manipulation results when training on an emotional dataset. It would be interesting to apply such progress to our system.

Although we successfully handle the shadows and the lighting reflections when people swing their heads, we still can not explicitly control these parameters. Relighting techniques [Sun et al. 2019] can be applied directly to our rendering results to control the environment lighting. Gestures are another important component for people to deliver expression. We are looking forward to future work on gesture generation driven by audio.

## 7 ETHICAL CONSIDERATIONS

With the rapid development of 'Deepfake' techniques, the threshold for people to synthesize fake videos of arbitrary person is becoming lower. In most kinds, they facilitate the movie and entertainment industry and reduces the bandwidth of video streaming by sending the audio signals only. However, these techniques can be misused. Due to the fact that it's more difficult for people to distinguish fake videos, the algorithm may be utilized to spread misinformation or obtain illegal profits. Our method achieves real-time photorealistic talking-head animation and only needs to be trained on a several-minute length video, which can be easily found on the Internet. For non-celebrities, their faces and voices are harder to recognize than celebrities, and therefore generating their fake videos is

more deceptive. Potential solutions like digital face forensics methods [Rössler et al. 2018; Rossler et al. 2019] to detect deepfakes must be considered. We hope the public be aware of the potential risks of the misuse of new techniques.

## ACKNOWLEDGMENTS

We would like to thank Shuaizhen Jing for the help with the Tensorrt implementation. We are grateful to Qingqing Tian for the facial capture. Yuanxun Lu would also like to thank Xinya Ji for her mental support and proof-reading during the project. This work was supported by the NSFC grant 62025108, 61627804 and Leading Technology of Jiangsu Basic Research Plan (BK20192003).

## REFERENCES

- Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common Voice: A Massively-Multilingual Speech Corpus. In *Proceedings of The 12th Language Resources and Evaluation Conference*. 4218–4222.
- Jean-Yves Bouguet et al. 2001. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel corporation* 5, 1–10 (2001), 4.
- G. Bradski. 2000. The OpenCV Library. *Dr. Dobbs's Journal of Software Tools* (2000).
- Matthew Brand. 1999. Voice Puppetry. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., USA, 21–28. <https://doi.org/10.1145/311535.311537>
- Christoph Bregler, Michele Covell, and Malcolm Slaney. 1997. Video Rewrite: Driving Visual Speech with Audio. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '97)*. ACM Press/Addison-Wesley Publishing Co., USA, 353–360. <https://doi.org/10.1145/258734.258880>
- Chen Cao, Yanlin Weng, Stephen Lin, and Kun Zhou. 2013. 3D shape regression for real-time facial animation. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.
- Chen Cao, Hongzhi Wu, Yanlin Weng, Tianjia Shao, and Kun Zhou. 2016. Real-time facial animation with image-based dynamic avatars. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.
- Lele Chen, Guofeng Cui, Celong Liu, Zhong Li, Ziyi Kou, Yi Xu, and Chenliang Xu. 2020a. Talking-head Generation with Rhythmic Head Motion. In *European Conference on Computer Vision*. Springer, 35–51.
- 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*. 7832–7841.
- Shu-Yu Chen, Wanchao Su, Lin Gao, Shihong Xia, and Hongbo Fu. 2020c. DeepFace-Drawing: Deep generation of face images from sketches. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 72–1.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020b. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.
- Jan Chorowski, Ron J Weiss, Samy Bengio, and Aaron van den Oord. 2019. Unsupervised speech representation learning using wavenet autoencoders. *IEEE/ACM transactions on audio, speech, and language processing* 27, 12 (2019), 2041–2053.
- Joon Son Chung, Amir Jamaludin, and Andrew Zisserman. 2017. You said that? *arXiv preprint arXiv:1705.02966* (2017).
- Yu-An Chung and James Glass. 2020. Generative pre-training for speech with autoregressive predictive coding. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 3497–3501.
- Daniel Cudeiro, Timo Bolkart, Cassidy Laidlaw, Anurag Ranjan, and Michael J Black. 2019. Capture, learning, and synthesis of 3D speaking styles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10101–10111.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
- Mohamed Elgharib, Mohit Mendiratta, Justus Thies, Matthias Niessner, Hans-Peter Seidel, Ayush Tewari, Vladislav Golyanik, and Christian Theobalt. 2020. Egocentric videoconferencing. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–16.
- Patrick Esser, Ekaterina Sutter, and Björn Ommer. 2018. A variational u-net for conditional appearance and shape generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8857–8866.
- Tony Ezzat, Gadi Geiger, and Tomaso Poggio. 2002. Trainable videorealistic speech animation. *ACM Transactions on Graphics (TOG)* 21, 3 (2002), 388–398.
- Bo Fan, Lijuan Wang, Frank K Soong, and Lei Xie. 2015. Photo-real talking head with deep bidirectional LSTM. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4884–4888.
- Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. 2019. Text-based editing of talking-head video. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–14.
- Pablo Garrido, Levi Valgaerts, Hamid Sarmadi, Ingmar Steiner, Kiran Varanasi, Patrick Perez, and Christian Theobalt. 2015. Vdub: Modifying face video of actors for plausible visual alignment to a dubbed audio track. In *Computer graphics forum*, Vol. 34. Wiley Online Library, 193–204.
- Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* (2014).
- David Greenwood. 2018. *Predicting Head Pose From Speech*. Ph.D. Dissertation. University of East Anglia.
- David Greenwood, Iain Matthews, and Stephen Laycock. 2018. Joint Learning of Facial Expression and Head Pose from Speech. *Proc. Interspeech 2018* (2018), 2484–2488.
- Xintong Han, Zuxuan Wu, Weilin Huang, Matthew R Scott, and Larry S Davis. 2019. Finet: Compatible and diverse fashion image inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4481–4491.
- Awani Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567* (2014).
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9729–9738.
- Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. 2020. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–14.
- Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1125–1134.
- Xinya Ji, Hang Zhou, Kaisiyuan Wang, Wayne Wu, Chen Change Loy, Xun Cao, and Feng Xu. 2021. Audio-Driven Emotional Video Portraits. *arXiv preprint arXiv:2104.07452* (2021).
- Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*. Springer, 694–711.
- Tero Karras, Timo Aila, Samuli Laine, Antti Herva, and Jaakko Lehtinen. 2017. Audio-driven facial animation by joint end-to-end learning of pose and emotion. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–12.
- Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4401–4410.
- Hyeonwoo Kim, Mohamed Elgharib, Michael Zollhöfer, Hans-Peter Seidel, Thabo Beeler, Christian Richardt, and Christian Theobalt. 2019. Neural style-preserving visual dubbing. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–13.
- Hyeonwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. 2018. Deep video portraits. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–14.
- Alexander H Liu, Yu-An Chung, and James Glass. 2020. Non-Autoregressive Predictive Coding for Learning Speech Representations from Local Dependencies. *arXiv preprint arXiv:2011.00406* (2020).
- Yilong Liu, Feng Xu, Jinxiang Chai, Xin Tong, Lijuan Wang, and Qiang Huo. 2015. Video-audio driven real-time facial animation. *ACM Transactions on Graphics (TOG)* 34, 6 (2015), 1–10.
- Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2794–2802.
- Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohamed Elgharib, Pascal Fua, Hans-Peter Seidel, Helge Rhodin, Gerard Pons-Moll, and Christian Theobalt. 2020. XNect: Real-time Multi-Person 3D Motion Capture with a Single RGB Camera. *ACM Transactions on Graphics* 39, 4, 17. <https://doi.org/10.1145/3386569.3392410>
- Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016a. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499* (2016).
- Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. 2016b. Conditional image generation with PixelCNN decoders. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. 4797–4805.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan



Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems* 32, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 8024–8035. <http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* (2018).

Kaizhi Qian, Yang Zhang, Shiyu Chang, Mark Hasegawa-Johnson, and David Cox. 2020. Unsupervised speech decomposition via triple information bottleneck. In *International Conference on Machine Learning*. PMLR, 7836–7846.

Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. 2019. Autovc: Zero-shot voice style transfer with only autoencoder loss. In *International Conference on Machine Learning*. PMLR, 5210–5219.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. 2018. Faceforensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint arXiv:1803.09179* (2018).

Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. 2019. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1–11.

Sam T Roweis and Lawrence K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science* 290, 5500 (2000), 2323–2326.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised Pre-Training for Speech Recognition. *Proc. Interspeech 2019* (2019), 3465–3469.

Fuhao Shi, Hsiang-Tao Wu, Xin Tong, and Jinxiang Chai. 2014. Automatic acquisition of high-fidelity facial performances using monocular videos. *ACM Transactions on Graphics (TOG)* 33, 6 (2014), 1–13.

Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. 2019. First order motion model for image animation. *Advances in Neural Information Processing Systems* 32 (2019), 7137–7147.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

Pu Sun, Yuezun Li, Honggang Qi, and Siwei Lyu. 2020. LandmarkGAN: Synthesizing Faces from Landmarks. *arXiv preprint arXiv:2011.00269* (2020).

Tiancheng Sun, Jonathan T Barron, Yun-Ta Tsai, Zexiang Xu, Xueming Yu, Graham Fyffe, Christoph Rhemann, Jay Busch, Paul E Debevec, and Ravi Ramamoorthi. 2019. Single image portrait relighting. *ACM Trans. Graph.* 38, 4 (2019), 79–1.

Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman. 2017. Synthesizing obama: learning lip sync from audio. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–13.

Sarah Taylor, Taehwan Kim, Yisong Yue, Moshe Mahler, James Krahe, Anastasio Garcia Rodriguez, Jessica Hodgins, and Iain Matthews. 2017. A deep learning approach for generalized speech animation. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–11.

Justus Thies, Mohamed Elgharib, Ayush Tewari, Christian Theobalt, and Matthias Nießner. 2020. Neural voice puppetry: Audio-driven facial reenactment. In *European Conference on Computer Vision*. Springer, 716–731.

Justus Thies, Michael Zollhöfer, and Matthias Nießner. 2019. Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–12.

J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt. 2015. Real-time Expression Transfer for Facial Reenactment. *ACM Transactions on Graphics (TOG)* 34, 6 (2015).

Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. 2016. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2387–2395.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. 2018. End-to-end speech-driven facial animation with temporal gans. *arXiv preprint arXiv:1805.09313* (2018).

Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. 2019. Realistic speech-driven facial animation with gans. *International Journal of Computer Vision* (2019), 1–16.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018a. Video-to-Video Synthesis. *Advances in Neural Information Processing Systems* 31 (2018), 1144–1156.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018b. High-resolution image synthesis and semantic manipulation

with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 8798–8807.

Olivia Wiles, A Koepke, and Andrew Zisserman. 2018. X2face: A network for controlling face generation using images, audio, and pose codes. In *Proceedings of the European conference on computer vision (ECCV)*. 670–686.

Xinwei Yao, Ohad Fried, Kayvon Fatahalian, and Maneesh Agrawala. 2021. Iterative text-based editing of talking-heads using neural retargeting. *ACM Transactions on Graphics (TOG)* 40, 3 (2021), 1–14.

Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. 2019. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9459–9468.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Hang Zhou, Yu Liu, Ziwei Liu, Ping Luo, and Xiaogang Wang. 2019. Talking face generation by adversarially disentangled audio-visual representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9299–9306.

Yang Zhou, Xintong Han, Eli Shechtman, Jose Echevarria, Evangelos Kalogerakis, and Dingzeyu Li. 2020. MakeltTalk: speaker-aware talking-head animation. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–15.









Yang Zhou, Zhan Xu, Chris Landreth, Evangelos Kalogerakis, Subhansu Maji, and Karan Singh. 2018. Visemenet: Audio-driven animator-centric speech animation. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–10.

Michael Zollhöfer, Justus Thies, Pablo Garrido, Derek Bradley, Thabo Beeler, Patrick Pérez, Marc Stamminger, Matthias Nießner, and Christian Theobalt. 2018. State of the art on monocular 3D face reconstruction, tracking, and applications. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 523–550.

## A APPENDIX

In this appendix, we describe all the sequences we used in detail (Table 4).

Table 4. List of dataset used in our experiments. Video *May* ©UK government (Open Government Licence). Video *Obama1* ©White House (public domain). Video *Obama2* ©Barack Obama Foundation (public domain). Video *Nadella* ©IEEE Computer Society (public domain). Video *Trump* ©White House (public domain). Video *Ford* ©Ontario Office (public domain). Video *McStay* ©Darren McStay (CC BY).

Video	Name	Length
	May	4min 02s
	Obama1	2min 59s
	Obama2	3min 42s
	Nadella	3min 9s
	Actor A	3 min 45s
	Trump	3min 31s
	Ford	3 min 10s
	McStay	4min 30s