DOI: 10.1111/cgf.14640
ACM SIGGRAPH / Eurographics Symposium on Computer Animation 2022
D. L. Michels and S. Pirk
(Guest Editors)
Volume 41 (2022), Number 8

Voice2Face: Audio-driven Facial and Tongue Rig Animations with cVAEs

Monica Villanueva Aylagas, Hector Anadon Leon, Mattias Teye and Konrad Tollmar

SEED - Electronic Arts (EA)

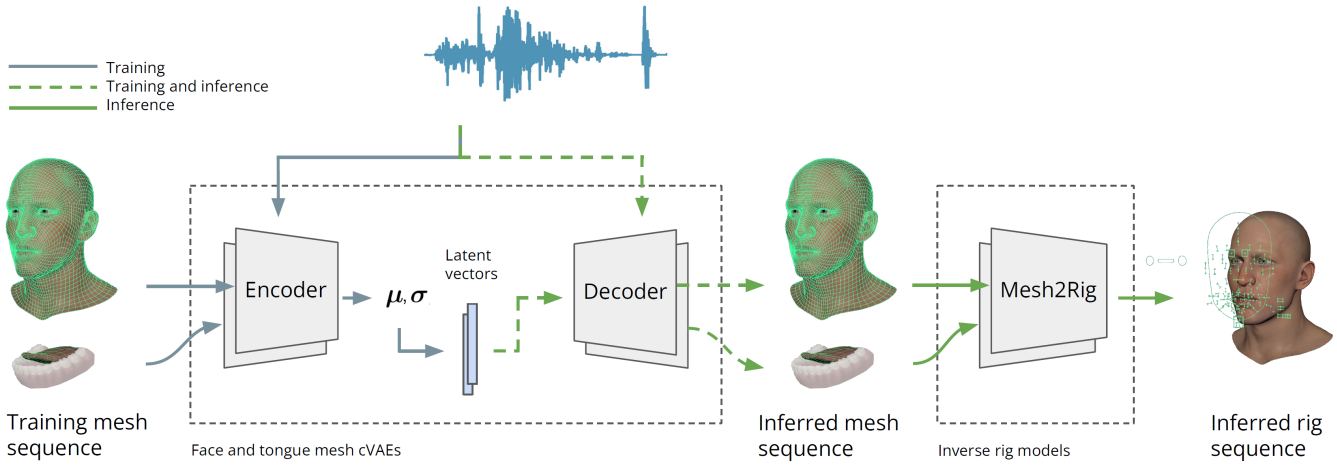

Figure 1: *System overview. During training, two cVAEs are used to encode and generate facial and tongue mesh animations conditioned on speech. During inference, fixed latent vectors are used by the decoders to generate mesh animation sequences, that are then transformed into rig space via models approximating the inverse rig function.*

Abstract

We present Voice2Face: a Deep Learning model that generates face and tongue animations directly from recorded speech. Our approach consists of two steps: a conditional Variational Autoencoder generates mesh animations from speech, while a separate module maps the animations to rig controller space. Our contributions include an automated method for speech style control, a method to train a model with data from multiple quality levels, and a method for animating the tongue. Unlike previous works, our model generates animations without speaker-dependent characteristics while allowing speech style control.

We demonstrate through a user study that Voice2Face significantly outperforms a comparative state-of-the-art model in terms of perceived animation quality, and our quantitative evaluation suggests that Voice2Face yields more accurate lip closure in speech with bilabials through our speech style optimization. Both evaluations also show that our data quality conditioning scheme outperforms both an unconditioned model and a model trained with a smaller high-quality dataset. Finally, the user study shows a preference for animations including tongue. Results from our model can be seen at https://go.ea.com/voice2face.

CCS Concepts

• ***Computing methodologies*** → ***Animation; Neural networks;*** *Latent variable models; Learning latent representations;*

Additional Key Words and Phrases: Deep Learning, Facial animation, Tongue animation, Lip synchronization, Rig animation

1. Introduction

Vision-based performance capture techniques are a crucial part of the facial animation pipeline for modern video games and movies [KAL*17; ZXL*18]. While visually impressive, a downside of such techniques is that they require significant resources in terms of setting, equipment and labor. At the same time, games in particular may now include hundreds of thousands of lines of recorded speech

[New12], increasing the need to generate quality speech animations at scale.

Automated approaches may, in such cases, be a viable complement to performance captures. While performance captures could be used for scenes of higher importance, bulk animations could be generated by less resource intensive options. Such techniques are often based on recorded speech [JAL21] or speech and script [FFX; SG21]. A common approach for such tools is to generate sparse key-framed animations, which lend themselves well for editing by artists but lack the dynamics of performance capture-based animations, as interpolation between poses ignores the face's natural dynamics [Bra99].

Recently, a number of methods have been proposed using Deep Learning (DL) to train speech-driven models directly on performance capture-based data [KAL*17; CBL*19; RZW*21; CWWZ22], generating animations with the qualities attributed to performance captures but requiring only recorded speech as input. While these works excel at e.g. imitating individual speech styles, allowing editability, or generating full face animations, we propose a method that produces generic animations (without speaker-specific characteristics) while providing control over a character's speech style. We argue that these are relevant properties for a bulk animation system: a generic look to make animations applicable to multiple characters, and speech style control to be able to adjust minute details. In this paper we present Voice2Face (V2F), a DL-based method specialized for this task.

While speech-driven facial animation is an important research problem, it also poses several challenges. In the output space, face poses from speech are inherently multi-modal. For a given speaker, any phoneme can be uttered with multiple expressions while different speakers have different speech styles. Similarly, the speech signal also carries both inter and intra personal variations. Any given speaker can repeat a phrase with a wide range of variability stemming from changes such as emotion, spoken volume, and speed, whereas physiological differences yield e.g., timbre differences [BDD*07].

V2F is designed to handle both of these challenges. To decouple the speech-driven and speech-agnostic components of a face pose, we adopt a probabilistic model where the distribution of plausible face poses given a speech window is modeled by a latent vector. Our latent optimization scheme makes it possible to emulate different speech styles in the generated animations. To generalize well to unseen voices, our model is trained on a dataset consisting of 19 speakers in total. We make no language-specific transformations of the speech signal, allowing our model to be used on languages other than those in the training set.

Furthermore, our model can be conditioned on training data quality, allowing the use of lower quality animations in addition to a smaller dataset of high quality. The low quality dataset may be used for improved generalization, since our conditioning allows the model to maintain a high quality.

Our contributions can be summarized as follows:

- We propose using a conditional Variational Autoencoder (cVAE) [KW14; Doe16] to deal with the many-to-many mapping between speech and face poses, encouraging the latent space to decouple the parts of the animations driven by speech and intra- or interpersonal variations.
- Our architecture allows training a model using animations from different quality levels to accommodate time or budget limitations.
- We introduce a method to optimize the latent variable of our model, controlling the overall facial expression of our animations, to a desired style.
- We propose a method to generate tongue animations, and show that animating tongue significantly improves perceived animation quality through a user study.

2. Related work

Speech-driven lip sync has a long history in academia and several approaches have been proposed for the task. Early attempts frequently used Hidden Markov Models (HMMs) to resolve the many-to-many relation between speech and face poses [BCS97; Bra99; VRS03]. In recent years, DL-based models have replaced HMMs as the de-facto standard approach, regardless of output space (2D, 3D or neural rendering). In the following paragraphs, we briefly cover related approaches, focusing on 3D lip sync generation.

Taylor et al. [TKY*17] uses phonemes as an intermediate speech signal to drive coefficients of an Active Appearance Model [CET01], representing the lower face and jaw of a head model. Their phoneme representation yields impressive generalization, but relies on a pretrained module to map speech to a phonetic transcript. A downside of such an approach is that the phoneme representation introduces a certain extent of language dependency. Models that solely take phoneme sequences as inputs will also struggle with gesture magnitudes and lip articulation due to loss of vocal energy and phrasing information [Bra99]. In addition, phoneme representations do not generalize to non-verbal vocalizations.

Cudeiro et al. [CBL*19] uses unnormalized log probabilities of characters from DeepSpeech [HCC*14] as input features in a system that animates 3D mesh vertices given a speaker identity. By conditioning the animations on speaker, their approach is able to separate inter-speaker animation styles and will therefore also generate personalized animations.

With a focus on animator-centric outputs, Visemenet [ZXL*18] predicts sparsely activated viseme- and co-articulation parameters for a FACS-rig from speech, using both phonemes and raw speech features. While providing editor-friendly animations, Visemenet requires pretraining phoneme group and facial landmark prediction components.

Closer to our approach, a number of works produce 3D animations directly from speech [KAL*17; PWP18; TPL*20; RZW*21; CWWZ22]. Using formants as sound representation, Karras et al. [KAL*17] achieves impressive results from less than 4 minutes of training data. Two Convolutional Neural Network (CNN) stacks reduce a sound window's formant and time dimensions respectively, before two fully-connected (FC) layers upscale the signal to mesh vertices. By conditioning the latter convolutional stack on a learnt latent vector per frame of training data, [KAL*17] is able to handle output multimodality. This model is trained individually for each speaker, therefore maintaining personalized speech characteristics.

Pham et al. [PWP18] uses raw spectrogram speech representations from 20 speakers to train a model to predict blendshape and rotation parameters. Their model structure is similar to [KAL*17], but instead of an explicit emotion parameter, they use the hidden state of a Recurrent Neural Network (RNN) layer to capture contextual information implicitly (leading to different facial expressions). A similar approach is taken by Tzirakis et. al. [TPL*20], which uses a Long Short-Term Memory (LSTM) layer to model long term dynamics on a model trained on blendshapes animations temporally aligned to fit speech recordings from the in-the-wild dataset Lip Reading Words [CZ16].

More recently, Chai et al. [CWWZ22] gathers information along the frequency dimension of a speech window with a stack of convolutions, but uses self-attention layers to collect information along the time dimension. Similar to Cudeiro et al. [CBL*19], their model takes speaker identity as auxiliary input and is thus able to explicitly model different speaking styles.

Richard et al. [RZW*21] trains two separate encoders to learn lip sync from audio and facial expressions respectively, fusing the results in a categorical latent vector that then animates a template mesh. While the fidelity of the generated animations is impressive, they train a separate model to generate audio-conditioned categorical latent vectors since an expression sequence is not available at inference time, thereby foregoing explicit control of speech style.

Our approach is designed to produce lip sync suitable for bulk animations for a wide range of characters, as a lighter weight option to performance captures. Unlike [CBL*19; KAL*17; CWWZ22], we aim to produce generic animations, without speaker-dependent characteristics. In contrast to e.g. [PWP18; TPL*20; RZW*21; CWWZ22], we maintain control of the generated animations by explicitly modeling speech style as a latent variable, optimized to yield a desired look before inference. In addition, by mapping face poses in mesh space directly to speech features from the sound signal, we avoid any information loss associated with discretizing to phonemic or visemic representations [Bra99].

Our method also produces tongue animation, an important factor for perceived realism, as some phonemes are not distinguishable with just the lip shape [PvOS94]. While tongue animation is a smaller research area than lip sync, several methods have previously been proposed. In terms of speech-driven animations, Luo et al. [LYLZ17] controls the deformation of the tongue using HMMs. Other methods [FHG*17; EB03] propose to model tongue movement by using ultrasound data or 3D face landmarks.

Finally, our facial and tongue animations are translated from mesh space to rig space. A rig function [LA10] maps an artist-crafted set of controllers (rig information/parameters) to mesh vertex positions. The rig representation is used by professional animators to create the desired movements using a high-level interface with semantic meaning [HSK16; OBP*12], thereby making editing animations easier. It also constrains the deformations to plausible shapes and standardizes animations, even when produced by multiple animators [LAR*14]. Using rig animations also carries other benefits, like compression and reusability of the same animation on different rigged heads. Our objective is to estimate the inverse of the rig function, for mapping mesh to rig parameters. Traditional approaches can produce this inverse mapping, i.e. least squares optimization, in a highly inefficient way by calling the rig function repeatedly [HMT*12; HTC*13]. In our case, a data-driven approach is desired instead as it is independent from the implementation of the rig function and can be retrained for different functions. Several methods have been proposed e.g. using Neural Networks (NN) or Gaussian Processes [HSK16]. In addition, Bailey et al. [BODO20] proposes an inverse kinematics method to drive rig parameters using landmarks by predicting deformation maps in the UV space. Several NNs are used to estimate multi-resolution deformations for different vertex subsets. Our approach uses a feed forward NN similar to Holden et al. [HSK16].

Quantitative evaluation of speech-driven facial animations is an open problem in the literature. Some works [KAL*17; CBL*19] omit quantitative evaluations due to the many-to-many mapping between speech utterances and visemes, relying only on perceptual evaluations via user studies. Other works [RZW*21; ZXL*18; RLM*21] design custom quantitative metrics that are tailored to their use cases. An example is the maximal $\ell 2$ error of all lip vertices averaged over all frames in the test set to measure lip synchronization [RZW*21]. We argue that the test set ground truth is just one realization of the multiple plausible face poses associated with a given speech signal, making this metric inadequate to capture the variability of speech styles and idiosyncrasies. Zhou et al. [ZXL*18] proposes two approaches. The *motion curve differences*, defined as the absolute difference of the rig parameter values compared with the ground truth, is similar to the maximal $\ell 2$ error, thus suffering from the same problem. On the other hand, computing precision and recall on the rig parameter binary activations only informs about the correctness of activating the parameter, but not the accuracy of the parameter value. Richards et al. [RLM*21] report F1-scores based on achieved lip closure for corresponding sections from test data. We use the same metric, but only measure achieved lip closure for annotated occurrences of the bilabial consonants (/p b m/) as they correspond to face poses requiring closed lips regardless of the overall expression.

3. Method

The goal of this work is to present a method that models a speaker-independent, multi-modal distribution of face poses in rig space given a speech signal over time. We propose a two-step procedure: two cVAEs first generate speech-conditioned face and tongue mesh poses respectively, and two smaller Mesh2Rig (M2R) modules then convert the mesh poses to rig parameters. An overview of the system is shown in Fig. 1. In this section, we will introduce both steps sequentially.

To train the cVAEs, we assume that we are given a mesh animation dataset $A = \mathbf{a}_{1:T}$ of total length T animation frames, with corresponding speech $S = S_{1:T}$. Here, $\mathbf{a}_t \in \mathbb{R}^{3V}$ are mesh vertex coordinates where V is the number of vertices, and $S_t \in \mathbb{R}^{F \times B}$ is a speech window with B bins of F speech features centered at frame t. Additionally, a categorical indicator $\mathbf{q}_t$ of animation quality may be used to condition mesh pose generation on quality. For M2R training, we assume a dataset of mesh animations A, with corresponding rig animations $R = \mathbf{r}_{1:T}$, where $\mathbf{r}_t \in \mathbb{R}^P$ with P rig parameter attributes. Since we treat face and tongue independently,

mesh and rig data correspond to either face or tongue poses, but we omit such notation for readability.

3.1. Conditional Variational Autoencoders

Intuitively, the first step of our two-step procedure consists of a mapping f from some contextual speech $\tilde{S}_t$ to the mesh pose $\mathbf{a}_t$ at time t, i.e. $\hat{\mathbf{a}}_t = f(\tilde{S}_t)$. In our work, $\tilde{S}_t = S_{t-k:t}$ is a sequence of $k+1$ speech windows ending with S_t.

Simply regressing the pose on the speech signal however would fail to capture the multi-modal nature of face and tongue poses associated with speech and default to the average position [KAL*17]. We therefore propose a probabilistic approach using conditional Variational Autoencoders (cVAEs) [SLY15]. cVAEs assume that there exists a latent variable $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, I)$ that encodes speech-agnostic variations in facial poses, such as individual speech styles. Thus, we model f as a probability distribution p_θ, that infers the facial pose at time t as follows:

$$\hat{\mathbf{a}}_t \sim p_\theta(\mathbf{z}_t, \tilde{S}_t, \mathbf{q}_t). \tag{1}$$

We assume p_θ to be independent and identically distributed (iid) Gaussian variables with a fixed variance scalar.

In order to infer the posterior distribution over latent variables given a specific observation, cVAEs make use of variational inference. To this end, cVAEs assume an approximate posterior distribution $q_\phi(\mathbf{z}_t|\mathbf{a}_t, \tilde{S}_t)$. The parameters of the approximate posterior q_ϕ and the likelihood p_θ are modelled by deterministic NNs, denoted as the encoder E and the decoder D respectively. Assuming iid Gaussian latent variables, the latent code at time t during training is thus sampled as:

$$\mu_{\mathbf{a}_t|\tilde{S}_t},\ \sigma_{\mathbf{a}_t|\tilde{S}_t} = E(\mathbf{a}_t, \tilde{S}_t) \tag{2}$$

$$\mathbf{z}_t \sim \mathcal{N}(\mu_{\mathbf{a}_t|\tilde{S}_t}, \sigma_{\mathbf{a}_t|\tilde{S}_t} I). \tag{3}$$

In summary, by conditioning E and D on contextual speech information $\tilde{S}_t$ associated with each $\mathbf{a}_t$ during training, the latent distribution learns to represent the space of plausible face poses given speech. A one-hot-encoded variable $\mathbf{q}_t$ is used to disentangle animation quality when training on datasets that contain animations with varying quality, improving generalization without degrading the quality of the generated animations.

Architecturally, the cVAEs we train to generate face and tongue animations respectively are identical. An overview of the cVAE structure is shown in Fig. 2. A more detailed description of all the layers and parameters in the architecture can be found in the supplementary material.

3.1.1. cVAE training

We train our cVAE models to minimize the standard VAE objective function [KW14]. The first loss term corresponds to the log likelihood (LL) of the ground truth mesh coordinates given the estimated likelihood distribution, while the second term represents the loss of the latent distribution using the Kullback–Leibler (KL) divergence,

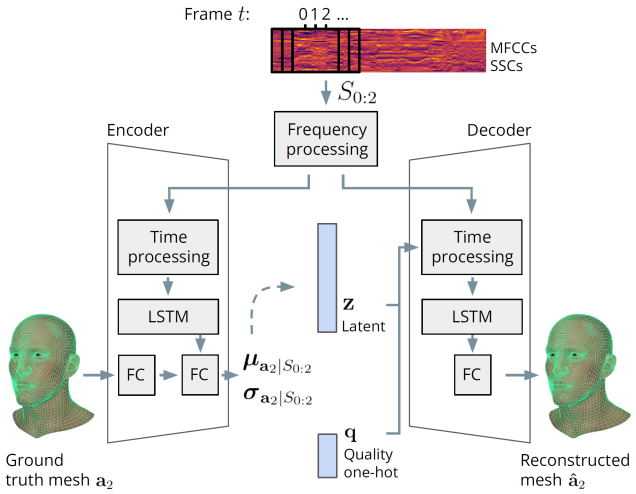

Figure 2: *Schematic of the cVAE used to generate mesh frames for face and tongue. We show an example of reconstructing the face mesh frame* $\mathbf{a}_2$ *during training. Three speech windows are processed over frequency and time before being aggregated by a Long Short-Term Memory (LSTM) layer. The encoder conditions the ground truth mesh with speech to generate latent distribution statistics, which are used for sampling by the decoder to reconstruct the mesh.*

as follows:

$$\mathcal{L}_{LL} = \mathbb{E}_{q_\phi(\mathbf{z}_t|\mathbf{a}_t, \tilde{S}_t)}[\log p_\theta(\mathbf{a}_t|\mathbf{z}_t, \tilde{S}_t, \mathbf{q}_t)] \tag{4}$$

$$\mathcal{L}_{KL} = -\mathrm{KL}(q_\theta(\mathbf{z}_t|\mathbf{a}_t, \tilde{S}_t)||\mathcal{N}(\mathbf{0}, I)). \tag{5}$$

We also include the vertex normal cosine distance as an additional reconstruction loss term to exploit 3D neighboring information, similar to [VAPE20; BODO20].

$$\mathcal{L}_N = 1 - \frac{\sum_{v=0}^{V} \cos(n_v, \hat{n}_v)}{V}, \tag{6}$$

where n_v is the normalized sum of all normal vectors of faces that share vertex v, and $\hat{n}_v$ is the corresponding prediction. The intuition behind this loss is that it can help with rotations, e.g., for lip roll.

The final objective of the mesh generation cVAE models is constructed as the sum of the terms above.

$$\mathcal{L} = \mathcal{L}_{LL} + \mathcal{L}_{KL} + \mathcal{L}_N. \tag{7}$$

During training, we enforce a temporal latent stability strategy to encourage the model to produce smooth transitions between frames. We encourage E to produce codes that are valid for more than one frame. Specifically, a single latent vector $\mathbf{z}_t$ sampled from the distribution of the encoded frame $\mathbf{a_t}$ is used to reconstruct three consecutive decoded frames: $\hat{\mathbf{a}}_{t:t+2}$ (Fig. 3). We adopt this training scheme since we use a single latent optimized to yield a neutral facial expression during inference. However, it also prevents the network from learning sparse, short-term variations like eye blinking,

since they cannot be captured in the latent and are not correlated with speech.

3.1.2. cVAE inference

After training, we use the decoder D to generate animations for new speech. We left pad the speech signal with enough silence to make $\hat{\mathbf{a}}_0$ coincide with the start of the speech recording (i.e. $\tilde{S}_0 = S_{-k:0}$). If the cVAE was trained with datasets of varying quality, $\mathbf{q}_t$ is set to represent the highest quality level used during training.

In order to guarantee a desired facial expression, the latent vector $\mathbf{z}_t$ is maintained constant throughout inference (i.e $\mathbf{z}_t = \mathbf{z}$), and set to represent a desired speech style through our latent optimization scheme. For the cVAE corresponding to tongue poses, we set $\mathbf{z}$ equal to the zero vector as we don't see a significant reason to fine-tune tongue poses.

For the cVAE corresponding to facial poses, we optimize $\mathbf{z}$ to force the output animation to resemble the average speech style of a collection of short validation clips. This approach is inspired by the gradient-based reconstruction [LT17] which has become popular in association with powerful pretrained networks such as StyleGAN [KLA19; KLA*20; KAL*21], enabling editing applications [AQW20; LZG*21].

The optimization works as follows. First, short validation clips (ca. 20 frames) of a desired speech style are selected, chosen in particular to include bilabial sounds and silences. Samples with speech from different speakers help with generalization. Freezing the weights in D, we update $\mathbf{z}$ by gradient descent, minimizing the mean-squared error (MSE) between the ground truth animation $\mathbf{a}_t$ and the output of the model $\hat{\mathbf{a}}_t$.

This latent optimization is only performed once, with the resulting $\mathbf{z}$ being saved for use in future inference.

Unless specifically mentioned, throughout this work we optimize the latent vector on a collection of validation clips with neutral facial expressions.

3.1.3. cVAE architecture

We follow the sound processing described by Karras et. al. [KAL*17], reducing the frequency and time dimensions of the speech feature windows in $\tilde{S}_t$ respectively through two stacks of 2D convolutions. The first stack compresses the windows in the frequency domain to detect features that correlate with face poses, such as phonemes. Weights in this stack are shared between E and D. The second stack compresses the windows in the time domain, distilling temporal information to recognize co-articulation patterns. This implementation differs in E and D due to the role of $\mathbf{z}_t$ and $\mathbf{q}_t$. In D, $\mathbf{z}_t$ and $\mathbf{q}_t$ are concatenated, replicated along the time dimension, and appended to the channels of the previous layer's output, equivalently to [KAL*17]. This allows conditioning information to influence multiple levels of detail in the animation. In E, no conditioning is used to encourage the network to find a generic representation that learns an expressive latent space.

Temporal information from the sequence of processed speech is aggregated by a single Long Short-Term Memory (LSTM) layer. The purpose of this sequential aggregation is to produce temporally stable animations. During development, we noticed a trade-off between short sequences leading to responsive but jittery results, and long sequences resulting in temporally stable but slow animations. We found that $k = 2$ at 30 frames per second gave satisfactory results, and theorize that three speech windows allow enough context for our model to handle long term co-articulation, e.g. determining if a silence is long enough to close the mouth in a resting pose.

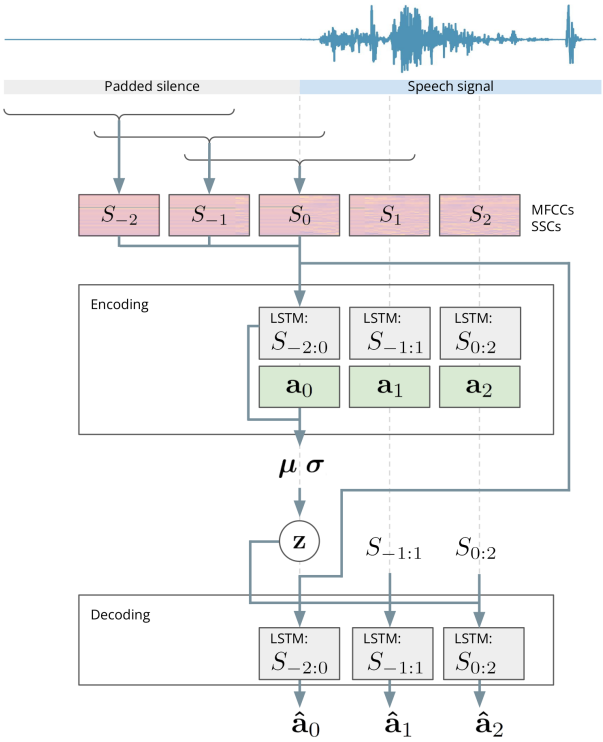

Figure 3: *Overview of data matching during the encoding and decoding phases. A context of three speech representation windows S_{t-2}, S_{t-1}, S_t is used encode frame $\mathbf{a_t}$. The same sampled latent $\mathbf{z}_t$ is used to decode the frames $\hat{\mathbf{a}}_\mathbf{t}, \hat{\mathbf{a}}_{\mathbf{t+1}}, \hat{\mathbf{a}}_{\mathbf{t+2}}$.*

In the encoder, the low-dimensional representation of a face pose decoupled from speech is achieved by passing $\mathbf{a}_t$ through a FC layer, concatenating the LSTM's output, and processing through subsequent FC layers. These layers act as a bottleneck that reduces the dimensionality according to the desired size of the latent variable $\mathbf{z}_t$. The output of the last layer represents the statistics of the multivariate isotropic Gaussian distribution that models the latent space following Equation 3.

In the decoder, we map the LSTM's output to the generated animation frame $\hat{\mathbf{a}}_\mathbf{t}$ using an FC layer stack, following Equation 1.

3.2. Mesh2Rig

The second step of our procedure consists of mapping the mesh animations to a rig parameter space. Formally, M2R can be described as a function g approximating the inverse rig function, i.e. $\mathbf{r}_t = g(\mathbf{a}_t)$. To this aim, we train two separate M2R modules

to perform the conversion for face and tongue respectively. As a post-processing step, resulting rig animations from both models are joined together.

The face M2R is similar to the NN solution in [HSK16], and consists of a multilayer perceptron (MLP) with two hidden layers. The model is trained to minimize the MSE loss between predicted and ground truth rig parameter attributes, taking mesh pose as input. We found that applying Additive Gaussian Noise (AGN) [Bis95] to the input layer during training resulted in improved performance when the M2R model is used on inputs which are not perfectly representable by the rig, such as the outputs from the cVAE. Details of the architecture can be found in the supplementary material.

The model yields the best results when trained on data from speech animations, so that only the subset of poses resembling speech are learned by the model. In case of a lack of native speech animations, it is still possible to train a M2R model by generating data sampled randomly from the rig. We found however that adding such randomly sampled data to our existing dataset did not yield superior results.

For the tongue M2R, we used a least square solver of the tongue vertices as a 3rd degree polynomial. We found this solution to be sufficient for the tongue, and speculate that this is due to tongue dynamics being simpler as they have fewer degrees of freedom and sparse rig activations. We found a need to clip the rig values to prevent them from reaching extreme poses. The tongue M2R model is trained on the same speech animation frames as the face M2R model.

3.3. Implementation details

Both animations and speech need to be in a suitable format before being processed by the system. The mesh vertex coordinates $\mathbf{a}_t$ correspond to the offset of the current animation frame to the head's neutral pose. Each speech window S_t is composed of 64 bins B of 13 MFCCs and 26 SSCs for a total of 39 normalized speech features F. We take 8 ms steps between bins and use a sliding window of length 25 ms for each bin.

The NNs are implemented in PyTorch [PGM*19] and are trained with Adam [KB15] as our optimization strategy. Training the facial mesh generation network takes 44 hours, while facial M2R requires 50 minutes. Producing rig animations from pre-recorded speech takes 1.68 ms per frame. All measurements are performed on a single GeForce GTX 1080 Ti.

4. Results

In this section we describe the evaluation methodology adopted to assess the proposed system. In order to validate our contributions we compare our proposed model (*Ours*) with ablated versions, and VOCA [CBL*19], a comparative state-of-the-art model for facial lip sync:

- Non-Generative Model (NGM): This model is trained without generative properties, i.e. the decoder does not take the latent vector as input and is trained without the KL divergence loss (equation 5).

Table 1: *Overview of the datasets used for training, validation and testing. The number of participants and minutes of data is split between males and females. Note that the test set is larger than the training set since it contains only audio, thus easier to source.*

Dataset	Subjects (M / F)	Minutes (M / F)	Rig	Mesh	Tongue	Audio
Train - *gold*	7 / 5	26.4 / 18.8	✓	✓	✓	✓
Train - *silver*	2 / 5	12.1 / 71.1	✗	✓	✗	✓
Val - *gold*	1 / 1	0.1 / 0.1	✓	✓	✓	✓
Val - audio-only	10 / 7	64.1 / 56.2	✗	✗	✗	✓
Total Train	9 / 10	38.6 / 89.9	N/A	N/A	N/A	N/A
Total Val	11 / 8	64.2 / 56.4	N/A	N/A	N/A	N/A
Test	13 / 10	117.9 / 83.3	✗	✗	✗	✓

- No Normal Loss (NNL): This model is trained using only the core cVAE framework without the normal loss (equation 6).
- No Quality Conditioning (NQC): This model does not make use of the data quality conditioning described in Sec. 3, treating all data equally.
- No Silver Data (NSD): This model is trained exclusively with the *gold* dataset, the smaller subset of high quality data in Sec. 4.1.
- No Latent Optimization (NLO): This ablation does not change the training procedure but the inference. Instead of performing latent optimization as described in Sec. 3.1.2, the latent employed for inference is the zero vector.
- VOCA: We use the official codebase, and train the model using our training data.

4.1. Dataset

The training dataset is composed of pairs of animation and synchronized audio. Animations are extracted at 30 frames per second and animate a target head with 7071 vertices and 78 rig parameter controller attributes. All speech is recorded in English with a wide variety of accents.

Our training set contains 128 minutes of animation from 19 participants solved on the same target head. Participants were recorded reading phonetic pangram scripts, which cover all phonetic sounds in English. All participants were instructed to maintain a neutral expression while reading the scripts. The animations were obtained from two quality sources. The *gold* dataset consists of high quality performance captures, while the *silver* dataset is recorded with a wide variety of devices and solved to mesh animation via an in-house video-based solution, resulting in lower quality animations.

The test set is larger than the training set since it contains a large compendium of voices without ground truth animation. It consists of 23 subjects, amounting to 201 min. This data consists of non-pangram scripts and includes non-verbal sounds and a wider emotional range. All of our quantitative experiments are performed on the test set.

Finally, the validation set includes a small subset of hold-out data from the *gold* training set and the audio-only set. It contains 19 subjects and 120 minutes. For a more comprehensive description of the contents of each subset of data, see Table 1.

Table 2: *Quantitative metrics for the ablated models and VOCA. We report mean and standard deviation over 5 runs, aggregating also over 12 gold IDs for VOCA.*

Model	Precision %	Recall %	F1 %
NGM	72.61 ± 4.70	24.39 ± 6.00	36.25 ± 7.00
NNL	65.93 ± 5.52	55.17 ± 7.10	**59.59 ± 2.61**
NQC	79.30 ± 19.62	12.69 ± 14.21	18.76 ± 19.66
NSD	56.33 ± 6.71	42.07 ± 12.12	47.65 ± 9.67
NLO	77.96 ± 6.74	25.24 ± 3.36	37.94 ± 3.51
Ours	62.01 ± 4.54	**55.86 ± 14.43**	57.55 ± 7.13
VOCA	**86.87 ± 6.93**	30.06 ± 13.46	42.79 ± 16.14

4.2. Quantitative evaluation

In this section we quantitatively evaluate the lip sync performance of the facial mesh generation network, and the ability to translate from mesh to rig space by the facial Mesh2Rig module.

4.2.1. Facial mesh generator

Bilabial (/p b m/) lip closure is one of the most important features that correlate with perceptual quality [RLM*21]. We compute lip closure by measuring the distance in mesh space between the upper and lower lip in the center of the mouth for all sound frames, classifying the observations as either open or closed. These instances are compared to annotated test speech recordings to compute precision, recall and F1-score. For these measurements, we define closed lips as a positive instance.

Each model was trained 5 times for 200 epochs, with checkpoints taken every 10 epochs. For each training run, the best epoch was selected by visual inspection of animations generated from a 1.5 min validation clip. For the VOCA models' epoch selection, a consistent speaker ID sampled from *gold* data was used for all inferences due to VOCA's speech style dependence on speaker. The metrics were calculated for the selected checkpoints and aggregated per model, as well as ID for VOCA, to report mean and standard deviation in Table 2.

It is important to note that these metrics only measure bilabial lip closure and do not correlate exactly with human perception, which can attend to other aspects of the animation such as jitter or the expression during silent sections.

4.2.2. Facial Mesh2Rig

We evaluated the effect of applying AGN to the inputs during M2R training. The evaluated models differed in the standard deviation of the noise added, $\sigma \in [0, 0.1, 0.2, 0.3, 0.4, 0.5]$. We trained and evaluated models following a 10-fold cross-validation scheme with the *gold* dataset in Table 1. To more closely simulate the intended use case of M2R, we also evaluated the same models on a 2.5 min validation animation, generated by a trained cVAE. In both cases, we evaluated MSE in the mesh space, meaning that the rig animations predicted by the M2R were converted back to a mesh representation using the rig function, and compared to the mesh used as inputs. Resulting MSE mean and standard deviation are shown in Table 3.

When evaluated on held-out animations from our *gold* dataset, the lowest MSE is observed with $\sigma = 0$. In contrast, evaluating the same models on cVAE generations yields the lowest errors for $\sigma = 0.3$.

Table 3: *MSE for mesh to rig models, varying AGN σ applied to the inputs during training. Metrics are gathered from 10 models per σ trained for 100 epochs. Hold-out speech shows errors from 10-fold cross-validation. cVAE generations shows errors from the same models tested on an animation sequence generated by the cVAE. The reported errors are in mesh space. Numbers are expressed in micrometre (μm) for improved readability.*

	Test data type			
	A. Hold-out speech		B. cVAE generations	
Model σ	Mean	SD	Mean	SD
0.0	**1.12**	**0.02**	7.41	0.52
0.1	1.24	0.02	4.62	0.34
0.2	1.43	0.02	3.89	0.35
0.3	1.64	0.03	**3.60**	**0.17**
0.4	1.94	0.09	3.84	0.17
0.5	2.14	0.03	3.93	0.12

4.3. Qualitative evaluation

Quantitative facial animation evaluation remains an unsolved problem. Therefore, we complement our metrics with qualitative user studies.

The blind user studies were carried out internally in the form of A/B comparisons, allowing ties, with videos rendered from rig animations. Participants were presented with a series of videos in which two side-by-side heads animated the same speech, each side generated by the proposed model or the ablations. The question asked in all experiments was: *Which one of the videos looks more natural to you?* The participants, with different levels of experience with facial animation, were provided with instructions and a tutorial showing three examples of videos they could encounter: a standard comparison and two types of attention checks. These attention checks showed either the same animation in both heads or one of the animations out of sync with the sound. Participants who failed any attention check were filtered out as unreliable.

To create the animations, we sampled two audio files per speaker, ensuring no overlap with lines used for validation. The length of the files ranged between 4-6 seconds, with at least 50% of non-silent frames. The models used for the user study were selected by visual comparison of animations generated from a 1.5 min validation clip among the best checkpoints per model. The lines shown to the participants and the ordering of the animations in the video were randomized. For the VOCA comparisons, so were the IDs selected for inference (sampling from the *gold* dataset). We allowed the users to play the videos as many times as necessary and at different speeds to permit a more detailed comparison.

We divided the evaluations into two user studies. The first one was designed to compare the ablation models, validate our contributions through human preference, and correlate the results to

our quantitative metrics. For this study, we gathered 860 valid responses from 43 participants (16 animators, 27 non-animators). In the second study, we evaluated whether the inclusion of tongue animation increases the overall perceptual quality of the animation and compared our model to VOCA [CBL*19], trained on our dataset. For this survey, we received 230 and 460 valid responses respectively from 46 participants (16 animators, 30 non-animators). In Fig. 4, we report the analysis of the user studies using a binomial sign test excluding ties following [JKHB20]. In addition, Fig. 5, shows the results of the ablation in terms of aggregated preference percentage.

5. Discussion

The purpose of the ablation study was to validate the contributions to our face generator and select the best model to compare against VOCA. The results show that using the quality encoding is the feature that improves the quality the most, with a statistically significant preference for those models over NQC ($p < 0.001$). The analysis also suggests that using a larger amount of data, conditioned on the quality is perceptually better than using a smaller subset of high quality data (NSD), and statistically preferred if the latent is optimized (*Ours* and NNL). These two models yield very similar visual results, which translates to votes being almost uniformly distributed between the three options. Our motivation to include the vertex normal as part of the loss function was improve lip rolls for sounds such as /f v/. The sparsity and sample dependence of these sounds in the study, together with the difficulty to recognize these differences, can account for indicative but not significant preference for our proposed approach.

Comparing the results of the quantitative metrics (Table 2) with the preference of the participants in the perceptual study (Fig. 4-5), we can observe partial correspondence. The highest F1 scores belong to the NNL and our proposed model with overlapping statistics, which are also the models with the largest user preference shares on aggregate. This is also the case for recall, which may suggest that a model's ability not to miss lip closure during bilabial speech is important for perceived animation quality. While not evaluated qualitatively, we observe that NGM's low recall suggests mouth closure issues for bilabials, likely caused by the multimodal output space. Recall, F1 and user preferences are also consistent with NSD outperforming NQC.

We also note, however, that NLO, which ranks low in our quantitative evaluation, performs better than expected according to participants, almost as high as NNL and *Ours*. One explanation could be that the animation can give the illusion of lip closure when playing in real time, if lips are almost closing during bilabials. This idea is reinforced by the fact that only 12 participants (27.9%) used the slow motion functionality during the study. Another explanation is simply that our metrics are not able to capture the full picture of perceptual quality, and we cannot expect full correspondence in every instance.

As the winner of the ablation comparison, our proposed model is used in the second survey. This study was designed to assess two important contributions: the perceptual importance of our tongue animations and a comparison with VOCA, a state of the art method for audio-driven animations.

When comparing against VOCA, our proposed model is preferred with statistical significance ($p < 0.05$). Interestingly, animators have a stronger preference for our model than non-animators ($p = 0.002$), with 58% voting for *Ours*, 36% for VOCA and 6% showing no preference.

Similar to [RZW*21], we also identify in our experiments that the quality of VOCA is dependent on the identity used for inference. Analyzing the results by ID, we find that our approach is preferred for 10 out of the 12 high quality identities.

The comparison between animations with and without tongue shows a high percentage of ties. According to the feedback received, this is both caused by a lack of preference or the inability to see the movement due to the nature of the sentence. After removing the ties, there is a statistically significant preference for the animations that include tongue ($p < 0.001$), which suggests that using the tongue animation does not damage the overall quality of the animation.

One aspect we noticed while evaluating our results is the amount of variance in a model's properties between training runs. This is true for *Ours* and its variants, as well as VOCA. We observe similar variances as those reported in Table 2 even when comparing training runs on a common epoch. This variance is also observable in the generated animations, which can vary significantly between two training runs differing only on the random seed. We are transparent about this variance by reporting mean and std. dev. in the quantitative metrics over 5 training runs, and choosing the best variant out of the 5 runs for each model for the qualitative evaluation. We would like to highlight this observation as we notice it is not only linked to our model and, therefore, want to encourage the community to take and report similar precautions.

5.1. Effect of different latent vectors

In this work, our main objective is to generate facial animations with a generic, neutral delivery. Subjects captured with both quality methods were instructed to maintain a neutral facial expression throughout their performances. In all evaluations, we optimized the latent to yield a neutral facial expression (Sec. 3.1.2).

Naturally, some variance in the expressions adopted by the participants can still be found in our training data. We found that we are able to emulate different expressions by optimizing the latent on selected subsets, as well as individual speaking styles if the optimization clips belong to a single speaker. The poses in Fig. 6 are snapshots of animations generated from a single V2F model, taken at the same point in time but differing in the latent vector used during inference. More examples of speech generated from a single V2F model where the latent vector is optimized to match different speech styles are shown in our accompanying video (https://go.ea.com/voice2face).

5.2. Facial Mesh2Rig

Visual inspection on animation sequences generated from previously unseen speech confirmed that the model trained with $\sigma = 0.3$, while worse at mapping mesh animations perfectly representable by the rig, performs better at predicting rig parameters that match

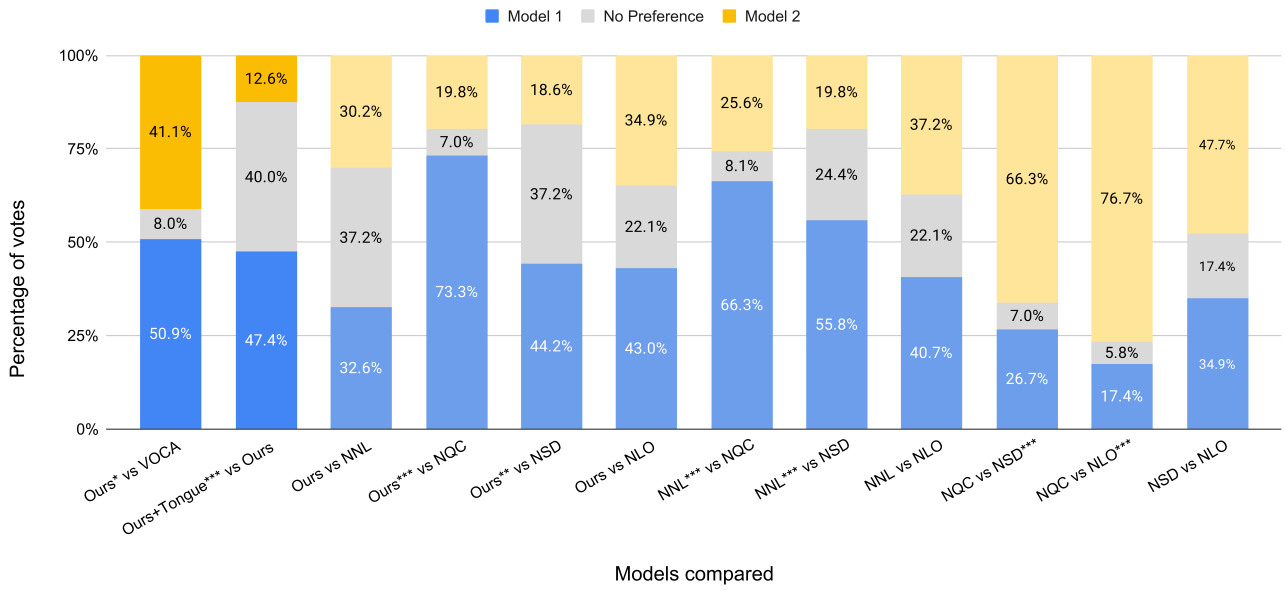

Figure 4: *Distribution of votes for all comparisons. The asterisks accompanying the name indicate different values of statistical significance (* = $p < 0.05$; ** = $p < 0.01$, *** = $p < 0.001$).*

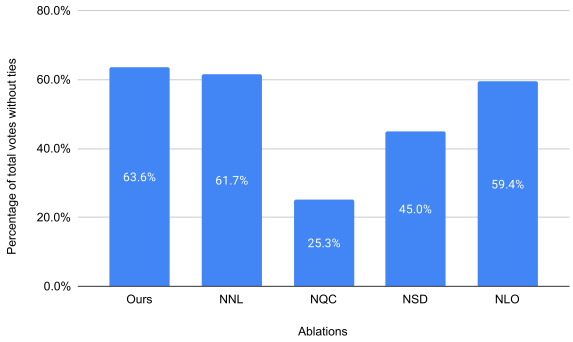

Figure 5: *Percentage of model preference after tie exclusion. All ablation comparisons in Survey 1 are accounted for.*

Figure 6: *By varying the latent variable used for inference we can generate animations with different speech styles. These snapshots come from the same point in time in animations inferred by a single V2F model, using a different latent vector.*

the mesh poses generated from our cVAE. We believe that the noise introduced during training helps the M2R model handle input mesh vertices outside of the distribution of the rig representation, making it more suitable for handling the frames generated by our cVAE.

The conversion of the cVAE's mesh outputs to rig animations will always be subjected to rig limitations in terms of quality, regardless of the source of the mesh generator's training data. Note that this conversion is optional and depends on the application. If the use case does not require the animations to be in rig space, training the mesh generation network on 4D captured animation data would likely yield higher quality results, as long as they are retargeted to the same head.

6. Conclusions and future work

We have presented Voice2Face, a Deep Learning model that generates face and tongue animations directly from audio input. Our approach seamlessly integrates multiple modules: a cVAE that generates mesh animation and a module that translates the mesh animations to rig controller space. We show that by using quality conditioning, we can train on datasets with animations from multiple quality levels without degrading animation performance. In addition, we introduce a method to optimize the latent variable of our model, controlling the overall speech style of our animations. We also design a novel method to quantify lip sync quality by measuring lip closure for bilabial sounds. We perform a user study where we observe partial correspondence between perceived quality and the quantitative metric, a significant perceived quality improvement when including tongue animation and a significant preference over a state-of-the-art method, particularly among animators. Our ap-

proach demonstrates accurate lip sync while showing natural motions on the lip and jaw region. Unlike previous works, our model generates identity independent animations while allowing speech style control. We believe these properties make the model particularly useful as a bulk animation tool for the entertainment industry and, with improved performance, for cinematics.

Currently, the latent vector allows limited control of expressions due to the reduced emotional range of the dataset. Emotional control can enable artists to edit and manipulate the generated animations. However, animators would benefit from a semantic layer on top of the latent.

Our evaluation results suggest that there is a partial correspondence between our metrics and human perception, yet current metrics do not capture the full extent of animation quality. New metrics should be considered in order to reduce the evaluation time of new trained models and to establish a benchmark to compare facial animation quality. Future work in this area may include designing metrics that reflect other perceptual qualities like jittering or pose during silences as well as the construction of a meta-metric that correlates with human preference, similar to [LAK*16].

The results of the user study show that our method outperforms state-of-the-art animation quality. Furthermore, it proves the viability of our tongue animation generation technique. We acknowledge, however, that this is a first approach that requires further investigation. Our method could also benefit from including more features that would increase the realism of the animation such as head and saccadic eye movements or gaze behavioral patterns related to emotions (e.g. gazing up when thinking).

The main focus of the proposed method is to produce off-line bulk animations and not real-time applications. As such, optimization was not among the objectives of this work. Apart from reducing computational costs, our model is subjected to an inherent latency of 0.26 s since the last sound window in the input sequence $S_{t-k:t}$ is centered on the target animation frame. More research is needed in the field to bypass this issue. Finally, the usability of our method in a production pipeline is outside of the scope of this work and something that will be explored in the future.

7. Acknowledgements

The authors would like to thank Kristoffer Sjöö for his work in developing the model, Timur Solovev for all the infrastructure support, and our colleagues at EA for participating in the user study.

References

[AQW20] ABDAL, RAMEEN, QIN, YIPENG, and WONKA, PETER. "Image2stylegan++: How to edit the embedded images?": *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, 8296–8305 5.

[BCS97] BREGLER, CHRISTOPH, COVELL, MICHELE, and SLANEY, MALCOLM. "Video rewrite: Driving visual speech with audio". *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 1997, 353–360 2.

[BDD*07] BENZEGHIBA, MOHAMED, DE MORI, RENATO, DEROO, OLIVIER, et al. "Automatic speech recognition and speech variability: A review". *Speech communication* 49.10-11 (2007), 763–786 2.

[Bis95] BISHOP, CHRIS M. "Training with noise is equivalent to Tikhonov regularization". *Neural computation* 7.1 (1995), 108–116 6.

[BODO20] BAILEY, STEPHEN W, OMENS, DALTON, DILORENZO, PAUL, and O'BRIEN, JAMES F. "Fast and deep facial deformations". *ACM Transactions on Graphics (TOG)* 39.4 (2020), 94–1 3, 4.

[Bra99] BRAND, MATTHEW. "Voice puppetry". *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. 1999, 21–28 2, 3.

[CBL*19] CUDEIRO, DANIEL, BOLKART, TIMO, LAIDLAW, CASSIDY, et al. "Capture, Learning, and Synthesis of 3D Speaking Styles". *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2019, 10101–10111 2, 3, 6, 8.

[CET01] COOTES, TIMOTHY F., EDWARDS, GARETH J., and TAYLOR, CHRISTOPHER J. "Active appearance models". *IEEE Transactions on pattern analysis and machine intelligence* 23.6 (2001), 681–685 2.

[CWWZ22] CHAI, YUJIN, WENG, YANLIN, WANG, LVDI, and ZHOU, KUN. "Speech-driven facial animation with spectral gathering and temporal attention". *Frontiers of Computer Science* 16.3 (2022), 1–10 2, 3.

[CZ16] CHUNG, J. S. and ZISSERMAN, A. "Lip Reading in the Wild". *Asian Conference on Computer Vision*. 2016 3.

[Doe16] DOERSCH, CARL. "Tutorial on variational autoencoders". *arXiv preprint arXiv:1606.05908* (2016) 2.

[EB03] ENGWALL, OLOV and BESKOW, JONAS. "Resynthesis of 3D tongue movements from facial data". *Eighth European Conference on Speech Communication and Technology*. 2003 3.

[FFX] FFX. *FaceFX*. URL: https://facefx.com/ (visited on 09/03/2021) 2.

[FHG*17] FABRE, DIANDRA, HUEBER, THOMAS, GIRIN, LAURENT, et al. "Automatic animation of an articulatory tongue model from ultrasound images of the vocal tract". *Speech Communication* 93 (2017), 63–75 3.

[HCC*14] HANNUN, AWNI, CASE, CARL, CASPER, JARED, et al. "Deep speech: Scaling up end-to-end speech recognition". *arXiv preprint arXiv:1412.5567* (2014) 2.

[HMT*12] HAHN, FABIAN, MARTIN, SEBASTIAN, THOMASZEWSKI, BERNHARD, et al. "Rig-space physics". *ACM transactions on graphics (TOG)* 31.4 (2012), 1–8 3.

[HSK16] HOLDEN, DANIEL, SAITO, JUN, and KOMURA, TAKU. "Learning Inverse Rig Mappings by Nonlinear Regression". *IEEE transactions on visualization and computer graphics* 23.3 (2016), 1167–1178 3, 6.

[HTC*13] HAHN, FABIAN, THOMASZEWSKI, BERNHARD, COROS, STELIAN, et al. "Efficient simulation of secondary motion in rig-space". *Proceedings of the 12th ACM SIGGRAPH/eurographics symposium on computer animation*. 2013, 165–171 3.

[JAL21] JALI. *Jali Research Inc.* 2021. URL: http://jaliresearch.com (visited on 09/03/2021) 2.

[JKHB20] JONELL, PATRIK, KUCHERENKO, TARAS, HENTER, GUSTAV EJE, and BESKOW, JONAS. "Let's Face It: Probabilistic Multi-modal Interlocutor-aware Generation of Facial Gestures in Dyadic Settings". *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents*. 2020, 1–8 8.

[KAL*17] KARRAS, TERO, AILA, TIMO, LAINE, SAMULI, et al. "Audio-driven facial animation by joint end-to-end learning of pose and emotion". *ACM Transactions on Graphics (TOG)* 36.4 (2017), 1–12 1–5.

[KAL*21] KARRAS, TERO, AITTALA, MIIKA, LAINE, SAMULI, et al. "Alias-Free Generative Adversarial Networks". *CoRR* abs/2106.12423 (2021). arXiv: 2106.12423 5.

[KB15] KINGMA, DIEDERIK P. and BA, JIMMY. "Adam: A Method for Stochastic Optimization". *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by BENGIO, YOSHUA and LECUN, YANN. 2015 6.

[KLA*20] KARRAS, TERO, LAINE, SAMULI, AITTALA, MIIKA, et al. "Analyzing and Improving the Image Quality of StyleGAN". *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, 8107–8116. DOI: 10.1109/CVPR42600.2020.00813 5.

[KLA19] KARRAS, TERO, LAINE, SAMULI, and AILA, TIMO. "A Style-Based Generator Architecture for Generative Adversarial Networks". *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, 4401–4410. DOI: 10.1109/CVPR.2019.00453 5.

[KW14] KINGMA, DIEDERIK P. and WELLING, MAX. "Auto-Encoding Variational Bayes". *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. 2014. arXiv: http://arxiv.org/abs/1312.6114v10 [stat.ML] 2, 4.

[LA10] LEWIS, JOHN P and ANJYO, KEN-ICHI. "Direct manipulation blendshapes". *IEEE Computer Graphics and Applications* 30.4 (2010), 42–50 3.

[LAK*16] LI, ZHI, AARON, ANNE, KATSAVOUNIDIS, IOANNIS, et al. "Toward a practical perceptual video quality metric". *The Netflix Tech Blog* 6.2 (2016) 10.

[LAR*14] LEWIS, JOHN P, ANJYO, KEN, RHEE, TAEHYUN, et al. "Practice and theory of blendshape facial models." *Eurographics (State of the Art Reports)* 1.8 (2014), 2 3.

[LT17] LIPTON, ZACHARY C. and TRIPATHI, SUBARNA. "Precise Recovery of Latent Vectors from Generative Adversarial Networks". *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net, 2017 5.

[LYLZ17] LUO, CHANGWEI, YU, JUN, LI, XIAN, and ZHANG, LEILEI. "HMM based speech-driven 3D tongue animation". *2017 IEEE International Conference On Image Processing (ICIP)*. IEEE. 2017, 4377–4381 3.

[LZG*21] LIN, JI, ZHANG, RICHARD, GANZ, FRIEDER, et al. "Anycost GANs for Interactive Image Synthesis and Editing". *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, 14986–14996 5.

[New12] NEWS, GUINNESS WORLD RECORDS. *Star Wars: The Old Republic Recognised Guinness World Records 2012 Gamer's Edition*. 2012. URL: https://www.guinnessworldrecords.com (visited on 12/17/2021) 2.

[OBP*12] ORVALHO, VERÓNICA, BASTOS, PEDRO, PARKE, FREDERIC I, et al. "A Facial Rigging Survey." *Eurographics (State of the Art Reports)* (2012), 183–204 3.

[PGM*19] PASZKE, ADAM, GROSS, SAM, MASSA, FRANCISCO, et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". *Advances in Neural Information Processing Systems 32*. Ed. by WALLACH, H., LAROCHELLE, H., BEYGELZIMER, A., et al. Curran Associates, Inc., 2019, 8024–8035 6.

[PvOS94] PELACHAUD, CATHERINE, van OVERVELD, CORNELIUS WAM, and SEAH, CHIN. "Modeling and animating the human tongue during speech production". *Proceedings of Computer Animation'94*. IEEE. 1994, 40–49 3.

[PWP18] PHAM, HAI XUAN, WANG, YUTING, and PAVLOVIC, VLADIMIR. "End-to-End Learning for 3D Facial Animation from Speech". *Proceedings of the 20th ACM International Conference on Multimodal Interaction*. ICMI '18. Boulder, CO, USA: Association for Computing Machinery, 2018, 361–365. ISBN: 9781450356923. DOI: 10.1145/3242969.3243017 2, 3.

[RLM*21] RICHARD, ALEXANDER, LEA, COLIN, MA, SHUGAO, et al. "Audio-and gaze-driven facial animation of codec avatars". *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2021, 41–50 3, 7.

[RZW*21] RICHARD, ALEXANDER, ZOLLHOEFER, MICHAEL, WEN, YANDONG, et al. "MeshTalk: 3D Face Animation from Speech using Cross-Modality Disentanglement". *arXiv preprint arXiv:2104.08223* (2021) 2, 3, 8.

[SG21] SG. *Speech Graphics*. 2021. URL: https://www.speech-graphics.com/ (visited on 09/03/2021) 2.

[SLY15] SOHN, KIHYUK, LEE, HONGLAK, and YAN, XINCHEN. "Learning structured output representation using deep conditional generative models". *Advances in neural information processing systems* 28 (2015) 4.

[TKY*17] TAYLOR, SARAH, KIM, TAEHWAN, YUE, YISONG, et al. "A deep learning approach for generalized speech animation". *ACM Transactions on Graphics (TOG)* 36.4 (2017), 1–11 2.

[TPL*20] TZIRAKIS, PANAGIOTIS, PAPAIOANNOU, ATHANASIOS, LATTAS, ALEXANDROS, et al. "Synthesising 3D facial motion from "In-the-Wild" speech". *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*. IEEE. 2020, 265–272 2, 3.

[VAPE20] VF, ABREVAYA, A, BOUKHAYMA, PH, TORR, and E, BOYER. "Cross-modal deep face normals with deactivable skip connections". *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, 4979–4989 4.

[VRS03] VERMA, ASHISH, RAJPUT, NITENDRA, and SUBRAMANIAM, L VENKATA. "Using viseme based acoustic models for speech driven lip synthesis". *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03)*. Vol. 5. IEEE. 2003, V–720 2.

[ZXL*18] ZHOU, YANG, XU, ZHAN, LANDRETH, CHRIS, et al. "Visemenet: Audio-driven animator-centric speech animation". *ACM Transactions on Graphics (TOG)* 37.4 (2018), 1–10 1–3.