Neural Dubber: Dubbing for Silent Videos According to Scripts

Chenxu Hu[1,3], Qiao Tian[2], Tingle Li[1], Yuping Wang[2], Yuxuan Wang[2], Hang Zhao[1]
[1]Tsinghua University [2]ByteDance [3]Zhejiang University

1. Introduction

Dubbing is a post-production process of re-recording actors' dialogues in a controlled environment (i.e., a sound studio), which is extensively used in filmmaking and video production. In real life, there are many professional voice actors who specialize in dubbing films, TV series, cartoons and other video products. In films' post-production stage, for example, voice actors may need to dub for the film clips due to the language barrier or accent of the original actor or some technical problems (e.g., hard to get clean sound when filming). Note that the pre-recorded high-definition video clips are not modified during the dubbing process. Voice actors are remarkably capable of dubbing according to lines with proper prosody such as stress, intonation and rhythm, which allows their speech to be synchronized with the pre-recorded video.

While dubbing is an impressive ability of professional actors, we aim to achieve this ability computationally. We name this novel task silent video dubbing (SVD): synthesizing human speech that is temporally synchronized with the given silent video according to the corresponding text. The main challenges of the task are two-fold: (1) temporal synchronization between synthesized speech and video, i.e., the synthesized speech should be synchronized with the lip movement of the speaker in the given video; (2) the content of the speech should be consistent with the input text.

Text-to-speech (TTS) synthesis is a task closely related to dubbing, which aims at converting given texts into natural speech. Most previous neural TTS models generate mel-spectrograms autoregressively [9] or non-autoregressively [7, 6] from input text, and then synthesize speech from the generated mel-spectrograms using vocoder [10]. However, several limitations prevent TTS from being applied in the dubbing problem: 1) TTS is an one-to-many mapping problem (i.e., multiple speech variations can be spoken from the same text) [6], so it is hard to control the variations (e.g., prosody, pitch and duration) in synthesized speech; 2) with only text as input, TTS can not make use of the visual information from the given video to control speech synthesis, which greatly limits its applications in dubbing scenarios where synthesized speech are required to be synchronized with the given video.

We introduce Neural Dubber, the first model to solve the SVD task. Neural Dubber is a multi-modal speech synthesis model, which generates high-quality and lip-synced speech from the given text and silent video. In order to control the duration and prosody of synthesized speech, Neural Dubber works in a non-autoregressive way following [7, 6]. The problem of length mismatch between phoneme sequence and mel-spectrogram sequence in non-autoregressive TTS is usually solved by up-sampling the phoneme sequence according to the predicted phoneme duration. Instead, we use the text-video aligner which adopts an attention module between the video frames and phonemes, and then upsample the text-video context sequence according to the length ratio between mel-spectrogram sequence and video frame sequence. The text-video aligner not only solves the length mismatch problem, but also allows the lip movement in the video to control the prosody of the generated speech.

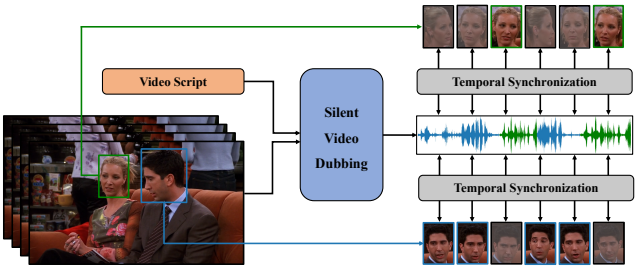


Figure 1: The schematic diagram of the silent video dubbing (SVD) task. Given the video script and the silent video as input, the SVD task aims to synthesize speech that is temporally synchronized with the video. This is a scene where two people are talking with each other.

In the real dubbing scenario, voice actors need to alter the timbre and tone according to different performers in the video. In order to better simulate the real case in the SVD task, we propose the image-based speaker embedding (ISE) module, which aims to synthesize speech with different timbres conditioning on the speakers' face in the multi-speaker setting. To the best of our knowledge, this is the first attempt to predict a speaker embedding from a face image with the goal of generating speech with a reasonable timbre that is consistent with the speaker's facial features (e.g., gender and age). This is achieved by taking advantage of the natural co-occurrence of faces and speech in videos without the supervision of speaker identity. With ISE, Neural Dubber can synthesize speech with a reasonable timbre according to the speaker's face. In other words, Neural Dubber can use different face images to control the timbre of the synthesized speech.

We conduct experiments on the chemistry lecture dataset

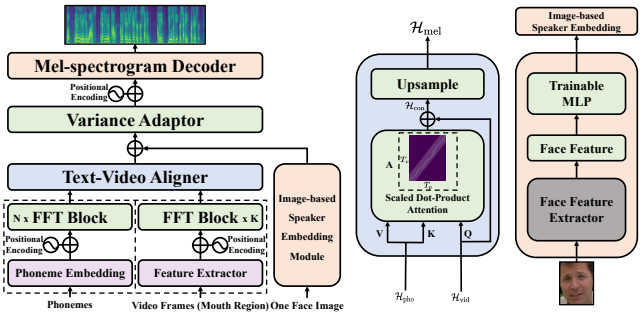
from Lip2Wav [4] for the single-speaker SVD, and the LRS2 [1] dataset for the multi-speaker SVD. The results of extensive quantitative and qualitative evaluations show that in terms of speech quality, Neural Dubber is on par with state-of-the-art TTS models [9, 6]. Furthermore, Neural Dubber can synthesize speech temporally synchronized with the lip movement in video. In the multi-speaker setting, we demonstrate that the ISE enables Neural Dubber to generate speech with reasonable timbre based on the face of the speaker, resulting in Neural Dubber outperforming FastSpeech 2 in a big margin in term of audio quality on the LRS2 dataset.

2. Method

In this section, we first introduce the novel silent video dubbing (SVD) task; we then describe the overall architecture of our proposed Neural Dubber; finally we detail the main components in Neural Dubber.

2.1. Silent Video Dubbing

Given a sentence T and a corresponding silent video clip V, the goal of silent video dubbing (SVD) is to synthesize natural and intelligible speech S whose content is consistent with T, and whose prosody is synchronized with the lip movement of the active speaker in the video V. Compared to the traditional speech synthesis task which only generates natural and intelligible speech S given the sentence T, SVD task is more difficult due to the synchronization requirement.



(a) Neural Dubber (b) Text-Video Aligner (c) ISE Module

Figure 2: The architecture of Neural Dubber.

2.2. Neural Dubber

2.2.1 Design Overview

Our Neural Dubber aims to solve the SVD task. Concretely, we formulate the problem as follows: given an phoneme sequence $S_p = \{P_1, P_2, \ldots, P_{Tp}\}$ and a video frame sequence $S_v = \{I_1, I_2, \ldots, I_{Tv}\}$, we need to predict a target mel-spectrogram sequence $S_m = \{Y_1, Y_2, \ldots, Y_{Tm}\}$.

The overall model architecture of Neural Dubber is shown in Figure 2. First, we apply a phoneme encoder f_p and a video encoder f_v to process the phonemes and images respectively. Note that the images we feed to the video encoder only contain mouth region of the speaker following [2]. We use S_v^m to represent these images. After the encoding, raw phonemes turn into $\mathcal{H}_{pho} = f_p(S_p) \in \mathbb{R}^{T_p \times d}$ while images turn into $\mathcal{H}_{vid} = f_v(S_v^m) \in \mathbb{R}^{T_v \times d}$. Then we feed $\mathcal{H}_{pho}$ and $\mathcal{H}_{vid}$ into the text-video aligner (see Section 2.2.3) and get the expanded sequence $\mathcal{H}_{mel} \in \mathbb{R}^{T_m \times d}$ with the same length as the target mel-spectrogram sequence S_m. Meanwhile, a face image randomly selected from the video frames is input into image-based speaker embedding (ISE) module (see Section 2.2.4) to generate a image-based speaker embedding. We add $\mathcal{H}_{mel}$ and ISE together and feed them into the variance adaptor to add some variance information (e.g., pitch and energy). Finally, we use the mel-spectrogram decoder to convert the adapted hidden sequence into mel-spectrogram sequence following [7, 6].

2.2.2 Phoneme and Video Encoders

The phoneme encoder and video encoder are shown in Figure 2a, which are enclosed in a dashed box. The function of the phoneme encoder and video encoder is to transform the original phoneme and image sequences into hidden representation sequences which contain high-level semantics. The phoneme encoder we use is similar to that in FastSpeech [7], which consists of an embedding layer and N Feed-Forward Transformer (FFT) blocks. The video encoder consists of a feature extractor and K FFT blocks. The feature extractor is a CNN backbone that generates feature representation for every input mouth image. And then we use the FFT blocks to capture the dynamics of the mouth region because FFT is based on self-attention [8] and 1D convolution where self-attention and 1D convolution are suit for capturing long-term and short-term dynamics respectively.

2.2.3 Text-Video Aligner

The most challenging aspect of the SVD task is alignment: (1) the content of the generated speech should come from the input phonemes; (2) the prosody of the generated speech should be aligned with the input video in time axis. So it does not make sense to produce speech solely from phonemes, nor video. In our design, the text-video aligner (Figure 2b) aims to find the correspondence between text and lip movement in the video first, so that synchronized speech audio can be generated in the later stage.

In the text-video aligner, an attention-based module learns the alignment between the phoneme sequence and the video frame sequence, and produces the text-video context

sequence. Then an upsampling operation is performed to change the length of the text-video context sequence $\mathcal{H}_{con}$ from T_v to T_m. In practice, we adopt the popular Scaled Dot-Product Attention [8] as the attention module, where $\mathcal{H}_{vid}$ is used as the query, and $\mathcal{H}_{pho}$ is used as both the key and the value. After the attention module, we get the text-video context sequence, i.e., the expanded sequence of phoneme hidden representation by linear combination. In the attention module, the obtained monotonic alignment between video frames and phonemes contributes to the synchronization between the synthesized speech and the video on fine-grained (phoneme) level.

In practice, the length of a mel-spectrograms sequence is n times that of a video frame sequence. We denote the n as $n = T_{mel}/T_v$. We upsample the text-video context sequence $\mathcal{H}_{con}$ to $\mathcal{H}_{mel}$ with scale factor is n. After that, the length of the text-video context sequence is expanded to that of the mel-spectrograms sequence. Thus, the problem of length mismatch between the phoneme and mel-spectrograms sequence is solved. Because of the attention between video frames and phonemes, the speed and part of prosody of synthesized speech are controlled by input video explicitly, which makes the synthesized speech and input video well synchronized.

2.2.4 Image-based Speaker Embedding Module

Image-based speaker embedding (ISE) module (Figure 2c), a new multi-modal speaker embedding module that we propose, generates an embedding that encapsulates the characteristics of the speaker's voice from an image of his/her face. We randomly select a face image I_i^f from $S_v^f = \left\{I_1^f, I_2^f, \ldots, I_{Tv}^f\right\}$, and obtain a high-level face feature by feeding the selected face image into a pre-trained and fixed face recognition network. Then we feed the face feature to a trainable MLP and gain the ISE. The predicted ISE is directly broadcasted and added to $\mathcal{H}_{mel}$ so as to control the timbre of synthesized speech. Our model learns face-voice correlations which allow it to produce speech that coincides with various voice attributes of the speakers (e.g., gender and age) inferred from their face.

3. Experiments and Results

3.1. Datasets

In the single-speaker setting, we evaluate Neural Dubber on the chemistry lecture dataset from Lip2Wav [4]. After data segmentation and cleaning, the dataset contains 6,640 samples , with the total video length of approximately 9 hours. In the following subsections, we refer to this dataset as chem for short. In multi-speaker setting, we evaluate Neural Dubber on the LRS2 [1] dataset. Note that we only train on the training set of the LRS2 dataset, which only contains data of approximately 29 hours.

3.2. Model Configuration

Neural Dubber Our Neural Dubber consists of 4 feed-forward Transformer (FFT) blocks [7] in the phoneme encoder, the mel-spectrogram decoder, and 2 FFT blocks in the video encoder. The feature extractor in the video encoder is the ResNet18 except for the first 2D convolution layer being replaced by 3D convolutions. The variance adaptor contains pitch predictor and energy predictor [6].

Baseline We propose a baseline model based on the Tacotron [9] system with some modifications which make it fit to the new SVD task. We call this baseline model **Video-based Tacotron**. We concatenate the spectrogram frames with the corresponding $\mathcal{H}_{vid}$, and use it as the decoder input to make use of the information in video.

3.3. Evaluation

3.3.1 Metrics

Since the SVD task aims to synthesize human speech synchronized with the video from text, the audio quality and the audio-visual synchronization (av sync) are the important evaluation criteria. We conduct the mean opinion score (MOS) evaluation on the test set to measure the audio quality and the av sync. For each video clip, the raters are asked to rate scores of 1-5 from bad to excellent (higher score indicates better quality) on the audio quality and the av sync, respectively. In order to measure the av sync quantitatively, we use the pre-trained SyncNet [3] following [4]. We adopt two metrics: Lip Sync Error - Distance (LSE-D) and Lip Sync Error - Confidence (LSE-C) from Wav2Lip [5].

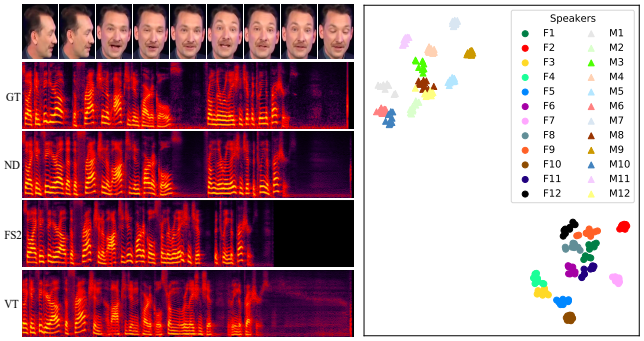
3.3.2 Single-speaker SVD

We first conduct MOS evaluation on the chem single-speaker dataset, to compare the audio quality and the av sync of the video clips generated by Neural Dubber (ND) with other systems, including 1) GT, the ground-truth video clips; 2) GT-MEL, where we first convert the ground-truth audio into mel-spectrograms, and then convert it back to audio using Parallel WaveGAN; 3) FastSpeech 2 (FS2); 4) Video-based Tacotron (VT). Note that the systems in 2), 3), 4) and Neural Dubber use the same pre-trained Parallel WaveGAN for a fair comparison. In addition, we compare Neural Dubber with those systems on the test set using the LSE-D and LSE-C metrics. The results for single-speaker SVD are shown in Table 1. It can be seen that Neural Dubber can surpass the Video-based Tacotron baseline and is on par with FastSpeech 2 in terms of audio quality, which demonstrates that Neural Dubber can synthesize

Method	Audio Quality	AV Sync	LSE-D ↓	LSE-C ↑
GT	3.93 ± 0.08	4.13 ± 0.07	6.926	7.711
GT-MEL	3.83 ± 0.09	4.05 ± 0.07	7.384	6.806
FS2	3.71 ± 0.08	3.29 ± 0.09	11.86	2.805
VT	3.55 ± 0.09	3.03 ± 0.10	11.79	2.231
ND	3.74 ± 0.08	3.91 ± 0.07	7.212	7.037

Table 1: The evaluation results for the single-speaker SVD.

high-quality speech. Furthermore, in terms of the av sync, Neural Dubber outperforms FastSpeech 2 and Video-based Tacotron in a big margin and matches GT (Mel + PWG) system in both qualitative and quantitative evaluations, which shows that Neural Dubber can control the prosody of speech and generate speech synchronized with the video. We also show a qualitative comparison in Figure 3a which contains mel-spectrograms of audios generated by the above systems. It shows that the prosody of the audio generated by Neural Dubber is closed to that of ground truth recording, i.e., well synchronized with the video.



(a) Mel-spectrograms (b) Speaker embeddings

Figure 3: Some visualizations.

3.3.3 Multi-speaker SVD

Similar to Section 3.3.2, we conduct human evaluation and quantitative evaluation on the LRS2 dataset. Due to the failure of Video-based Tacotron in single-speaker SVD, we no longer compare our model with it. The results are shown in Table 2. We can see that Neural Dubber outperforms FastSpeech 2 in a significant margin in terms of audio quality, exhibiting the effectiveness of ISE in multi-speaker SVD. The qualitative and quantitative evaluations show that the speech synthesized by Neural Dubber is much better than that of FastSpeech 2 and is on par with the ground truth recordings in terms of av sync. These results show that Neural Dubber can address the more challenging multi-speaker SVD task.

Some audio clips are generated by Neural Dubber with the same phoneme sequence and mouth image sequence but different speaker face images as input. We select 12 males and 12 females from the test set of the LRS2 dataset. For each person, we chose 10 face images with different head posture, illumination and facial makeup, etc. We visualize the voice embedding of these audios in Figure 3b, which are generated by a pre-trained speaker encoder. It can be seen that the utterances generated from the images of the same speaker form a tight cluster, and that the cluster representing each speaker is separated from each other. In addition, there is a distinctive discrepancy between the speech synthesized from the face images of different genders. It concludes that Neural Dubber can use the face image to alter the timbre of the generated speech.

Method	Audio Quality	AV Sync	LSE-D ↓	LSE-C ↑
GT	3.97 ± 0.09	3.81 ± 0.10	7.214	6.755
GT-MEL	3.92 ± 0.09	3.69 ± 0.11	7.317	6.603
FS2	3.15 ± 0.14	3.33 ± 0.10	10.17	3.714
ND	3.58 ± 0.13	3.62 ± 0.09	7.201	6.861

Table 2: The evaluation results for the multi-speaker SVD.

References

[1] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Deep audio-visual speech recognition. *TPAMI*, 2018. 2, 3

[2] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Lip reading sentences in the wild. In *CVPR*, pages 3444–3453. IEEE, 2017. 2

[3] J. S. Chung and A. Zisserman. Out of time: automated lip sync in the wild. In *ACCV*, pages 251–263. Springer, 2016. 3

[4] K. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. Jawahar. Learning individual speaking styles for accurate lip to speech synthesis. In *CVPR*, pages 13796–13805, 2020. 2, 3

[5] K. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. In *ACM Multimedia*, pages 484–492, 2020. 3

[6] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu. Fastspeech 2: Fast and high-quality end-to-end text to speech. In *ICLR*, 2021. 1, 2, 3

[7] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu. Fastspeech: Fast, robust and controllable text to speech. In *NeurIPS*, 2019. 1, 2, 3

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *NeurIPS*, 2017. 2, 3

[9] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, et al. Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135*, 2017. 1, 2, 3

[10] R. Yamamoto, E. Song, and J.-M. Kim. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP*, pages 6199–6203. IEEE, 2020. 1