Personalized Visual Dubbing through Virtual Dubber and Full Head Reenactment

Bobae Jeon

A Thesis

in

The Department

of

Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Computer Science (Computer Science) at

Concordia University

Montréal, Québec, Canada

December 2024

© Bobae Jeon, 2025

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Bobae Jeon**

Entitled: **Personalized Visual Dubbing through Virtual Dubber and Full Head Reenactment**

and submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science (Computer Science)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Name of the Chair

_____ External Examiner
Dr. Name of External Examiner

_____ Examiner
Dr. Name of Examiner One

_____ Supervisor
Dr. Sudhir Mudur

_____ Co-supervisor
Dr. Tiberiu Popa

Approved by _____
Dr. Joey Paquet, Chair
Department of Computer Science and Software Engineering

_____ 2024

_____
Dr. Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Personalized Visual Dubbing through Virtual Dubber and Full Head Reenactment

Bobae Jeon

Visual dubbing aims to modify facial expressions to ”lip-sync” a new audio track. While person-generic talking head generation methods have made significant progress in expressive lip synchronization across arbitrary identities, they usually lack person-specific details and fail to generate high-quality results. On the other hand, person-specific approaches enable realistic identity preservation and high lip-sync quality but require extensive training, limiting their adaptability in real-world applications.

Our method combines the strengths of both approaches to generate balanced results in lip synchronization and visual quality while achieving training efficiency. To this end, our pipeline incorporates a virtual dubber, a person-generic talking head, as an intermediate representation. This simplifies identity swapping, enhances efficiency, and improves both visual quality and expression accuracy.

Key innovations include full-head identity swapping and reenactment, eliminating artifacts such as the double chin effect while ensuring temporal stability. Through extensive quantitative and qualitative evaluations, we demonstrate that our approach achieves a superior balance between lip-sync accuracy and realistic facial reenactment.

Furthermore, we validate the robustness of our method with experiments in challenging real-world scenarios, including tilted head poses and facial occlusions. Notably, our pipeline operates effectively with short video clips, emphasizing its efficiency and practicality.

Acknowledgments

Text of acknowledgments.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Visual dubbing is a video synthesis task that aims to synchronize the facial expression of an actor with arbitrary audio, such as translated speech. While traditional dubbing only replaces the original audio, visual dubbing addresses the mismatch between mouth movements and the new audio. It is highly applicable to real-world scenarios such as movies, news broadcasts, and advertisements to ensure a more natural viewing experience by aligning the visual and audio components.

The ultimate purpose of visual dubbing is to generate a natural-looking "lip-sync" video in these practical scenarios. Specifically, the output videos must retain the original actor identity and pose, while preserving the background and matching the production-level visual quality of the original actor videos, only modifying mouth expressions to follow the dubber speech. Achieving these goals presents significant challenges, as the synthesized video must balance lip synchronization, visual quality, temporal consistency, and person-specific details.

Recent advancements in lip-syncing have been driven by person-generic talking head generation methods that generalize across multiple identities. These approaches are typically trained on diverse datasets with various identities and expressions. This training enables generalization on arbitrary identities and these methods have shown great performance in generating expressive mouth movements. However, the outputs of these methods often fail to achieve the high visual quality required for real-world videos. Moreover, they struggle to preserve person-specific details, often resulting in uncanny or unnatural facial features. Furthermore, many of these methods rely on a single-image as input, focusing on synthesizing the face. This inherently results in static backgrounds and body

parts, making it challenging to preserve the dynamic pose and background from the original video. This limitation makes it challenging to directly adapt these techniques to real-world visual dubbing scenarios, where maintaining all parts of the video except for the mouth expressions as original is crucial.

In contrast, person-specific methods aim to train customized models tailored to specific individual. Previously, they have shown realistic identity preservation and good lip-sync quality, but often require large amounts of person-specific data for training. This presents a significant drawback to be applied to many real-world scenarios. For example, TV ads are typically less than 30 seconds long, which is insufficient for training such models.

To address these limitations, we propose a novel pipeline that combines the strengths of both person-generic and person-specific approaches. Our method introduces a virtual dubber, a static background talking head generated from a person-generic talking head method, as an intermediate representation. This virtual dubber captures the dubber's expressive mouth movements while preserving the actor's identity. Although the virtual dubber is not suitable as a final output due to pose differences, static backgrounds, and occasional uncanny facial features, it serves as a powerful intermediate step. By leveraging the expressiveness of person-generic methods while reducing the need for extensive data typically required for person-specific model training, the virtual dubber bridges the gap between the two approaches.

Building on the work of Patel et al. [1], we use an autoencoder-based identity swapping network to transfer the actor's identity onto the virtual dubber frame-by-frame. The virtual dubber provides a closer representation of the desired output as it already retains some aspects of the actor identity, simplifying the identity swap and making the overall process more efficient. Conversely, the approach by Patel et al. requires a second identity transfer pass to address the mismatch between the actor's mouth style and the dubber expressions in their initial output. This additional step not only significantly adds complexity to the pipeline, but also tends to average expressions, leading to less expressive results. Furthermore, we perform identity swap in a full head, including the hair and neck, rather than limiting it to the face. This extended identity swap not only allows better preservation in identity and visual quality, but also enables adjusting the size of the face parts, such as jaws, helping avoid undesirable artifacts such as the double chin as discussed next.

The face reenactment step of Patel et al. relies on landmark-based compositing of the mouth region, which introduces temporal inconsistencies (e.g., jittering) due to alignment errors. Additionally, when the original actor face has an open mouth and a wider jaw, but the synthesized face has closed mouth, the mismatch between facial geometry results in unnatural appearance, resembling a double chin. These issues arise because landmark-based compositing does not allow adjustments to facial features. To address these challenges, we reenact the extended head region generated by the identity swapping network. Our method synthesizes and directly reenacts the full head, eliminating the need for mouth-region compositing. As a result, we resolve the double chin effect and improve temporal consistency.

Figure 1.1 illustrates challenges in earlier approaches: pose and background preservation (SadTalker [2]), visual quality– blurred face (Wav2Lip [3]), identity preservation with unnatural eyes (LivePortrait [4]), and double chin effect– the jaw is too large in the synthesized image (Patel et al. [1]).

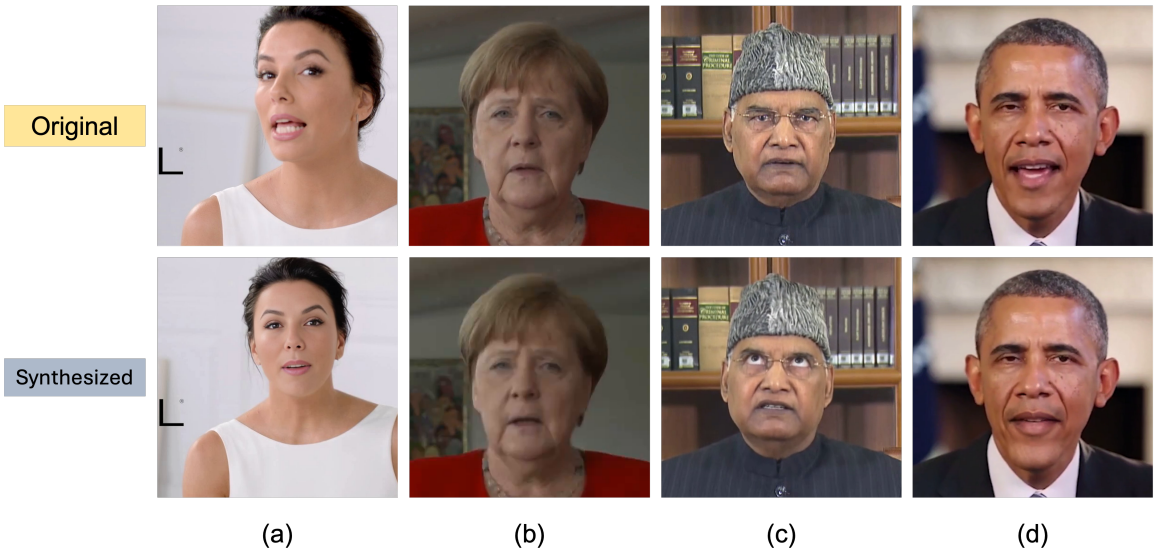

Figure 1.1: Challenges of earlier approaches. Note that ideally, the synthesized faces should appear natural, with only the expressions modified from the original. (a) pose and background preservation (SadTalker [2]), (b) visual quality (Wav2Lip [3]), (c) identity preservation (unnatural eyes) (LivePortrait [4]), (d) face artifacts– double chin (Patel et al. [1])

We introduce several novel strategies to achieve high-quality and realistic visual dubbing, addressing these limitations. Our contributions can be summarized as follows:

(1) **Introduction of a virtual dubber**: An expressive intermediate representation that accelerates

identity swap while preserving expression.

(2) **Full-head identity swap**: Effectively preserves identity and visual quality.

(3) **Full-head reenactment**: Ensures temporal consistency and eliminates artifacts such as the double chin effect.

(4) **Extensive evaluation on real-world videos**: We demonstrate our method achieves balanced results in lip syncing and visual quality, outperforming the competitive approaches.

Chapter 2

Background

2.1 Visual Dubbing

Dubbing is a post-processing step in video production that replaces the audio, often with a translation into another language. It allows the production of a video with various languages, without affecting the original video. Therefore, it is convenient for viewers who speak different languages as it eliminates the need to read subtitles. Voice actors record the translated speech in another language, and it replaces the original speech later in the post-production. However, only the audio is altered in traditional dubbing, so the mouth expressions remain the same as the original language. Due to this discrepancy in mouth motion and audio, the viewers can find the dubbed video unnatural.

Visual dubbing is introduced to address the ”unmatched mouth” problem of traditional dubbing. It involves synthesizing mouth and lip motion to follow the new speech. Early works [5, 6] map the audio features to the facial features of the original video and find particular phonemes to match the new audio. Since they rearrange the original video sequence and match the expression, they can only reconstruct the mouth expression from the original video. Later, [7] introduced a learnable approach to synthesize unseen facial expressions using mouth appearance and shapes. More recently, deep learning-based approaches have become dominant as they have shown the ability to generate unseen expressions more effectively.

Visual dubbing is a complex task that requires syncing the facial expression to the new speech, ensuring viewers can process the video naturally. Therefore, it is required to have realistic and

expressive mouth motion synchronized to the audio while preserving the original actor's identity and maintaining the production quality of the video.

2.2 Image Generative Models

Image generative models are fundamental to facial synthesis, offering the ability to generate unseen photorealistic faces and enabling face manipulation. Here, we introduce the most prominent architectures in this domain.

2.2.1 Autoencoders (AE)

Encoder-Decoder Models

In encoder-decoder models, the encoder E processes the input x, converting it to a high-dimensional latent vector z that retains only the most essential information. This compressed representation passes through a bottleneck or submodules S that adapt the latent representation for specific tasks. The decoder D then processes the output of S to generate the final output.

This architecture serves as a common backbone for many generative models. It is highly flexible and adaptable for a wide range of different tasks, as its modularity allows submodule integration with different functionalities between the encoder and decoder.

$$F(x) = D(S(E(x))) \tag{1}$$

Standard Autoencoders

While general encoder-decoder models are not restricted to having the same input-output domain, autoencoders are a specific type of encoder-decoder architecture designed to reconstruct the input. In the face generation context, the encoder learns the high-level features of the face, and the bottleneck lowers its dimensionality. Then the decoder maps the latent representation from the bottleneck to re-generate the input face. Although it can reconstruct input faces, the latent space is too compressed to generate unseen faces.

Variational Autoencoders (VAE)

Variational autoencoders (VAE) [8] are one of the AE modifications designed to improve the generation capabilities. Unlike standard AE, the encoder in VAE encodes the input x into mean μ and standard deviation σ parameters of a Gaussian distribution:

$$E(x) \rightarrow (\mu_x, \sigma_x) \tag{2}$$

The model uses a probabilistic latent space, which is regularized to approximate a standard normal Gaussian distribution during training. From the latent space, the latent representation z is sampled using the encoded parameters, where $\epsilon \sim \mathcal{N}(0, 1)$:

$$z = \mu_x + \epsilon \cdot \sigma_x \tag{3}$$

Since it uses a continuous latent space, unseen data can be generated by sampling new latent vectors from the prior Gaussian distribution and passing them through the decoder.

Typical AE-based approaches are relatively easy to train because of the straightforward architecture. Although they generally produce stable outputs, their outputs are less expressive and prone to averaging. In addition, they typically generate low-quality images, lacking fine details.

2.2.2 Generative Adversarial Network (GAN)

Generative Adversarial Networks (GAN) [9] consist of two main components: a generator and a discriminator. The generator learns to create realistic images from a latent vector z, and the discriminator learns to distinguish between real images from the training dataset and synthetic images from the generator. The term "adversarial" refers to the training process where the goal of the generator is to output highly realistic images to fool the discriminator, while the discriminator is trained to label the real and generated images. Formally, we can consider the generator and the discriminator to be minimax game players with a value function $V(D, G)$.

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[1 - \log D(G(z))] \tag{4}$$

StyleGAN

StyleGAN [10] enhances the generator in GANs to produce photorealistic images with improved disentanglement and control. Unlike the vanilla GAN generator, StyleGAN uses a mapping network to transform the input into a latent code w in an intermediate latent space W. This intermediate latent space is more disentangled than that of traditional GANs, enabling the generator to create more realistic images effectively. The output of the mapping network is affine-transformed into a style vector, which controls the generator through the Adaptive Instance Normalization (AdaIN) layer formulated as:

$$\text{AdaIN}(x_i, y) = y_{\text{s,i}} \frac{x_i - \mu(x_i)}{\sigma(x_i)} + y_{\text{b,i}} \tag{5}$$

The AdaIN operation normalizes each feature map, and then applies scaling and bias based on the style vector. Additionally, StyleGAN introduces noise inputs to generate stochastic details, such as variations in hair and skin. This design successfully separates high-level features (styles) from stochastic details. This separation also enables style mixing during inference by injecting different latent codes w, improving controllability in face synthesis.

Although StyleGAN yields high-resolution photorealistic image synthesis, it has shown characteristic artifacts in its results– such as "blob"-like artifacts. To address these issues, StyleGAN2 [11] has been introduced. It modifies the generator to replace AdaIN which introduces artifacts. Instead, they apply demodulation operation to the weights of convolutional layers. These changes result in an improvement in the output image quality with fewer artifacts.

Building on StyleGAN2, StyleGAN3 [12] has shown further improvements. It solves the "texture sticking" problem from StyleGAN2, in that the face details are fixed to the same pixel position when interpolating the latent code. It redesigns the architecture alias-free, using continuous signal interpretation. It has shown improved geometric consistency in the generated images, especially under transformations.

GANs have demonstrated a remarkable ability to generate high-resolution, photorealistic images. However, their adversarial training scheme makes them more challenging to train and prone to dramatic failures, often resulting in uncanny or unrealistic outputs compared to autoencoders.

2.2.3 Diffusion Models

Diffusion models [13] are one of the most promising methods in face manipulation, along with GAN. They work by gradually adding Gaussian noise to the input data according to a variance schedule $\beta_1, ..., \beta_T$, where $x_1, x_2, ..., x_T$ is a generation sequence, in the forward diffusion process:

$$q(x_{1:T}|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1}), \quad q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I) \tag{6}$$

The model learns to reverse this by removing the noise to recover the original data.

$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \quad p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)) \tag{7}$$

Recently, diffusion models have gained a reputation for generating high-fidelity results in image synthesis, often outperforming GANs in visual quality [14, 15]. They also allow for flexible conditioning, such as text or audio guidance, through integrating the embeddings. However, they are slower in generation because of the iterative denoising process during inference. In addition, they require large datasets to train, making them less accessible.

Among generative models, we leverage an autoencoder-based model due to their ease of training and stable results. However, as discussed earlier it has limited generation ability in unseen faces. To address this, we incorporate a StyleGAN-based submodule.

2.2.4 3D Face Representation

Some face synthesis frameworks leverage 3D face representation techniques along with generative models. In this combination, 3D face representation provides controllability for facial attributes to generative models. By parameterizing and modeling the facial features in the 3D space, they enable modifications in the face, which is difficult to achieve in the original 2D image. The most prominent 3D face representation methods include 3D face reconstruction and neural radiance field (NeRF) [16]. In our framework, we utilize 3D face reconstruction as an intermediate representation,

which produces a face mesh to enable head pose transfer.

3D Face Reconstruction

3D face reconstruction is a process of generating the 3D face model from the 2D image. Many approaches predict the camera pose and aim to reconstruct the face in the 3D space. Common outputs include face meshes and depth maps, which explicitly encode facial geometry. This disentanglement in facial geometry enables facial editing. One of the most common approaches within 3D face reconstruction is 3D morphable models (3DMM) [17]. 3DMM represents a face by a parameterized model. It encodes the facial attributes such as shape, expression, and texture, which are not easily measured. These coefficients define a 3D facial geometry, typically represented as a mesh. In face synthesis and reenactment, 3DMM provides a controllable framework, allowing the manipulation of facial attributes.

Neural Radiance Fields

Neural radiance fields (NeRF) [16] represent 3D scenes in a continuous parameterized space using a neural network for volume rendering. It is typically used in the neural rendering process rather than as an intermediate representation. While earlier works on 3D face representation typically focus on monocular 3D reconstruction (single-view), NeRF allows multi-view synthesis, which enables rerendering images from novel views of input images. Unlike traditional 3D face models that use explicit representations such as meshes, it implicitly represents the shape and appearance of the face in a continuous 3D space. This neural rendering technique demonstrated high-fidelity output with fine details and complex lighting effects [16, 18, 19].

2.3 Face Reenactment

Face reenactment is a conditional face synthesis task where the target face is manipulated to change specific facial attributes such as expressions, eye gaze, and head pose. Early works utilize encoder-decoder architecture and disentangle facial features in the latent space to interpolate them [20, 21]. However, they failed to produce photorealistic results.

More recently, some methods have leveraged 3D face models. They typically render the image in 3D space, deform face features, and re-render the synthesized face. For instance, Face2Face [22] uses 3D Morphable Model (3DMM) coefficients to parameterize facial attributes. They employ deformation transfer to transfer the expression components from the source to the target. Similarly, Ma et al. [23] utilize facial geometry from the input for reenactment.

Another prominent line of work is GAN-based methods, following the remarkable success of GANs in generating realistic synthetic images. Some methods [24, 25] learn facial boundaries and transfer them to the target face. FSGAN [26] employs landmarks to train a recurrent neural network and uses segmentation information to complete the reenactment. Similarly, FReeNet [27] performs landmark-guided face reenactment by utilizing a landmark converter. Unlike those methods based on 2D landmarks, MarioNETte [28] transforms detected 3D landmarks into identity-independent to reenact arbitrary faces. Meanwhile, ICface [29] uses head pose and action units (AU) [30] to control face attributes. On the other hand, Yao et al. [31] propose a method that preserves the source identity by learning the appearance adaptive parameters. Furthermore, other approaches [32, 33] use 3DMM coefficients as guidance to warp the face features.

With recent advances in GANs, several methods have leveraged pretrained GAN models, particularly StyleGAN2 [11]. Some approaches [34, 35] disentangle head pose, expression, and identity from the source and target latent codes from pretrained StyleGAN2. Based on the disentangled facial features, new latent code is generated to synthesize an image that transfers the head pose and expression of the target to the source. Alternatively, Bounareli et al. [36] disentangle the appearance and pose features but updates the weights of the StyleGAN2 generator, instead of the latent codes. These methods are simple and effective in one-shot settings; however, they require GAN inversion when applied to real-world images, which often struggles to faithfully reconstruct the input image.

Chapter 3

Related Work

Visual dubbing approaches can be classified into two categories: person-generic and person-specific methods. A summary of these approaches is provided in Table 3.1.

	Visual Quality	Lip Synchronization	Identity Preservation	Efficiency
Person Generic Methods	✗	✓	✗	✓
Person Specific Methods	✓	✓	✓	✗
Ours(Combined Method)	✓	✓	✓	✓

Table 3.1: Summary of visual dubbing methods

3.1 Person Generic Methods

In person-generic methods, a universal model is trained on multiple identities. These approaches have gained popularity for their ability to process arbitrary identities without further training.

One line of methods utilizes GAN [3,37–42] . Wav2lip [3] leverages reconstruction-based learning to align arbitrary speech with lip movements, achieving highly accurate lip synchronization. TalkLip [41] incorporates a lip-reading module that transcribes image sequences into text, allowing the generator to be penalized for generating incorrect facial images. More recently, DaGAN [39] has utilized facial keypoints derived from the 3D geometry of the face to generate image-driven talking heads.

Recent works based on diffusion models have shown remarkable success in audio-driven talking head generation [42–46]. DiffDub [45] employs a diffusion-based autoencoder [47] to utilize latent

representations rather than explicit ones like blendshapes or landmarks for image reconstruction. It builds on the person-specific method DAE-Talker [48] as its backbone and extends it to generalize across unseen identities. AniPortrait [42], in contrast, uses a 2D projection of a 3D facial mesh and head pose extracted from audio as an intermediate representation.

Other prominent approaches use 3D geometry as an intermediate representation for audio-driven talking head generation [2, 49–52]. While most of these methods rely on 3DMM [53] coefficients to disentangle facial features, approaches like HiDe-NeRF [49], GeneFace [51], and SyncTalk [52] also incorporate NeRF [16] for 3D geometry modeling.

Some methods leverage keypoints– either explicit or implicit– to guide facial animation [4, 42, 54–58]. Explicit keypoint-based methods typically rely on facial landmarks [42, 54–57] as intermediate representation. Meanwhile, other approaches, such as LivePortrait [4] and MCNet [58], utilize implicit keypoints. Specifically, LivePortrait builds upon Face Vid2Vid [59], employing a canonical keypoint detector and transforming the output into implicit keypoints with extracted head pose and expression.

While these methods are generally expressive and generalizable, they often fail to preserve the unique talking style of the original actor. Particularly image-driven, "one-shot" methods [2, 4, 42, 50, 54, 57, 60] are prone to this as a single image lacks sufficient information to synthesize a full video. For example, when driven by an image with a closed mouth, these methods cannot accurately generate the specific mouth interior of an unseen actor. Moreover, they are inherently restricted to static backgrounds and often unable to replicate the original movement. Furthermore, these methods are often pretrained on low-resolution datasets, which restricts their ability to handle high-resolution real-world videos, leading to low-quality outputs.

3.2 Person Specific Methods

Person-specific methods focus on building a specialized model for each identity. A common approach involves synthesizing faces based on audio input.

Suwakanakorn et al. [61] use a recurrent neural network (RNN) to map audio to mouth shapes. While this methods allows for generating decent lip sync and restores the original identity, it requires

enormous amounts of data (e.g., 17 hours of Obama video). The study emphasizes that having a large training dataset is critical. Additionally, it involves the composite in the lower face, which leads to the occasional double chin effect due to 3D geometry failures.

Thies et al. [62] propose a two-stage training scheme to employ generalized audio-expression mapping from multiple identities and personalize it to the target identity. By using the linear mapping of the audio-expression blendshapes, the method employs neural face rendering for face reenactment with generated expressions. The neural face rendering network is fine-tuned on the target video.

LipSync3D [63] follows an encoder-decoder architecture that predicts a 3D mesh from audio, generating output by rendering the mesh and blending the lower face region. This method disentangles the 3D information of the image, enabling normalization of lighting and pose for better audio-to-mouth expression mapping.

These approaches utilize 3D models as an intermediate representations. Although they can capture face dynamics and abstract them, they lack fine facial details. In contrast, AD-Nerf [64] introduces an approach that directly feeds input audio into the neural radiance field. Using two neural radiance fields, it not only generates the head but also synthesizes the torso accordingly. This approach effectively addresses unnatural talking face animations where only the head moves.

SyncTalkFace [65] generates talking heads in an end-to-end manner with an encoder-decoder architecture. By leveraging lip and audio features generated from respective encoders, its memory network learns to map audio to lip expressions. Their facial decoder generates synthesized faces based on the identity and audio features.

More recently, DAE-Talker [48] utilizes a diffusion autoencoder to predict latent representations from speech, which are used to generate images through an autoencoder.

Audio-based models can generate unseen expressions to synchronize the speech. However, their expressions are often not realistic enough since they have to rely only on actor videos.

On the other hand, methods like [66, 67] synthesize the original actor video with target expressions directly from dubber video. Both methods first disentangle the 3D model parameters and use a recurrent generative adversarial network to translate facial expression parameters into video

frames. Along with translated expressions, the 3D model parameters are used to generate the full-head synthesized video. These methods achieve photorealistic result with well-maintained target expressions.

Person-specific models require training per person. This identity-specific training enables greater faithfulness to the original identity and visual quality on which they are trained. However, the main drawback of these methods is their typically high data requirements, ranging from a few minutes [62, 64, 67] to hours [61] of video, even when they are pretrained on various identities [62, 64]. Despite their ability to preserve the original identity and generate photorealistic frames, these high data requirements make such methods impractical in many real-world situations. Unlike these methods, our work follows a person-specific architecture while significantly reducing the required training data to just a few seconds.

3.3 Combined Methods

Some methods combine these approaches to address the limitations by pretraining a model on a large dataset to generalize facial features, followed by fine-tuning it for a specific identity. They typically aim to synthesize an original video from the given audio.

DFRF [68] is based on NeRF and uses 3D-aware reference image features to condition a dynamic facial radiance field, which improves generalizability of typical NeRF-based methods.

Dubbing for Everyone [69] adapts a prior deferred neural rendering network. It uses 3DMM parameters to generate a 3D representation of the face, modifying the expression parameters based on the given audio. Fine-tuning is performed using neural textures, leveraging both person-generic and person-specific data to balance generalizability and person-specific details.

StyleSync [70] modifies StyleGAN2 [11], using masked target frames, with the training objective to recover the unmasked target image. This method encodes both audio dynamics and facial information into the style space.

While these methods are highly generalizable and capable of preserving the identity of the original video, they share a common issue with other audio-based methods: unrealistic expressions.

Loosely inspired by these methods, our pipeline targets person-specific visual dubbing while

leveraging a person-generic method. We utilize a video-based approach, ensuring more realistic synthesized expressions while maintaining a balance between expressiveness, visual quality, and identity preservation.

Chapter 4

Methodology

We leverage the visual dubbing pipeline from Patel et al. [1] as a backbone, since their modular approach to disentangle the visual dubbing components has shown the capability to produce high-quality outputs that faithfully preserve actor identity. Our enhancement of this pipeline focuses on improving lip synchronization and temporal consistency while maintaining the identity-preserving qualities and high-quality results.

Following Patel et al., we parameterize visual dubbing task as follows: head pose, background, identity, expression, and image quality. Our pipeline consists of three main stages that handle different parameters: preprocessing, a full-head identity swapping, and postprocessing.

In the preprocessing stage, we first create a "virtual dubber" using a static talking head to contain dubber expression with actor identity. Sequentially, we disentangle head pose and background, then transfer them from actor to virtual dubber.

This is followed by full-head identity swapping, where we disentangle identity and expression parameters. Our identity swapping network transfers the actor identity to the virtual dubber, with a goal of synthesizing frames with actor identity and dubber expression.

Finally, the postprocessing stage addresses image quality. In this stage, we focus on enhancing resolution and preserving identity-specific details to achieve production-quality results. The overall pipeline is illustrated in Figure 4.1.

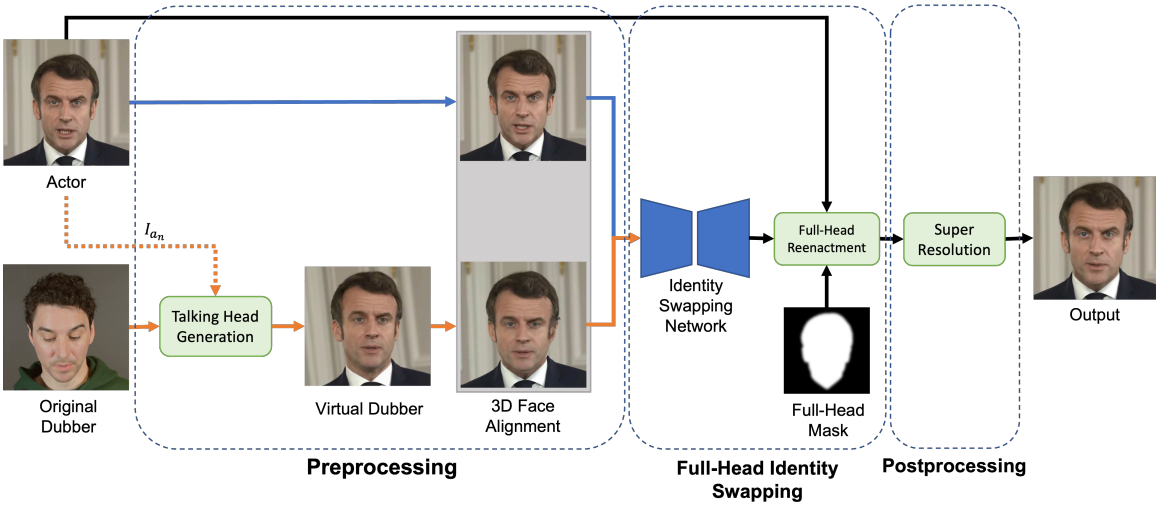

Figure 4.1: Our pipeline

4.1 Preprocessing

4.1.1 Virtual Dubber

For the generation of virtual dubber, we use LivePortrait [4] which generates a talking head given a driving video and a reference image. We have chosen this method because their pretrained model has shown to generate high-quality images with accurate lip sync in a one-shot setting. While we simply utilize real dubber video as driving video, we particularly choose an actor frame with a frontal face and neutral expression, due to the constraint of this method. As a result, we obtain virtual dubber video that has actor identity with dubber expression. Meanwhile, the head pose follows the real dubber video, and the background remains fixed as the reference image. Figure 4.2 shows frames of the real dubber and the corresponding virtual dubber, which is generated from the real dubber on the left.

4.1.2 3D Face Alignment

We next perform 3D face alignment to transfer the actor's head pose to the virtual dubber by utilizing 3D face reconstruction method PRNet [71], following the methodology proposed by Patel et al. Although PRNet may not be the current state-of-the-art in 3D face reconstruction, it still has advantages for our work. The dominant line of face reconstruction methods, such as [72–78],

Figure 4.2: (left) Real dubber, (right) virtual dubber

neglects the mouth interior. In contrast, PRNet provides full-face representation including the inner-mouth, which is critical for our approach to generating realistic faces.

PRNet outputs a 3D mesh of the face with 43,867 vertices from a single-view image along with the canonical and original 68 3D facial keypoints. Similar to Patel et al.'s approach, we apply temporal smoothing to the generated mesh vertex positions, helping to reduce jitters introduced by frame-by-frame reconstruction. Specifically, we employ Savitzky–Golay filter [79] with a window size of 5.

Using the adjusted 3D face mesh and landmarks, we rigid align the canonical virtual dubber object to match the actor's original head pose. Unlike Patel et al., where the dubber's face is rendered on a black background, we render the actor and the virtual dubber face directly onto the original actor frames. This allows us to include the hair and neck, which is important in the later stages of training where we aim to generate not only the face but the hair and neck as well. By doing so, we achieve consistent actor components in pose-aligned virtual dubber frames: background, hair, and neck with differences only in faces. Lastly, despite using the virtual dubber with the actor identity, inconsistencies in the color space may still exist between the actor and virtual dubber videos. To mitigate this, we adopt the color transfer step from Patel et al., which aligns the color space of the dubber video with the actors. Figure 4.3 shows actor frame with the corresponding virtual dubber and final pose-aligned virtual dubber frames.

We observe that PRNet occasionally introduces minor errors in face reconstruction and landmark detection, resulting in head pose misalignment. This misalignment includes jitters, a loose alignment of the reconstructed face with empty spaces around it, and inconsistency in face scales

between consecutive frames. However, we address this issue later in face reenactment to keep the temporal consistency in the final output.

Figure 4.3: 3D face alignment. (left) Original actor frame, (middle) virtual dubber frame, (right) aligned virtual dubber frame

4.2 Full-Head Identity Swapping

4.2.1 Identity Swapping Network

Architecture

We adopt the identity transfer network from Patel et al. as our main architecture, which leverages an autoencoder with a shared encoder and dual decoder. This network combines identity-shared and identity-specific architectures to synthesize realistic actor faces while containing dubber expressions. In particular, input images are passed through 4 main components: encoder, parallel fully-connected layers, GAN block (G-Block), and decoder.

We use Xception [80] for the encoder E. Xception architecture has depthwise separable convolution layers and residual connections. These operations reduce computational complexity while maintaining good performance, simplifying our architecture. Given an RGB color channel image (176x176x3), it outputs the feature map with a size of 6x6x2048. This is followed by the fully connected layer bottleneck for dimension reduction that results in 512-dimensional latent code.

The processed latent code is passed through two parallel fully-connected layers, referred as the *identity extractor* F_i and the *style extractor* F_s. This divided architecture aims to generate distinct embeddings with specialized information:

(1) **Identity Extractor**: Captures local features, preserving high-dimensional spatial information

that retains identity-specific details, resulting in the *identity embedding*.

(2) **Style Extractor**: Captures global features, focusing on low-dimensional abstract representations to ensure global consistency and enable expression synthesis, resulting in the *style embedding*.

While style extractor does not change the dimensionality of the input, we leverage the upsampling technique for identity extractor in order to decompress the encoded feature map. This disentangled design is inspired by StyleGAN [10], which has demonstrated that the separation of the global and local features improves style manipulation and synthesis quality.

The disentangled embeddings are processed by a GAN block(G-Block) g. Incorporating StyleGAN, the style embedding is refined through multiple fully connected layers. Concurrently, Gaussian noise is injected into the identity embedding. g applies Adaptive Instance Normalization (AdaIN) to normalize the identity embedding using the style embedding, followed by convolution layers. This process enables g to produce outputs with finely detailed features, preserving identity while effectively transferring the style.

Lastly, the decoder D learns to generate faces from the shared latent space. We have separate decoders per identity so that they learn subject-specific information. The final output is a 256x256 image with 3 color channels.

Figure 4.4 illustrates the overall architecture, with detailed views of each component provided in Figures 4.5, 4.6, and 4.7. The specification of the Xception architecture is shown in Figure A.1.

Training

In order to feed images to our network, we first detect and crop faces to 512x512 from a full-size frame using a pretrained face detector S3FD [81]. Next, we extract masks for training. Particularly, we utilize two types of mask: segmentation-based mask and landmark-based mask.

For the segmentation mask, we leverage a pretrained face parsing network BiSeNet [82] due to their computational efficiency and accuracy. We mask hair, face, ear, and neck from each original actor, virtual dubber, and pose-aligned dubber videos.

We also mask eyes and mouth to assign different importance in those regions as they are crucial

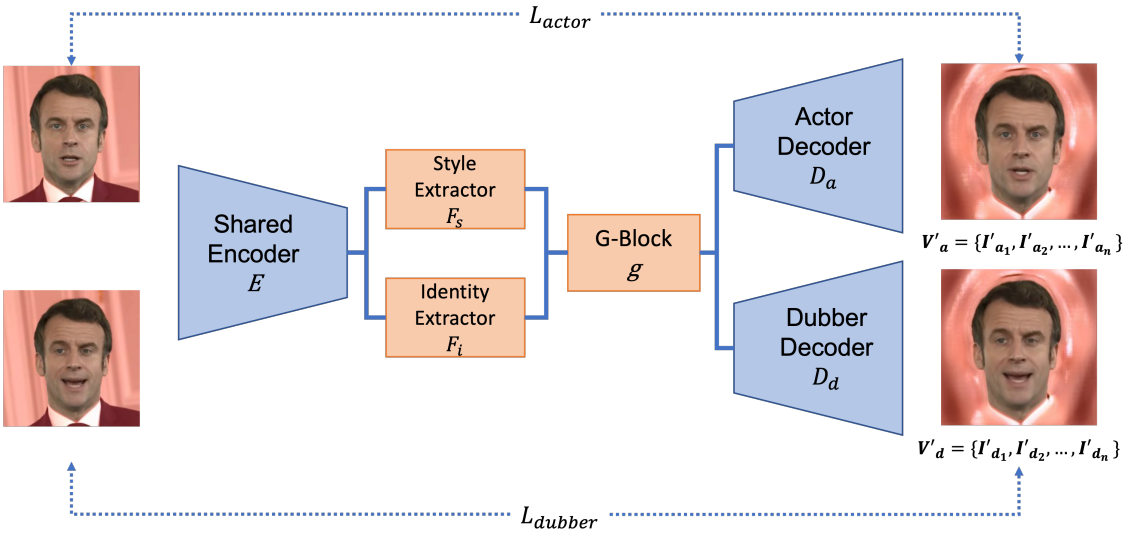

Figure 4.4: Overview of identity swapping network architecture

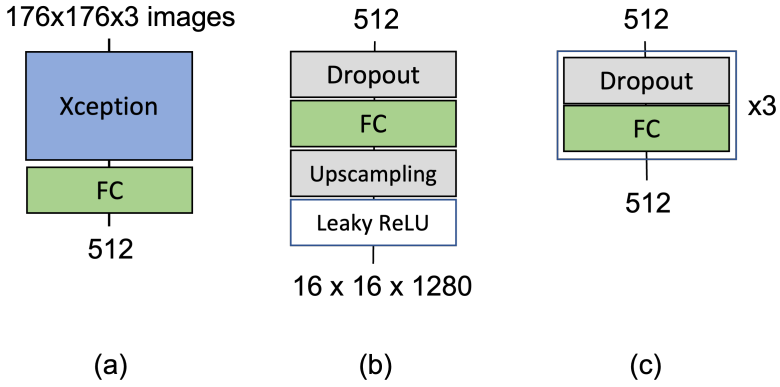

Figure 4.5: Details of encoder and parallel fully connected layers. (a) encoder, (b) identity extractor, (c) style extractor.

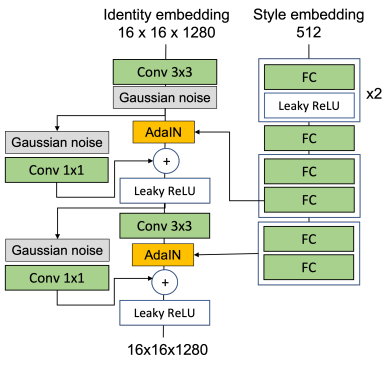

Figure 4.6: Details of G-Block.

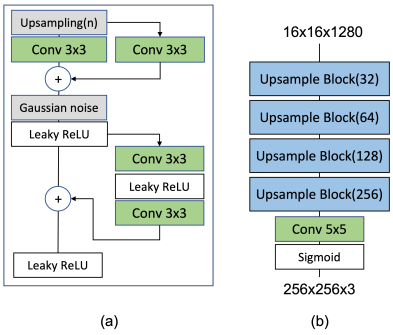

Figure 4.7: Details of decoder (a) upsample block(n), (b) decoder– ”n” refers to the size.

to generate realistic, expressive images. We utilize FAN [83] to detect facial landmarks, then define masks of eyes and mouth using them. Note that we apply a slight blur on the edge of the mask to soften the transitions and exclude any poorly masked edges. Finally, we resize the preprocessed face and masks to 176x176.

During training, we feed the cropped faces concatenated with the generated masks as input. The overall objective is to generate a face, with a focus on the masked region. To accelerate convergence, we leverage pretraining on the CelebA-HQ dataset [84]. Using pretrained weights for initialization, we train the model with not only the original actor and virtual dubber faces but also pose-aligned virtual dubber faces. This training enables: 1) the direct learning of original features such as identity and expression through the original actor and virtual dubber faces, without the need for an intermediate step, and 2) a focus on identity swap without compensating for pose variations, through pose-aligned virtual dubber faces.

Since information from both the actor and dubber is essential, we train separate decoders for each by feeding the corresponding images into the network. For each actor and virtual dubber, the network predicts $I' = G(I)$, aiming to reconstruct the input image. To achieve this, we use the Structural Similarity Index Measure (SSIM) as a reconstruction loss complemented by a Mean Squared Error (MSE) regularization term. The reconstruction and regularization terms are balanced to ensure effective learning.

$$\text{SSIM}(I, I') = \frac{(2\mu_I\mu_{I'} + C_1)(2\sigma_{II'} + C_2)}{(\mu_I^2 + \mu_{I'}^2 + C_1)(\sigma_I^2 + \sigma_{I'}^2 + C_2)} \tag{8}$$

We introduce separate loss terms for specific face regions, defined by predefined masks: M_{face}, M_{eyes}, and M_{mouth}. We formulate the overall face regional loss L_{face} combining pixel-level reconstruction loss L_{recon} and MSE regularization L_{reg} terms. L_{eyes} and L_{mouth} are defined using the same architecture as L_{face} but with a different mask applied to the respective regions (eyes or mouth). This cascade of loss terms effectively balances identity preservation and expressiveness.

$$L_{\text{recon}} = \frac{1}{2}(1 - \text{SSIM}(I \odot M_{\text{face}}, I' \odot M_{\text{face}})) \tag{9}$$

$$L_{\text{reg}} = \frac{1}{N} \sum_{i=1}^{N} \left(M_{\text{face}_i} \odot (I_i - I_i') \right)^2 \tag{10}$$

$$L_{\text{face}} = L_{\text{recon}} + \lambda_{\text{face}} L_{\text{reg}} \tag{11}$$

Since we train shared encoder with split decoders, we separate the loss per decoder. The actor loss L_{actor} incorporates the face, eyes, and mouth loss terms. The virtual dubber loss L_{actor} follows the same structure. Total loss L_{total} is defined as the sum of the actor and virtual dubber losses.

$$L_{\text{actor}} = L_{\text{face}} + \lambda_1 L_{\text{eyes}} + \lambda_2 L_{\text{mouth}} \tag{12}$$

$$L_{\text{total}} = L_{\text{actor}} + L_{\text{dubber}} \tag{13}$$

4.2.2 Full-Head Reenactment

Once the training is complete, the identity of the actor is transferred to each frame of the pose-aligned virtual dubber video by the trained model through the actor decoder, while retaining the expression of the dubber. However, misalignments in 3D face alignment can lead to significant temporal inconsistencies in the output. To address this, we composite the lower part of the pose-aligned virtual dubber faces onto the actor frames and use them as inference input. By compositing, we leverage the natural temporal consistency of the actor frames for the majority of the image, while allowing for the realignment of the pose-aligned virtual dubber frames to correct errors introduced by 3D face alignment.

This approach is inspired by the second identity pass from Patel et al. Patel et al. first perform the identity swap with the pose-aligned dubber and composite the result with the actor. Then they process this composite output through the same network the second time. This method produces more realistic and temporally consistent frames, demonstrating that the identity swapping network can correct minor misalignment in input faces. Similarly, we hypothesize that compositing pose-aligned virtual dubber frames with actor frames will reduce alignment errors and preserve temporal

consistency, while generating realistic results.

To preserve the actor identity while maintaining the mouth expression from the virtual dubber, we composite the lower half of the face, starting from the nose downward, using a landmark-based mask. Unlike the landmark-based mask used in training, we utilize a pretrained 2D landmark detector from SLPT [85] with 98 landmarks. This detector is more robust than PRNet and provides better alignment for the virtual dubber's face region. Although this composite step may still introduce minor misalignments, these can be effectively corrected by the trained network, ensuring the final output is both realistic and temporally consistent. The processing of the inference input is shown in Figure 4.8.

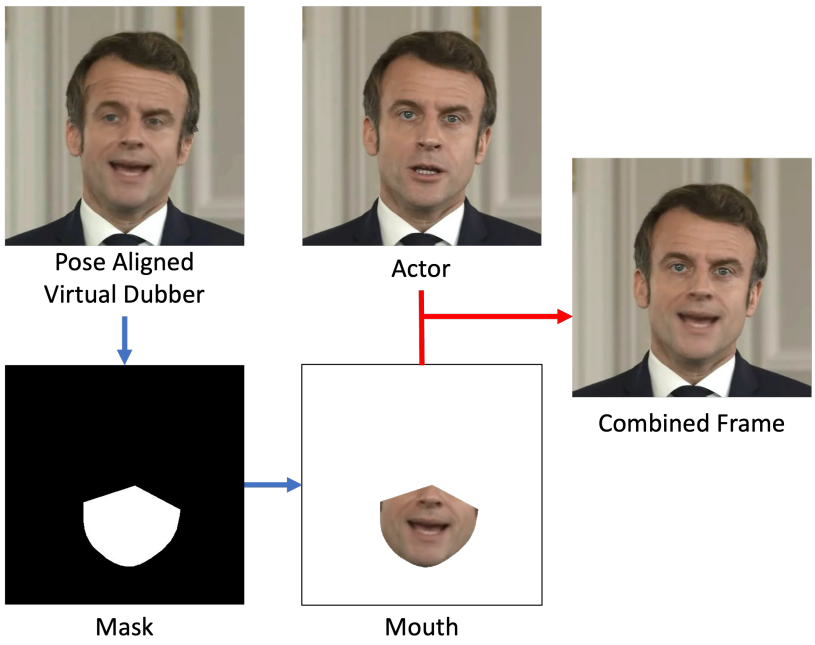

Figure 4.8: Processing the inference input

After the network processes the input, we apply a face segmentation mask to place the generated face back into the same region in the frame. We use the same full-head segmentation as during training. This full-head reenactment eliminates the need for additional compositing using landmarks, which may cause jittering due to small landmark errors. Additionally, by including the hair in reenactment, we can mitigate artifacts or missing details introduced when reenacting only the face, such as details in the forehead. Moreover, reenacting the neck along with the face helps

address the double chin problem, as it can reduce the size of the chin.

4.3 Postprocessing

4.3.1 Identity-specific super resolution

We adapt the fine-tuned super-resolution step from Patel et al. The output of the identity swapping network (256x256) is lower in quality and resolution compared to the original actor video, hence we upscale it to match the original resolution and the high image quality. However, pretrained super-resolution networks do not preserve identity-specific details or the actual quality of the actor video. For example, when the actor has facial wrinkles and folds, pretrained super-resolution often results in overly smooth faces. Moreover, the faces can be excessively enhanced, which gives a "plastic" look.

To address these issues, we fine-tune the pretrained super-resolution method GPEN [86] to be actor-specific. Inspired by Patel et al., we use a training set of both the original actor faces and self-converted faces through the network, using actor decoder. Starting from the input frames with a size of 256x256, we produce outputs at 512x512 with enhanced faces retaining actor-specific details. While we acknowledge that this output size is smaller than the typical original resolution, full HD (1920x1080), we find it sufficient since the face occupies only a portion of the frames. Figure 4.9 illustrates the effect of fine-tuning. Without fine-tuning, the result appears too sharp. Meanwhile, our method closely matches the quality of the original actor, with a minimal blurriness and color difference.

Figure 4.9: Identity-specific super resolution. (left) Original actor frame, (middle) super resolution without fine-tuning, (right) super resolution with fine-tuning

Chapter 5

Results

In this section, we present the evaluation of our experimental results. Our work aims to perform well in practical, real-world scenarios. To this end, we use videos of prominent public figures, including Obama, Macron, Kovind, and Merkel, ranging from 10s to 37s in length. They are paired with professional dubber videos, speaking another language with the translated script.

In addition, we assess our method in more challenging scenarios, such as TV commercials. Unlike the relatively stable head poses and neutral expressions typically seen in public speaking videos, TV commercials exhibit dynamic head movements and expressive facial expressions, and may contain occlusions that cover part of the face. These characteristics significantly increase the complexity of face synthesis. By including such challenging scenarios, we aim to evaluate the robustness and adaptability of handling real-world videos. The details of each video clip can be found in Table 5.1.

	FPS	# Frames
Obama	30	300
Macron	25	818
Kovind	30	1112
Merkel	25	650
Contrave	24	456
Crest	24	198

Table 5.1: Details of video clips used in experiments

Implementation Details

We utilize open source FaceSwap [87] framework for model building and training. First, the identity swapping network is pretrained on the CelebA-HQ dataset [84] for 150k iterations using the Adam optimizer [88] with a learning rate of 0.0001 and batch size of 12. After initializing with the pretrained weights, the person-specific identity swapping model is trained on the original actor and virtual dubber frames for 30k iterations, followed by additional 30k iterations on the original actor and pose-aligned virtual dubber frames. Finally, we fine-tune the model with original actor and virtual dubber frames for 5k iterations, without data augmentation. This training strategy is shown to effectively transfer the fine details and color consistency of the actor faces. We use a batch size of 64 and a learning rate of 5e-5 with the Adam optimizer. Additionally, we set λ_{mouth} and λ_2 to 0.01 and 4, respectively, while the other λ values (λ_{face}, λ_{eyes}, λ_1) are set to 1. The person-specific training takes approximately 2.5 days on four NVIDIA V100 32GB GPUs.

For identity-specific super-resolution, we utilize the pretrained model on 512x512 resolution and training code from GPEN with a batch size of 4 and a learning rate of 2e-2, keeping the remaining hyperparameters as specified in their original paper.

Baselines

We compare our method with state-of-the-art both audio-driven and video-driven lip-syncing approaches. While there are other recent methods, many do not publicly share their code or pre-trained models, which is crucial to replicate their work. Moreover, some methods are limited to cropped videos, making them less suitable for real-world scenarios. Therefore, we focus on methods capable of handling full-frame videos, aligning more closely with our target application.

Our comparison includes Wav2Lip [3] and TalkLip [41] for audio-driven methods, and Patel et al. [1] and LivePortrait [4] as video-driven methods. For Wav2Lip, TalkLip and LivePortrait, we used their official implementations. Specifically, as LivePortrait provides both image-based and video-based lip-sync video generation, we utilized the video-based lip-syncing script. In addition, note that TalkLip is strictly limited to 25fps videos, hence we convert the frame rate from original video to 25fps and reconvert it to original after generation.

Benchmarks

We evaluate our method using two criteria: generation visual quality and lip-sync quality, adopting the most commonly used metrics. To measure generation visual quality, we use Peak Signal-to-Noise Ratio (PSNR), Fréchet Inception Distance (FID) [89], and Learned Perceptual Image Patch Similarity (LPIPS) [90], with the original actor video as the reference. For lip-sync quality, we compute Landmark Distance (LMD) [91] within the mouth region, using the dubber's landmarks as the ground truth. All metrics are computed frame by frame and averaged.

5.1 Quantitative Evaluation

	Obama			Macron			Kovind			Merkel		
Method	PSNR↑	FID↓	LPIPS↓	PSNR↑	FID↓	LPIPS↓	PSNR↑	FID↓	LPIPS↓	PSNR↑	FID↓	LPIPS↓
Wav2Lip	35.07	8.53	0.10	38.41	1.71	0.06	36.79	4.34	0.07	38.61	6.73	0.07
TalLlip	32.55	13.77	0.15	33.98	7.81	0.15	34.04	5.40	0.10	32.72	17.51	0.19
Patel et al.	**38.92**	**2.74**	0.06	**48.89**	**0.23**	0.01	**54.19**	**0.28**	**<0.01**	**49.37**	1.66	**<0.01**
LivePortrait	31.72	9.95	0.23	39.54	3.31	0.04	38.64	7.19	0.04	39.93	5.16	0.05
Ours	37.56	3.71	**0.06**	47.56	0.29	**0.01**	46.44	1.15	<0.01	48.63	**0.96**	0.01

Table 5.2: Visual quality evaluation. Bold indicates the best, and underline indicates the second best.

Method	Obama	Macron	Kovind	Merkel
Wav2Lip	1.55	0.79	1.04	1.49
TalkLip	1.44	0.80	1.00	1.68
Patel et al.	1.42	0.82	1.07	1.30
LivePortrait	1.78	0.80	**0.79**	2.34
Ours	**1.21**	**0.54**	0.96	**1.25**

Table 5.3: Lip syncronization evaluation (LMD). The lower the value, the closer to the dubber mouth expression (GT).

Quantitative evaluations are presented in Tables 5.2 and 5.3. Table 5.2 demonstrates that Patel et al.'s method achieves superior results across most metrics for the tested video sequences. Especially in the Kovind sequence, their method significantly outperforms others in PSNR and FID metrics, indicating higher visual quality.

We attribute this difference to the design of their approach, as it focuses on compositing the mouth region while leaving the rest of the video untouched. By restricting reenactment to a smaller area, their method only makes changes inside the mouth region, resulting in higher visual quality

metrics.

In contrast, our method reenacts a larger portion of the video. This introduces a higher possibility of not only actually degrading the visual quality but also slight numerical divergence in metrics due to broader modifications. Despite the added "risk" of modifying a broader region, our method achieves competitive scores. For example, we achieve the second-best performance overall and present the best LPIPS score for the Obama and Macron sequences. This indicates that our results are perceptually closest to the original videos, demonstrating the robustness of our approach in preserving actor specific details and overall consistency. Similarly, although PSNR score for the Merkel sequence is behind the best, we achieve a lower FID score and a close second-best LPIPS score with a small difference. While PSNR might suggest more noise, both FID and LPIPS show that our method excels in generating realistic and perceptually accurate output.

Table 5.3 evaluates lip-syncing accuracy using the Landmark Distance (LMD) metric. Our method consistently outperforms other approaches across most test cases, demonstrating superior alignment between the generated lip movements and the dubber's ground-truth expressions. However, in the Kovind sequence, LivePortrait achieves the best LMD score. While LivePortrait presents the highest lip synchronization accuracy with the dubber, its generated head pose occasionally differs from the original actor's pose. This observation validates the importance of our method, which balances accurate lip synchronization with correct head pose alignment, a critical requirement for visual dubbing tasks.

While these quantitative metrics provide valuable insights, we emphasize that qualitative evaluations are critical for real-world applications, as they better capture subjective visual quality and lip synchronization. We highly recommend referring to supplementary videos to assess temporal consistency.

5.2 Qualitative Evaluation

Qualitative comparisons further highlight the strengths and limitations of our method. For evaluation, we process full-frame videos and crop the results to the face region for direct comparison. Overall, although Wav2Lip achieves quite accurate lip-sync, the visual quality is lower. Similarly,

TalkLip generates blurry results and, more critically, shows a noticeable lag compared to the original actor video. LivePortrait produces high-quality output; however, the head poses are not consistently aligned with the actor's, rather following the dubber's sometimes, which is not desired. Moreover, it occasionally generates uncanny-looking eyes. Patel et al.'s method delivers comparable results, but they are not as expressive as ours and sometimes introduce artifacts.

Expressiveness

Our method demonstrates superior expressiveness in reconstructing dubber expression, compared to other methods. For instance, as shown in Figure 5.1 (left), our method effectively recreates challenging expressions like mouth puckering, which other methods fail to capture adequately. Moreover, our method effectively captures the expression even under challenging head poses. In Figure 5.1 (right), where the dubber's head is tilted downward, our approach accurately extracts and generates the expressions with minimal loss of detail. TalkLip also captures the puckering expression, but the right corner of the lips is not moving, which is unrealistic.

Handling Jaw Artifacts

Since our method reenacts the full head, including the neck and jaw, artifacts such as the double-chin effect can be addressed, as illustrated in Figure 5.2. In comparison, Patel et al. preserves the original jaw size and only composite the mouth. Unlike their method, our approach generates and reenacts full head, enabling subtle adjustments to the jawline.

Although our results are slightly more blurry than those of Patel et al., which achieves the best sharpness among tested methods, we argue that this trade-off results in a more balanced output that achieves both lip-sync accuracy and realistic facial reenactment.

5.3 Evaluation on Commercial Videos

We further evaluate our method on commercial videos which typically contains dynamic head poses and often occlusions. In these scenarios, qualitative results are more relevant than numerical metrics, as they better illustrate the robustness of our approach in real-world conditions.

Figure 5.1: Qualitative results on Obama and Macron.

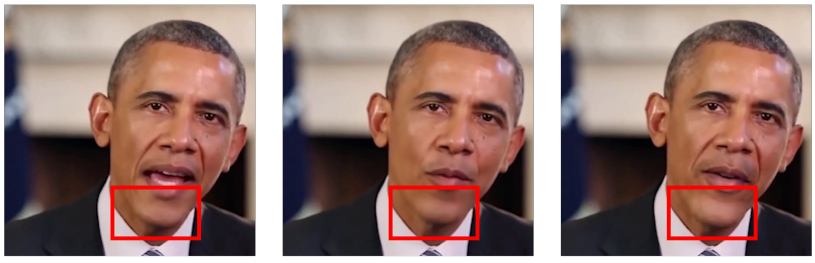

Figure 5.2: Handling jaw artifacts. (left) Original, (middle) ours, (right) Patel et al. Even trained from the same dubber video, due to the difference in training schema, the generated expression may not completely match.

As shown in Figure 5.3 (first row), our method effectively handles dynamic head poses. For example, in a tilted pose, our method successfully reenacts the new expression without introducing artifacts, maintaining visual quality.

Furthermore, in the second row of Figure 5.3 demonstrates the ability of our method to handle occlusions, such as objects partially covering the mouth. We can conclude that our approach generates realistic frames without introducing visible artifacts, even under challenging conditions.

Additionally, the identity-specific super-resolution allows high-quality output to closely match the original, ensuring production quality with only minimal blurriness and color change. These capabilities make our method practical in dynamic industry-level video content.

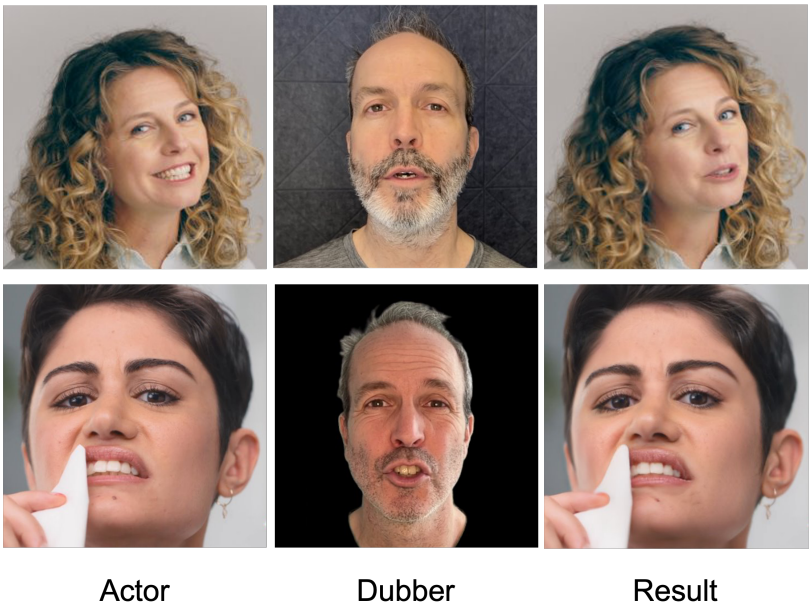

Figure 5.3: Results on commercial videos

5.4 Limitations

Despite the advantages, our method has limitations. The most notable issue is the slightly blurred output with slightly shifted color space. As we perform reenactment in entire head and neck, it is quite noticeable. This could be addressed by improving the super-resolution module within the pipeline.

In addition, the lack of explicit postprocessing for temporal consistency may lead to minor inconsistencies. For instance, temporal inconsistencies are visible in the Kovind video sequence,

which are small scale differences in the face (i.e. face becomes slightly bigger/smaller in consecutive frames). Although we indirectly maintain the temporal consistency by using actor and pose-aligned virtual dubber combined frames in inference, the current pipeline lacks a mechanism to correct the inconsistent frames generated by the network. It could be dealt with manual corrections or incorporating a temporal consistency module.

Another limitation is our dependence on a talking head generation method. If the underlying talking head model fails, the performance of our method is significantly affected. However, this can be easily addressed by using other talking head generation methods.

Chapter 6

Ablation Study

To further evaluate the necessity of incorporating the virtual dubber, we performed an ablation study using a commercial video sequence (Contrave). In this experiment, we removed the virtual dubber in an intermediate step and only used original dubber video. While the output retains actor identity in a comparable way, the result without the virtual dubber shows noticeably degraded lip synchronization. For example, Figure 6.1 illustrates a frame where teeth are incorrectly generated, while the ground truth is a closed mouth. In addition, the result is more blurry. This highlights the importance of the virtual dubber in capturing facial expressions and producing high-quality output.

Figure 6.1: Ablation study. (left) dubber, (middle) result without virtual dubber, (right) result with virtual dubber (ours). Dubber shows ground truth expression.

Chapter 7

Conclusion

In this thesis, we have introduced a visual dubbing pipeline that leverages a virtual dubber, a talking head that maintains the actor identity while mimicking the dubber's expressions, as an intermediate representation that helps significantly enhancing lip synchronization and visual quality. Our method performs identity swapping and reenactment in an extended head region including hair and neck, ensuring realistic, temporally consistent output without requiring additional postprocessing steps. Furthermore, our person-specific super-resolution allows for high-quality outputs that retain the quality of the original actor video. Overall, our method is able to generate videos that are visually fairly close to the original actor with only the expressions being synthesized.

Our quantitative and qualitative comparisons against competitive methods demonstrate that our method achieves a superior balance between visual quality and lip synchronization, outperforming other tested approaches.

Finally, our experiments on short TV commercial videos confirm the robustness of the pipeline under challenging real-world conditions, as well as the efficiency even with limited data.

Appendix A

Xception Network

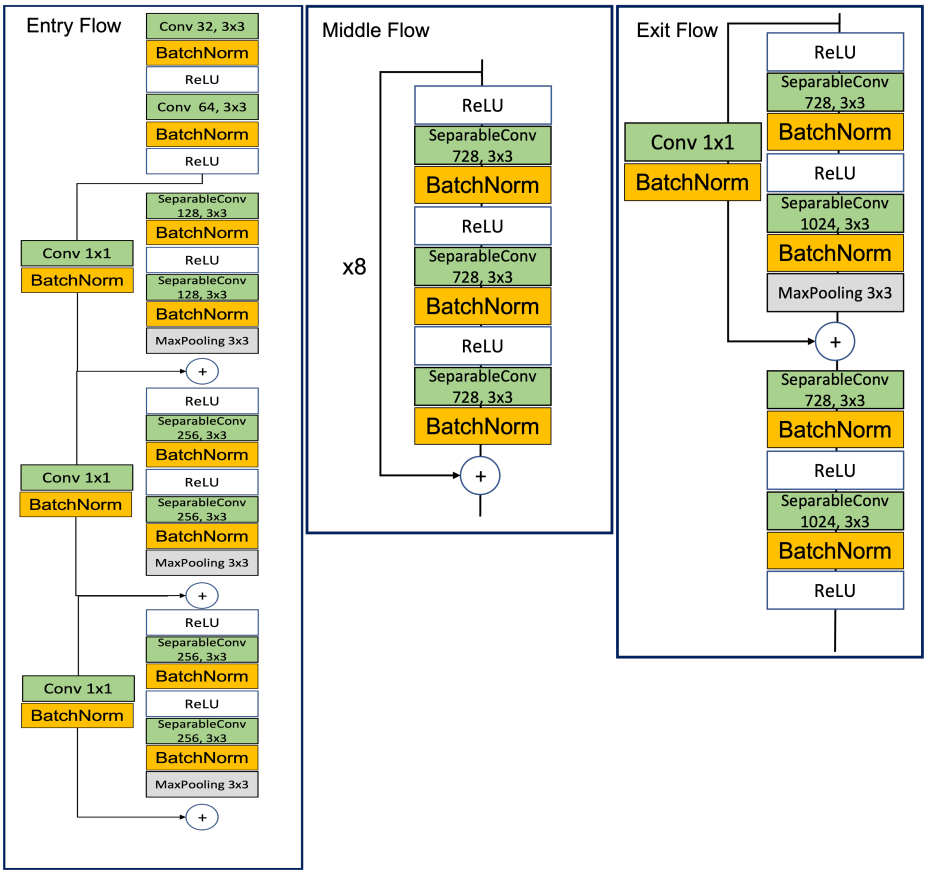

Figure A.1: Xception [80] architecture

Bibliography

[1] D. Patel, H. Zouaghi, S. Mudur, E. Paquette, S. Laforest, M. Rouillard, and T. Popa, "Visual dubbing pipeline with localized lip-sync and two-pass identity transfer," *Comput. Graph.*, vol. 110, p. 19–27, Feb. 2023.

[2] W. Zhang, X. Cun, X. Wang, Y. Zhang, X. Shen, Y. Guo, Y. Shan, and F. Wang, "Sadtalker: Learning realistic 3d motion coefficients for stylized audio-driven single image talking face animation," 2023.

[3] K. R. 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 *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, (New York, NY, USA), p. 484–492, Association for Computing Machinery, 2020.

[4] J. Guo, D. Zhang, X. Liu, Z. Zhong, Y. Zhang, P. Wan, and D. Zhang, "Liveportrait: Efficient portrait animation with stitching and retargeting control," 2024.

[5] C. Bregler, M. Covell, and M. Slaney, "Video rewrite: driving visual speech with audio," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, (USA), p. 353–360, ACM Press/Addison-Wesley Publishing Co., 1997.

[6] H. Yehia, P. Rubin, and E. Vatikiotis-Bateson, "Quantitative association of vocal-tract and facial behavior," *Speech Commun.*, vol. 26, p. 23–43, Oct. 1998.

[7] T. Ezzat, G. Geiger, and T. Poggio, "Trainable videorealistic speech animation," *ACM Trans. Graph.*, vol. 21, p. 388–398, July 2002.

[8] D. P. Kingma, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.

[10] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," 2019.

[11] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," 2020.

[12] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," 2021.

[13] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[14] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[15] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation," *Journal of Machine Learning Research*, vol. 23, no. 47, pp. 1–33, 2022.

[16] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," 2020.

[17] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, (USA), p. 187–194, ACM Press/Addison-Wesley Publishing Co., 1999.

[18] P. Kellnhofer, L. C. Jebe, A. Jones, R. Spicer, K. Pulli, and G. Wetzstein, "Neural lumigraph rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4287–4297, 2021.

[19] S. Yao, R. Zhong, Y. Yan, G. Zhai, and X. Yang, "Dfa-nerf: Personalized talking head generation via disentangled face attributes neural rendering," *arXiv preprint arXiv:2201.00791*, 2022.

[20] T. D. Kulkarni, W. Whitney, P. Kohli, and J. B. Tenenbaum, "Deep convolutional inverse graphics network," 2015.

[21] Z. Shu, M. Sahasrabudhe, A. Guler, D. Samaras, N. Paragios, and I. Kokkinos, "Deforming autoencoders: Unsupervised disentangling of shape and appearance," 2018.

[22] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," 2020.

[23] L. Ma and Z. Deng, "Real-time hierarchical facial performance capture," in *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '19, (New York, NY, USA), Association for Computing Machinery, 2019.

[24] W. Wu, Y. Zhang, C. Li, C. Qian, and C. C. Loy, "Reenactgan: Learning to reenact faces via boundary transfer," 2018.

[25] Y. Zhang, S. Zhang, Y. He, C. Li, C. C. Loy, and Z. Liu, "One-shot face reenactment," 2019.

[26] Y. Nirkin, Y. Keller, and T. Hassner, "Fsgan: Subject agnostic face swapping and reenactment," 2019.

[27] J. Zhang, X. Zeng, M. Wang, Y. Pan, L. Liu, Y. Liu, Y. Ding, and C. Fan, "Freenet: Multi-identity face reenactment," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5325–5334, 2020.

[28] S. Ha, M. Kersner, B. Kim, S. Seo, and D. Kim, "Marionette: Few-shot face reenactment preserving identity of unseen targets," 2019.

[29] S. Tripathy, J. Kannala, and E. Rahtu, "Icface: Interpretable and controllable face reenactment using gans," 2020.

[30] P. Ekman and W. V. Friesen, "Facial action coding system," *Environmental Psychology & Nonverbal Behavior*, 1978.

[31] G. Yao, Y. Yuan, T. Shao, S. Li, S. Liu, Y. Liu, M. Wang, and K. Zhou, "One-shot face reenactment using appearance adaptive normalization," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 3172–3180, 2021.

[32] G. Yao, Y. Yuan, T. Shao, and K. Zhou, "Mesh guided one-shot face reenactment using graph convolutional networks," in *Proceedings of the 28th ACM international conference on multimedia*, pp. 1773–1781, 2020.

[33] K. Yang, K. Chen, D. Guo, S.-H. Zhang, Y.-C. Guo, and W. Zhang, "Face2face ρ: Real-time high-resolution one-shot face reenactment," in *European conference on computer vision*, pp. 55–71, Springer, 2022.

[34] S. Bounareli, C. Tzelepis, V. Argyriou, I. Patras, and G. Tzimiropoulos, "Stylemask: Disentangling the style space of stylegan2 for neural face reenactment," in *2023 IEEE 17th international conference on automatic face and gesture recognition (FG)*, pp. 1–8, IEEE, 2023.

[35] S. Bounareli, V. Argyriou, and G. Tzimiropoulos, "Finding directions in gan's latent space for neural face reenactment," *arXiv preprint arXiv:2202.00046*, 2022.

[36] S. Bounareli, C. Tzelepis, V. Argyriou, I. Patras, and G. Tzimiropoulos, "Hyperreenact: one-shot reenactment via jointly learning to refine and retarget faces," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7149–7159, 2023.

[37] P. K R, R. Mukhopadhyay, J. Philip, A. Jha, V. Namboodiri, and C. V. Jawahar, "Towards automatic face-to-face translation," in *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, (New York, NY, USA), pp. 1428–1436, ACM, 2019.

[38] B. Liang, Y. Pan, Z. Guo, H. Zhou, Z. Hong, X. Han, J. Han, J. Liu, E. Ding, and J. Wang, "Expressive talking head generation with granular audio-visual control," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3377–3386, 2022.

[39] F.-T. Hong, L. Zhang, L. Shen, and D. Xu, "Depth-aware generative adversarial network for talking head video generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3397–3406, June 2022.

[40] Y. Ma, S. Wang, Z. Hu, C. Fan, T. Lv, Y. Ding, Z. Deng, and X. Yu, "Styletalk: One-shot talking head generation with controllable speaking styles," 2023.

[41] J. Wang, X. Qian, M. Zhang, R. T. Tan, and H. Li, "Seeing what you said: Talking face generation guided by a lip reading expert," 2023.

[42] H. Wei, Z. Yang, and Z. Wang, "Aniportrait: Audio-driven synthesis of photorealistic portrait animation," 2024.

[43] K. Cheng, X. Cun, Y. Zhang, M. Xia, F. Yin, M. Zhu, X. Wang, J. Wang, and N. Wang, "Videoretalking: Audio-based lip synchronization for talking head video editing in the wild," 2022.

[44] M. Stypułkowski, K. Vougioukas, S. He, M. Zieba, S. Petridis, and M. Pantic, "Diffused heads: Diffusion models beat gans on talking-face generation," 2023.

[45] T. Liu, C. Du, S. Fan, F. Chen, and K. Yu, "Diffdub: Person-generic visual dubbing using inpainting renderer with diffusion auto-encoder," 2024.

[46] Y. Ma, S. Zhang, J. Wang, X. Wang, Y. Zhang, and Z. Deng, "Dreamtalk: When emotional talking head generation meets diffusion probabilistic models," 2024.

[47] K. Preechakul, N. Chatthee, S. Wizadwongsa, and S. Suwajanakorn, "Diffusion autoencoders: Toward a meaningful and decodable representation," 2022.

[48] C. Du, Q. Chen, T. He, X. Tan, X. Chen, K. Yu, S. Zhao, and J. Bian, "Dae-talker: High fidelity speech-driven talking face generation with diffusion autoencoder," in *Proceedings of the 31st ACM International Conference on Multimedia*, MM '23, p. 4281–4289, ACM, Oct. 2023.

[49] W. Li, L. Zhang, D. Wang, B. Zhao, Z. Wang, M. Chen, B. Zhang, Z. Wang, L. Bo, and X. Li, "One-shot high-fidelity talking-head synthesis with deformable neural radiance field," 2023.

[50] Z. Ye, T. Zhong, Y. Ren, J. Yang, W. Li, J. Huang, Z. Jiang, J. He, R. Huang, J. Liu, C. Zhang, X. Yin, Z. Ma, and Z. Zhao, "Real3d-portrait: One-shot realistic 3d talking portrait synthesis," 2024.

[51] Z. Ye, Z. Jiang, Y. Ren, J. Liu, J. He, and Z. Zhao, "Geneface: Generalized and high-fidelity audio-driven 3d talking face synthesis," 2023.

[52] Z. Peng, W. Hu, Y. Shi, X. Zhu, X. Zhang, H. Zhao, J. He, H. Liu, and Z. Fan, "Synctalk: The devil is in the synchronization for talking head synthesis," 2024.

[53] B. Egger, W. A. P. Smith, A. Tewari, S. Wuhrer, M. Zollhoefer, T. Beeler, F. Bernard, T. Bolkart, A. Kortylewski, S. Romdhani, C. Theobalt, V. Blanz, and T. Vetter, "3d morphable face models – past, present and future," 2020.

[54] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li, "Makelttalk: speaker-aware talking-head animation," *ACM Transactions on Graphics*, vol. 39, p. 1–15, Nov. 2020.

[55] T. Xie, L. Liao, C. Bi, B. Tang, X. Yin, J. Yang, M. Wang, J. Yao, Y. Zhang, and Z. Ma, "Towards realistic visual dubbing with heterogeneous sources," in *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, p. 1739–1747, ACM, Oct. 2021.

[56] Y. Lu, J. Chai, and X. Cao, "Live speech portraits: real-time photorealistic talking-head animation," *ACM Trans. Graph.*, vol. 40, Dec. 2021.

[57] B. Zhang, C. Qi, P. Zhang, B. Zhang, H. Wu, D. Chen, Q. Chen, Y. Wang, and F. Wen, "Metaportrait: Identity-preserving talking head generation with fast personalized adaptation," 2023.

[58] F.-T. Hong and D. Xu, "Implicit identity representation conditioned memory compensation network for talking head video generation," 2023.

[59] T.-C. Wang, A. Mallya, and M.-Y. Liu, "One-shot free-view neural talking-head synthesis for video conferencing," 2021.

[60] A. Richard, M. Zollhoefer, Y. Wen, F. de la Torre, and Y. Sheikh, "Meshtalk: 3d face animation from speech using cross-modality disentanglement," 2022.

[61] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Synthesizing obama: learning lip sync from audio," *ACM Trans. Graph.*, vol. 36, July 2017.

[62] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural voice puppetry: Audio-driven facial reenactment," 2020.

[63] A. Lahiri, V. Kwatra, C. Frueh, J. Lewis, and C. Bregler, "Lipsync3d: Data-efficient learning of personalized 3d talking faces from video using pose and lighting normalization," 2021.

[64] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," 2020.

[65] S. J. Park, M. Kim, J. Hong, J. Choi, and Y. M. Ro, "Synctalkface: Talking face generation with precise lip-syncing via audio-lip memory," 2022.

[66] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies, M. Niessner, P. Pérez, C. Richardt, M. Zollhöfer, and C. Theobalt, "Deep video portraits," *ACM Trans. Graph.*, vol. 37, July 2018.

[67] H. Kim, M. Elgharib, M. Zollhöfer, H.-P. Seidel, T. Beeler, C. Richardt, and C. Theobalt, "Neural style-preserving visual dubbing," *ACM Transactions on Graphics*, vol. 38, p. 1–13, Nov. 2019.

[68] S. Shen, W. Li, Z. Zhu, Y. Duan, J. Zhou, and J. Lu, "Learning dynamic facial radiance fields for few-shot talking head synthesis," 2022.

[69] J. Saunders and V. Namboodiri, "Dubbing for everyone: Data-efficient visual dubbing using neural rendering priors," *arxiv*, 2024.

[70] J. Guan, Z. Zhang, H. Zhou, T. HU, K. Wang, D. He, H. Feng, J. Liu, E. Ding, Z. Liu, and J. Wang, "Stylesync: High-fidelity generalized and personalized lip sync in style-based generator," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[71] Y. Wang and J. M. Solomon, "Prnet: Self-supervised learning for partial-to-partial registration," 2019.

[72] J. Xie, R. Girshick, and A. Farhadi, "Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks," 2016.

[73] J. Guo, X. Zhu, Y. Yang, F. Yang, Z. Lei, and S. Z. Li, "Towards fast, accurate and stable 3d dense face alignment," 2021.

[74] V. Gorade, S. Mittal, D. Jha, and U. Bagci, "Synergynet: Bridging the gap between discrete and continuous representations for precise medical image segmentation," 2023.

[75] Y. Feng, H. Feng, M. J. Black, and T. Bolkart, "Learning an animatable detailed 3d face model from in-the-wild images," 2021.

[76] J. Shang, T. Shen, S. Li, L. Zhou, M. Zhen, T. Fang, and L. Quan, "Self-supervised monocular 3d face reconstruction by occlusion-aware multi-view geometry consistency," 2020.

[77] B. Lei, J. Ren, M. Feng, M. Cui, and X. Xie, "A hierarchical representation network for accurate and detailed face reconstruction from in-the-wild images," 2023.

[78] Z. Wang, X. Zhu, T. Zhang, B. Wang, and Z. Lei, "3d face reconstruction with the geometric guidance of facial part segmentation," 2024.

[79] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, pp. 1627–1639, Jan. 1964.

[80] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2017.

[81] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "S^3fd: Single shot scale-invariant face detector," 2017.

[82] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," 2018.

[83] A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2d amp; 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks)," in *2017 IEEE International Conference on Computer Vision (ICCV)*, p. 1021–1030, IEEE, Oct. 2017.

[84] T. Karras, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[85] J. Xia, W. qu, W. Huang, J. Zhang, X. Wang, and M. Xu, "Sparse local patch transformer for robust face alignment and landmarks inherent relation learning," 2022.

[86] X. X. Tao Yang, Peiran Ren and L. Zhang, "Gan prior embedded network for blind face restoration in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[87] Deepfakes, "FaceSwap." https://faceswap.dev. [Accessed: 2024-12-04].

[88] D. P. Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[89] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[90] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

[91] L. Chen, Z. Li, R. K. Maddox, Z. Duan, and C. Xu, "Lip movements generation at a glance," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 520–535, 2018.