Personalized Visual Dubbing through Virtual Dubber and Full Head Reenactment

Bobae Jeon[1], Eric Paquette[2], Sudhir Mudur[1], Tiberiu Popa[1]

[1]Concordia University, Montreal, Canada
[2]École de technologie supérieure, Montreal, Canada

Abstract

Visual dubbing aims to modify facial expressions to "lip-sync" a new audio track. While person-generic talking head generation methods achieve expressive lip synchronization across arbitrary identities, they usually lack person-specific details and fail to generate high-quality results. Conversely, person-specific methods require extensive training. Our method combines the strengths of both methods by incorporating a virtual dubber, a person-generic talking head, as an intermediate representation. We then employ an autoencoder-based person-specific identity swapping network to transfer the actor identity, enabling full-head reenactment that includes hair, face, ears, and neck. This eliminates artifacts while ensuring temporal consistency. Our quantitative and qualitative evaluation demonstrate that our method achieves a superior balance between lip-sync accuracy and realistic facial reenactment.

CCS Concepts

• ***Computing methodologies*** → ***Image manipulation; Animation;***

1. Introduction

Visual dubbing aims to synchronize the facial expression of an actor with arbitrary audio, while maintaining 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 work on arbitrary identities by training with diverse datasets, achieving expressive mouth movements. However, they often fail to generate high visual quality videos with person-specific details, leading to uncanny facial features. Furthermore, many rely on a single-image as input, synthesizing only the face or torso, inherently resulting in static backgrounds and unnatural body parts [ZCW*23, YZR*24, WYW24]. This poses a challenge for real-world visual dubbing applications, where maintaining all parts of the original video, except for the mouth expressions, is crucial.

In contrast, person-specific methods train customized models tailored to specific individuals. They have shown realistic identity preservation and good lip-sync quality, but often require large amounts of person-specific data for training [KGT*18, KEZ*19, TET*20]. This presents a significant drawback when applied to many real-world scenarios. For example, TV ads are typically less than 30 seconds long.

To address these limitations, we propose a novel pipeline that combines the strengths of person-generic and person-specific methods. Our method introduces a virtual dubber, defined as a talking head generated over a static background from a person-generic method driven by a dubber's facial expressions. This serves as an intermediate representation, capturing 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 background, and occasional uncanny facial features, it allows reducing the need for the several minutes to hours of footage typically required for person-specific model training.

Building on the work of Patel et al. [PZM*23], we use an autoencoder-based identity swapping network to transfer the actor's identity to the virtual dubber. The virtual dubber provides a closer representation of the desired output as it retains some aspects of the actor identity, simplifying the identity swap and making the process more efficient, unlike Patel et al., which require a second identity transfer pass to address the mismatch between the actor's mouth style and the dubber expressions in their initial output. Furthermore, we perform identity swap and reenactment in a full head, including the face, hair, and neck, not only allowing better preservation in identity and visual quality, but also enabling adjusting the size of the face parts, helping to avoid undesirable artifacts.

In addition, [PZM*23] 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 with a lower jaw, but the synthesized face has a closed mouth, this mismatch results in an unnatural appearance, resembling a double chin. Our method reenacts the full head, eliminating the need for mouth-region compositing, which

DOI: 10.2312/egs.20251034

resolves the double chin effect and improves temporal consistency. In summary, our contributions are: **(1) Introduction of a virtual dubber**: An expressive intermediate representation that improves efficiency in full-head identity swap. **(2) Full-head identity swap and reenactment**: Effectively preserves identity and visual quality, while ensuring temporal consistency and eliminating artifacts.

2. Method

Our pipeline consists of three stages: preprocessing, full-head identity swapping, and postprocessing (see Figure 1). In the preprocessing stage, we first create a "virtual dubber" using a static talking head to obtain the dubber expression with the actor identity. Next, we transfer head pose and background from the actor to the virtual dubber. This is followed by a full-head identity swapping. Our identity swapping network transfers the actor identity to the virtual dubber, synthesizing frames with the actor identity and the dubber expression. Although the virtual dubber retains most aspects of the actor identity, it does not fully preserve it due to occasional unnatural faces. Therefore, we need identity swapping to further enhance identity preservation. Finally, in the postprocessing stage, we enhance the resolution while preserving identity-specific details.

2.1. Preprocessing

Virtual dubber: For the generation of the virtual dubber, we use LivePortrait [GZL*24] which synthesizes a talking head given a driving video (dubber) and a reference image (actor). LivePortrait requires a neutral front-facing image, which we select manually. As a result, we obtain a virtual dubber video that has the actor identity with the dubber expression. Meanwhile, the background remains fixed as the reference image. The head pose follows the real dubber video, which we adjust in the 3D face alignment.

3D face alignment: We next transfer the actor's head pose to the virtual dubber by utilizing the 3D face reconstruction method PRNet [FWS*18], following 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. While the dominant line of face reconstruction methods neglects the mouth interior, PRNet provides full-face including the inner-mouth, which is critical for our method. We apply temporal smoothing to the generated mesh vertex positions, helping to reduce jitters introduced by the frame-by-frame reconstruction. Using the adjusted 3D face mesh and landmarks, we rigid align the mesh of the virtual dubber to match the actor's head pose. Specifically, we render the aligned virtual dubber face directly onto the original actor frames to include the hair and neck, as these are needed later in the full-head identity swapping. We observe that PRNet occasionally introduces minor errors or misalignments in face reconstruction and landmark detection. This includes jitters, a loose alignment of the reconstructed face, and inconsistency in face scale between consecutive frames. However, we do not perform manual corrections since these issues are addressed later during the reenactment.

2.2. Full-Head Identity Swapping

Identity Swapping Network: Our network leverages an autoencoder with a shared encoder and dual decoders. Particularly, input images are passed through four main components: encoder, parallel fully-connected layers, GAN block (G-Block), and decoder (supplementary material, Figure 3).

We use Xception [Cho17] for the encoder E. Given an RGB color channel image $(176 \times 176 \times 3)$, it outputs a feature map with a size of $6 \times 6 \times 2048$, followed by the fully connected layer bottleneck that results in a 512-dimensional latent code. The processed latent code is passed through two parallel fully-connected layers: identity extractor F_i and the style extractor F_s. This divided architecture is inspired by StyleGAN [KLA19], aiming to generate distinct embeddings with specialized information: the identity extractor preserves identity-specific details, resulting in the identity embedding. The style extractor focuses on abstract representations to enable expression synthesis, resulting in the style embedding.

The disentangled embeddings are processed by the G-Block which uses Adaptive Instance Normalization (AdaIN) to normalize the identity embedding using the style embedding, followed by convolution layers. This process enables the G-Block to produce outputs with finely detailed features, preserving identity while effectively transferring the style. Lastly, the separated decoders D generate heads from the shared latent space, learning subject-specific information.

Training: We feed the cropped heads (176×176) concatenated with grey-scale soft masks (176×176) we automatically generate by segmentation. The masks consist of: a head mask M_{head} covering hair, face, ears, and neck, and masks focused on eyes (M_{eyes}) and mouth (M_{mouth}). Our objective is to generate a head, with a focus on the masked region. To do so, we train the encoder with separate decoders to reconstruct the input image, one using the original actor frames and the other using the virtual dubber frames. For each actor and virtual dubber image I, the identity swapping network reconstructs an image I'. We use SSIM as a reconstruction loss complemented by an MSE regularization term:

$$\begin{aligned} L_{\text{head}} = &\frac{1}{2}(1 - \text{SSIM}(I \odot M_{\text{head}}, I' \odot M_{\text{head}})) \\ &+ \lambda_{\text{head}} \text{MSE}(I \odot M_{\text{head}}, I' \odot M_{\text{head}}), \end{aligned} \tag{1}$$

where $\odot$ indicates element-wise multiplication. Losses L_{eyes} and L_{mouth} are defined using the same architecture as L_{head} but with a different mask and λ applied to the respective regions (eyes or mouth). Similarly, we have a loss per decoder:

$$\begin{aligned} L_{\text{actor}} &= L_{\text{head}} + \lambda_1 L_{\text{eyes}} + \lambda_2 L_{\text{mouth}} \\ L_{\text{dubber}} &= L_{\text{head}} + \lambda_1 L_{\text{eyes}} + \lambda_2 L_{\text{mouth}}. \end{aligned} \tag{2}$$

Total loss L_{total} is defined as: $L_{\text{total}} = L_{\text{actor}} + L_{\text{dubber}}$.

Full-head reenactment: After training, 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 virtual dubber. However, misalignments of the face can lead to significant temporal inconsistencies in the output. To address this, we composite the lower part of the pose-aligned virtual dubber face onto the actor frame before feeding it to the network (see . Although some misalignments may persist in the composited frame, these are corrected by the identity swapping network.

After the network, we apply another compositing step, pasting

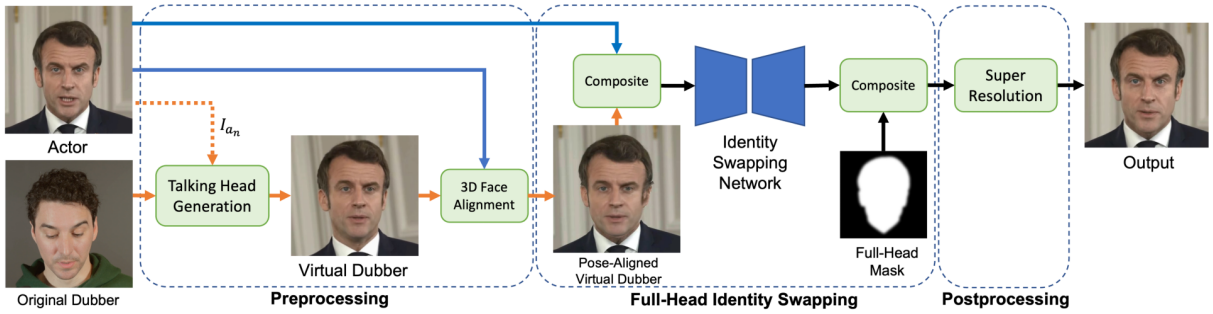

Figure 1: ***Overall pipeline: inference.*** *We first generate the virtual dubber video from one actor frame and the dubber video. Next, we align the pose of the virtual dubber to the actor's (Section 2.1: Preprocessing, supplementary material Figure 1), and from this point, the process is done frame by frame. We then transfer the actor identity to the virtual dubber through the identity swapping network (Section 2.2: Full-Head Identity Swapping). Finally, we upscale the output with the identity-specific fine-tuned super resolution (Section 2.3: Postprocessing).*

the whole head (including neck) back onto the same region of the actor frame, using the same head mask from training. Reenacting the neck along with the face helps address the double chin problem, as it can reduce the size of the chin (see Figure 2 in the supplementary material for the compositing steps).

2.3. Postprocessing

Finally, we perform **identity-specific super resolution**. The output of the trained model 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. To address this, we fine-tune the pretrained super-resolution method GPEN [YRXZ21] to be actor-specific. Inspired by Patel et al., we use a training set of both the original actor frames and output of the trained identity swapping network through the actor decoder. We produce outputs at 512×512 with enhanced faces retaining actor-specific details.

3. Experiments

Data: Our work aims to perform well in real-world scenarios. To this end, we use videos of public figures (Obama, Macron, Kovind, and Merkel) ranging from 10s to 37s in length, paired with professional dubber videos, speaking the translated script.

Implementation details: The network is pretrained on the CelebA-HQ dataset [KALL18] for 150k iterations using the Adam optimizer (learning rate 1e-4, batch size). After initializing with the pretrained weights, the person-specific model is trained for 65k iterations. We use a batch size of 64 and a learning rate of 5e-5 with the Adam optimizer. Additionally, we set $\lambda_{mouth} = 0.01$ and $\lambda_2 = 4$, while the other λ values are set to 1. The person-specific training takes approximately 2.5 days on four NVIDIA V100 32GB GPUs. For super-resolution, we utilize the training code from GPEN with a batch size of 4 and a learning rate of 2e-2.

Baselines: Our comparison includes Wav2Lip [PMNJ20] and TalkLip [WQZ*23] for audio-driven methods, and Patel et al. [PZM*23] and LivePortrait [GZL*24] as video-driven methods. We used their official implementations.

Metrics: We evaluate our method using two criteria: visual quality and lip-sync quality, adopting the most commonly used metrics. For visual quality, we use PSNR, FID [HRU*17], and LPIPS [ZIE*18], with the original actor video as the reference. For lip-sync quality, we compute LMD [CLM*18], using the dubber's mouth landmarks as the ground truth.

Evaluation. Table 1 shows our quantitative evaluation. Patel et al.'s method achieves superior results across most visual quality metrics, particularly in the Kovind sequence. We attribute this difference to the design of their method that composites only the mouth region. In contrast, our method reenacts a larger portion of the video, which introduces a risk of reducing the visual quality and slight metric divergence. Despite this, our method achieves competitive scores, achieving the second-best overall and the best LPIPS score for the Obama and Macron sequences, demonstrating the robustness in preserving actor specific details and overall consistency. Similarly, while PSNR for the Merkel sequence might suggest more noise, both FID and LPIPS reflect strong perceptual accuracy.

Moreover, our method consistently excels in lip-sync quality metrics (LMD), demonstrating superior alignment with the dubber's expressions. Only for the Kovind sequence, does LivePortrait surpass our method for LMD. Despite the high score, the head pose occasionally differs from the original actor's pose for LivePortrait. 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.

We show our qualitative evaluation in Figure 4 of the supplementary material and the accompanying video. Wav2Lip achieves accurate lip-sync but with lower visual quality, while TalkLip generates blurry results with a noticeable lag compared to the original actor video. LivePortrait produces high-quality outputs but struggles with inconsistent head pose, sometimes following the dubber's. Moreover, it occasionally generates uncanny-looking eyes. Patel et al.'s method delivers comparable results to ours, but lacks expressiveness and occasionally introduces artifacts. Our method

Method	Obama				Macron				Kovind				Merkel			
	PSNR↑	FID↓	LPIPS↓	LMD↓	PSNR↑	FID↓	LPIPS↓	LMD↓	PSNR↑	FID↓	LPIPS↓	LMD↓	PSNR↑	FID↓	LPIPS↓	LMD↓
Wav2Lip	35.07	8.53	0.10	1.55	38.41	1.71	0.06	0.79	36.79	4.34	0.07	1.04	38.61	6.73	0.07	1.49
TalLlip	32.55	13.77	0.15	1.44	33.98	7.81	0.15	0.80	34.04	5.40	0.10	1.00	32.72	17.51	0.19	1.68
Patel et al.	**38.92**	**2.74**	0.06	1.42	**48.89**	**0.23**	0.01	0.82	**54.19**	**0.28**	**<0.01**	1.07	**49.37**	1.66	**<0.01**	1.30
LivePortrait	31.72	9.95	0.23	1.78	39.54	3.31	0.04	0.80	38.64	7.19	0.04	**0.79**	39.93	5.16	0.05	2.34
Ours	37.56	3.71	**0.06**	**1.21**	47.56	0.29	**0.01**	**0.54**	46.44	1.15	<0.01	0.96	48.63	**0.96**	0.01	**1.25**

Table 1: ***Quantitative evaluation. Bold*** *indicates the best, and underline indicates the second best. The values are rounded to two decimal places.*

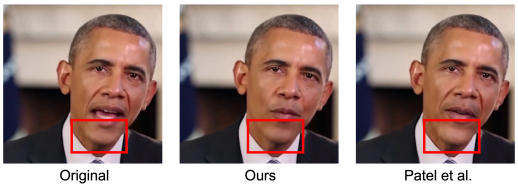

Figure 2: ***Handling double chin.*** *Ours: jaw is adjusted to align with the synthesized expression. Patel: yields an odd-looking chin.*

stands out for reconstructing challenging dubber expressions, such as mouth puckering. Additionally, our full head reenactment, including the neck and jaw, eliminates artifacts such as the double-chin effect by enabling subtle adjustments to the jawline, unlike Patel et al. which only composite the mouth, as illustrated in Figure 2. Although our results are slightly more blurry than the sharpest result of Patel et al., this trade-off achieves a more balanced output with lip-sync accuracy and realistic facial reenactment.

4. Conclusion and Limitations

In this paper, we introduce a visual dubbing pipeline that leverages a virtual dubber. While inspired by Patel et al., our work has several major differences: introduction of the virtual dubber, complete redesign and simplification of the face alignment, removal of the recirculation in the identity swapping network, and full-head reenactment. Our quantitative and qualitative comparisons against competitive methods demonstrate that our method achieves a superior balance between visual quality and lip synchronization. Our outputs exhibit slight blurring and color shifts, which could be improved by enhancing the super-resolution module. Additionally, occasional temporal inconsistency arises due to the lack of explicit postprocessing, which could be addressed with manual corrections or a temporal consistency module.

Acknowledgements. We would like to thank Serge Laforest and Tintin Rouillard from AudioZ and Webcargo for the data they provided and their invaluable inputs and discussions.

References

[Cho17] CHOLLET F.: Xception: Deep learning with depthwise separable convolutions. In *Proc. of IEEE CVPR* (2017), pp. 1251–1258. 2

[CLM*18] CHEN L., LI Z., MADDOX R. K., DUAN Z., XU C.: Lip movements generation at a glance. In *Proc. of ECCV* (2018). 3

[FWS*18] FENG Y., WU F., SHAO X., WANG Y., ZHOU X.: Joint 3D face reconstruction and dense alignment with position map regression network. In *Proc. of ECCV* (2018), pp. 534–551. 2

[GZL*24] GUO J., ZHANG D., LIU X., ZHONG Z., ZHANG Y., WAN P., ZHANG D.: Liveportrait: Efficient portrait animation with stitching and retargeting control. *CoRR abs/2407.03168* (2024). 2, 3

[HRU*17] HEUSEL M., RAMSAUER H., UNTERTHINER T., NESSLER B., HOCHREITER S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems 30* (2017). 3

[KALL18] KARRAS T., AILA T., LAINE S., LEHTINEN J.: Progressive growing of GANs for improved quality, stability, and variation. In *Int. Conf. on Learning Representations* (2018). 3

[KEZ*19] KIM H., ELGHARIB M., ZOLLHÖFER M., SEIDEL H.-P., BEELER T., RICHARDT C., THEOBALT C.: Neural style-preserving visual dubbing. *ACM TOG 38*, 6 (Nov. 2019). 1

[KGT*18] KIM H., GARRIDO P., TEWARI A., XU W., THIES J., NIESSNER M., PÉREZ P., RICHARDT C., ZOLLHÖFER M., THEOBALT C.: Deep video portraits. *ACM TOG 37*, 4 (2018), 1–14. 1

[KLA19] KARRAS T., LAINE S., AILA T.: A style-based generator architecture for generative adversarial networks. In *CVPR* (2019). 2

[PMNJ20] PRAJWAL K., MUKHOPADHYAY R., NAMBOODIRI V. P., JAWAHAR C.: A lip sync expert is all you need for speech to lip generation in the wild. In *Proc. of the 28th ACM int. conf. on multimedia* (2020), pp. 484–492. 3

[PZM*23] PATEL D., ZOUAGHI H., MUDUR S., PAQUETTE E., LAFOREST S., ROUILLARD M., POPA T.: Visual dubbing pipeline with localized lip-sync and two-pass identity transfer. *Comput. Graph. 110*, C (Feb. 2023), 19–27. 1, 3

[TET*20] THIES J., ELGHARIB M., TEWARI A., THEOBALT C., NIESSNER M.: Neural voice puppetry: Audio-driven facial reenactment. In *Proc. of ECCV* (2020), pp. 716–731. 1

[WQZ*23] WANG J., QIAN X., ZHANG M., TAN R. T., LI H.: Seeing what you said: Talking face generation guided by a lip reading expert. In *Proc. of IEEE CVPR* (2023), pp. 14653–14662. 3

[WYW24] WEI H., YANG Z., WANG Z.: Aniportrait: Audio-driven synthesis of photorealistic portrait animation. *arXiv* (2024). 1

[YRXZ21] YANG T., REN P., XIE X., ZHANG L.: Gan prior embedded network for blind face restoration in the wild. In *Proc. of IEEE CVPR* (2021), pp. 672–681. 3

[YZR*24] YE Z., ZHONG T., REN Y., YANG J., LI W., HUANG J., JIANG Z., HE J., HUANG R., LIU J., ZHANG C., YIN X., MA Z., ZHAO Z.: Real3D-portrait: One-shot realistic 3D talking portrait synthesis. *ICLR* (2024). 1

[ZCW*23] ZHANG W., CUN X., WANG X., ZHANG Y., SHEN X., GUO Y., SHAN Y., WANG F.: Sadtalker: Learning realistic 3D motion coefficients for stylized audio-driven single image talking face animation. In *Proc. of IEEE CVPR* (2023), pp. 8652–8661. 1

[ZIE*18] ZHANG R., ISOLA P., EFROS A. A., SHECHTMAN E., WANG O.: The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. of IEEE CVPR* (2018), pp. 586–595. 3