Face Reenactment: Crafting Realistic Talking Heads for Enhanced Video Communication and Beyond

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

Masters of Science
in
Computer Science and Engineering by Research

by

Madhav Agarwal
2020900022
madhav.agarwal@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, INDIA
June, 2023

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Face Reenactment: Crafting Realistic Talking Heads for Enhanced Video Communication and Beyond" by Madhav Agarwal, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Adviser: Prof. C. V. Jawahar
Date	Adviser: Prof. Vinay P. Namboodiri

To, my late Grandmother.

I hope you are at peace and surrounded by love.

Acknowledgments

I want to take this opportunity to express my deep gratitude to Prof C.V. Jawahar and Prof Vinay P. Namboodiri for their invaluable guidance and support during my MS journey. Their expertise and knowledge in the field have been instrumental in shaping my research and academic development. Their insightful feedback, constructive criticism, and encouragement have inspired me to strive for excellence and pursue my research with passion and dedication. I am also grateful for the time and effort they invested in mentoring me throughout the course of my research. Their patience, availability, and willingness to discuss complex ideas and theories have been truly invaluable. I am indebted to them for their unwavering support and guidance. I want to thank them for being wonderful role models and inspiring me to pursue my academic goals with commitment and dedication. Their guidance has been a source of strength and motivation, and I will always cherish the knowledge and insights I have gained from them.

I would also like to thank Dr. Ajoy Mondal for guiding me during my initial research year. The continuous guidance, discussions, and willingness to work thoroughly on a problem were truly inspiring. Sachin has been like a big brother to me. He always motivated me to put in the extra effort while doing research.

Throughout my Master's program, Rudrabha, my project partner and friend, has been a constant source of help and guidance, and it would be neglectful not to express my gratitude towards him. I remember the time before the WACV submission when I was struggling with my research and feeling overwhelmed by the challenges of my work. He spent countless hours helping me work through my problems, brainstorming solutions with me, and offering words of reassurance when I needed them the most.

Quoting Barney- *"Whatever you do in this life, it's not legendary, unless your friends are there to see it"*. I was fortunate enough to make many friends during my journey. The first person I met was Siddhant. From the very first day, our journey at IIIT was intertwined; whether it was working in the document domain, deciding to join MS, or taking courses together, we shared a lot in those three years. Pranav and Zeeshan were partners in crime. Most of my memories of IIIT can't go down without their involvement. The unplanned hangouts, going out to attend events, late-night chats at felicity, there is no end to this list. Shubham, my swimming buddy and movie partner, hardly said "No" to anything. I can never forget how we successfully wrote and submitted an entire application in one night. Shivanshu, the Intern, was always fun to be around. I will always cherish the memories of those moments and the sense of belonging and camaraderie we shared.

Furthermore, I would like to thank my CVIT lab mates Ravi, Prafful, Soumya, Siddharth, Anchit, Aditya, Bipasha, Seshadri, Rupak, George, Avijit, Darshan, Piyush, and Shashank for making my time in the lab memorable and fun. As I reflect on my MS journey, I realize that I could not have made it this far without your support. Your guidance, feedback, and encouragement have been invaluable to me, and I feel incredibly lucky to have you all in my life.

Rohitha and Aradhana helped a lot in taking care of administrative tasks and making plans to attend the conference. I would also like to thank Language and Content Editing Support Team (LACES) for proofreading my work, including papers, supplementary, and thesis.

I could never imagine completing my MS without the constant support of my two closest friends, Komal and Bhavya. From scolding me for all the stupid decisions I take (which I do a lot) to celebrating my small achievements, you always supported me in every ups and downs. Thanks for showing the true meaning of *"I'll be there for you"*. You have been not only my friends but also my confidants, and cheerleaders, and I will always be grateful for your role in shaping me into the person I am today.

Finally, I would like to express my gratitude to my parents and brother for their selfless support and affection. Your unwavering encouragement and support have been a source of strength for me, and I cannot thank you enough for all that you have done.

Abstract

Face Reenactment and Synthetic Talking Head works have been widely popular for creating realistic face animations by using a single image of a person. In light of the recent developments in processing facial features in images and videos, as well as the ability to create realistic talking heads, We are focusing on two promising applications. These applications include utilizing face reenactment for movie dubbing and compressing video calls where the primary object is a talking face. We propose a novel method to generate realistic talking head videos using audio and visual streams. We animate a source image by transferring head motion from a driving video using a dense motion field generated using learnable keypoints. We use audio as an additional input for high-quality lip sync, by helping the network to attend to the mouth region. We use additional priors using face segmentation and face mesh to preserve the structure of the reconstructed faces. Finally, we incorporate a carefully designed identity-aware generator module to get realistic quality of talking heads. The identity-aware generator takes the source image and the warped motion features as input to generate a high-quality output with fine-grained details. Our method produces state-of-the-art results and generalizes well to unseen faces, languages, and voices. We comprehensively evaluate our approach using multiple metrics and outperforming the current techniques both qualitative and quantitatively. Our work opens up several applications, including enabling low-bandwidth video calls and movie dubbing.

We leverage the advancements in talking head generation to propose an end-to-end system for video call compression. Our algorithm transmits pivot frames intermittently while the rest of the talking head video is generated by animating them. We use a state-of-the-art face reenactment network to detect keypoints in the non-pivot frames and transmit them to the receiver. A dense flow is then calculated to warp a pivot frame to reconstruct the non-pivot ones. Transmitting keypoints instead of full frames leads to significant compression. We propose a novel algorithm to adaptively select the best-suited pivot frames at regular intervals to provide a smooth experience. We also propose a frame-interpolater at the receiver's end to improve the compression levels further. Finally, a face enhancement network improves reconstruction quality, significantly improving several aspects, like the sharpness of the generations. We evaluate our method both qualitatively and quantitatively on benchmark datasets and compare it with multiple compression techniques.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Videos are a rich source of information and are widely used in day-to-day activities, ranging from entertainment and broadcasting information to communication in the form of video calls. The rapid growth in deep learning has changed the way we use and look at videos. Generative models [17] have enabled us to create synthetic images and videos using our imagination. It is now possible to create realistic human faces which do not exist in the real world [21]. The scope has even broadened with the advancement of diffusion models [15, 41]. Generating and editing images directly from the text prompt is now possible. One exciting application of generating models is face reenactment. The problem has been studied for a long time in the research community. The initial focus was to transfer the lip motion to create a lip-synchronized video, followed by the transfer of the entire motion and expression of the talking face. A lip-sync video provides an immersive viewing experience while watching it in a different language. It also eliminates the need for hours of effort by dubbing artists. The motion and expression transfer has made it possible to create face-swap videos without an expensive VFX setup[1]. It has a lot of potential to transform the entertainment industry and redefine visual storytelling, along with video compression and transmission.

Face reenactment is a fascinating area to work on, but at the same time, it is quite challenging. Given a source image and a driving video, the face reenactment aims to transfer the motion and expression from the driving video onto the identity source image and create a realistic-looking talking head video of that particular identity. The source image and driving video can be of different identities as well. The main reason it is challenging is the fine-grained detailing in human faces and the infinite number of expressions and motions a human face can generate. It isn't easy to quantify human expressions. The face structure is different for different individuals. The method should be robust enough to handle head rotation or eye movement and create sharp facial regions, such as teeth. The background in a video creates an additional degree of motion and can be very difficult to handle. Generating occluded or unseen areas in the image is also a challenge, as it should blend closely with the surroundings while maintaining the realism of the rendered video. This thesis explores talking head generation from a

[1] https://en.wikipedia.org/wiki/Visual_effects

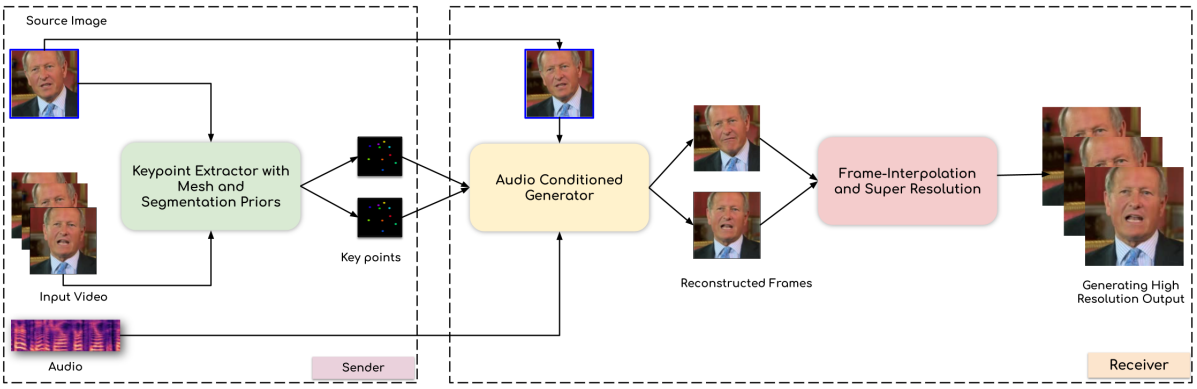

Figure 1.1 Face reenactment pipeline depicting high-resolution talking head generation using a single source image and a driving video. For video calling, only keypoints need to be transmitted along with source image and audio.

single identity image and the promising applications of face reenactment, including but not limited to high-quality movie dubbing and video call compression.

The problem of transferring motion while maintaining the identity information has been an active research area. Talking head generation researchers have focused on either of the three modalities to drive motion, namely: text, audio, or video. Text-driven methods, such as Li et al.[25] and Txt2Vid[50], rely on text to drive animation parameters or convert spoken language into text for low-bandwidth video conferencing. However, these methods heavily depend on generated speech, which can alter the original speaker's voice and introduce grammatical errors. They also lack fine-grained control over head and lip movements, making the problem ill-posed. Audio-driven methods, on the other hand, utilize audio as a more expressive and informative input. Early approaches like You-said-that?[13] and LipGAN[24] focused on achieving lip synchronization but failed to generate synchronized head movements. Later works, such as those by Song et al.[47] and Zhou et al.[63], employed conditional Recurrent Neural Networks to model temporal characteristics and disentangle audio and visual representations. Zhou et al.[65] introduced the use of dense flow to warp the source image based on audio, while Emotional Video Portraits[20] added emotion labels as input. However, these methods still face challenges in handling non-verbal cues, facial expressions, and modeling background information. Video-driven methods use a driving video to obtain motion and facial features necessary for reenacting a source image. The influential First-Order Motion Model (FOMM) by Siarohin et al.[44] estimated motion fields from sparse keypoints and used them to warp the source image. Subsequent works, such as Face-vid2vid[56] and DA-GAN [18], built upon FOMM's principles to improve quality. PC-AVS [64] combined audio and video to formulate pose and motion codes, achieving lip sync but with lower overall video quality compared to DA-GAN. These methods, however, still face challenges in terms of facial expressions, non-face regions, and controlling pose and expressions.

The major challenge to decouple the identity and pose information from given face images can be tackled by learning sparse keypoints from the face images and using them to create dense motion fields. The motion fields warped the feature map of the source image and drove the motion and expression based on the driving frame. The keypoints learned in this manner lack any information about the face structure and background. This generally leads to distortion and artifacts when the source and driving face structures are poorly aligned. The prevailing methods use either video or audio to create talking heads. Each method has its own drawbacks: audio-driven methods create better lip-sync, while video-driven methods create better head motions. The driving videos are generally accompanied by an audio signal. However, the researchers still need to adequately study combining audio and visual modalities. The other major challenge is to preserve the source identity, which generally got affected due to the involvement of driving images while estimating the optical flow.

Face reenactment networks hold immense practical value, such as it can be used to transmit talking head video calls at very low bandwidth. Instead of traditional video compression methods, which compress and transmit each frame, facial keypoints can be used to recreate talking faces at the receiving end. This method significantly reduces bandwidth since only ten keypoints, or 80 bytes of information, are required to generate a frame. However, standard talking head generation networks had limitations, such as difficulty handling large head movements, failures at high resolution, and the need to transmit keypoints for every frame. To address these limitations, a carefully designed pipeline is needed, which can be used for video calling. It should also be compatible with any standard keypoints based face reenactment network such as FOMM [44], to enable better adaptability with any future research work.

1.1 Contributions

In this thesis, we look into two primary facets: 'Where' the talking heads have promising applications and 'How' they can be improved to tailor the requirements of modern-day realistic media. We tackle many of the above-mentioned challenges in face reenactment. We propose many novel approaches that give much better results in comparison to the previous work in this domain. The main contributions of the thesis are as follows:

- We propose a face reenactment network that generates high-quality talking heads.
  - We propose the use of face mesh and face segmentation mask as additional priors to preserve the face structure.
  - We utilize the audio modality to improve the quality of lip synchronization in talking head videos.
  - The audio-visual attention help in better generation quality of mouth region.
  - We propose a novel identity-aware generator that produces high-quality output with fine-grained details leading to more realistic talking head videos with fewer flickers and artifacts.

- We propose the application of synthetic talking heads for video calling at extremely low bandwidth.
  - We carefully designed a pipeline that utilizes ten keypoints, or 80 bytes of information, to generate a talking head frame.
  - An adaptive frame selector was introduced to handle large head movements.
  - A patch-based super-resolution network was proposed to upscale generated images at high resolution.
  - A frame interpolation network was used to eliminate the need to transmit keypoints for every frame.
  - The various modules of the proposed pipeline can be used directly with standard keypoint-based face reenactment methods.

We discuss more about contributions in detail in Chapter 2 and Chapter 3.

1.2 Organization of Thesis

The rest of the thesis is organized as follows.

- In Chapter 2, we discuss about using the audio and visual modality for creating a novel face reenactment network. We demonstrated the effectiveness of our network in creating high-quality talking heads [8].
- In Chapter 3, we discuss the practical application of synthetic talking heads for low-bandwidth video calling. We show that our carefully designed pipeline can achieve better results in comparison to standard video compressing techniques while maintaining a comparable visual quality [7].
- Chapter 4 presents the concluding thoughts and future directions of talking head generation.

Chapter 2

Audio-Visual Face Reenactment

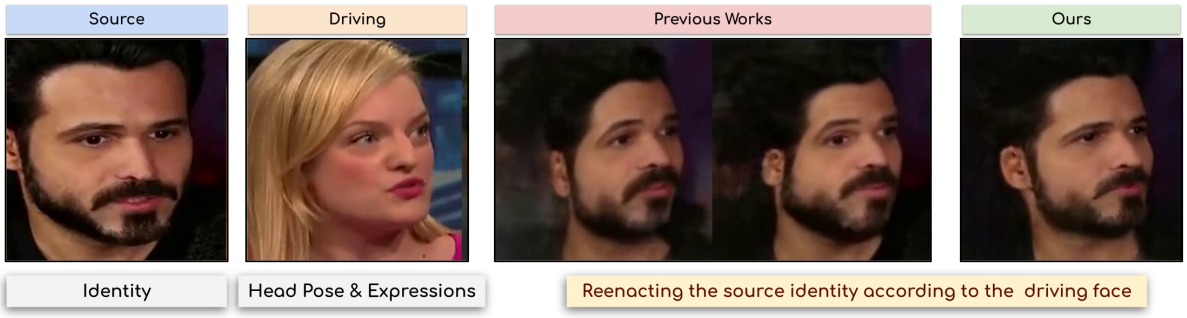

Figure 2.1 We propose AVFR-GAN, a novel method for face reenactment. Our network takes a source identity, a driving frame, and a small audio chunk associated with the driving frame to animate the source identity according to the driving frame. Our network generates highly realistic outputs compared to previous works like [44] and [45]. Results from our network contain significantly fewer artifacts and handle things like mouth movements, eye movements, etc. in a better manner.

Imagine your favorite celebrity giving daily news updates, motivating you to work out, or interacting with you on your mobile phone! What if a movie director could reenact an actor's image without actually recording the actor? Or, how about skilled content creators animating avatars in a metaverse to follow an actor's head movements and expressions in great detail? We can also reduce zoom fatigue [16] by animating a well-dressed image of ourselves in a video call without transmitting a live video stream! These ideas seem fictitious, infeasible, and not scalable. But, how about animating or "reenacting" a single image of any person according to a driving video of someone else? Face reenactment, thus, opens up many opportunities in a world that is becoming increasingly digital with each passing day.

Face Reenactment aims to animate a source image using a driving video's motion while preserving the source identity. Multiple publications have improved the quality of the generations. Existing works

[0]Webpage for the Paper: http://cvit.iiit.ac.in/research/projects/cvit-projects/avfr

on talking head generation generally use a single modality, i.e., either visual[18, 44, 56, 60] or audio features[20, 54, 47]. Audio-driven talking head generation models are good at generating quality lip-sync; however, they have a serious drawback in handling non-verbal cues. The video-driven methods heavily rely on the disentanglement of motion from the appearance [26]. These methods generally use keypoints as an intermediate representation [44, 18, 56] and try to align the detected keypoints of source and driving frames. These works learn keypoints in an unsupervised manner and fail to focus on specific regions of the face. This stems from inadequate priors regarding the face structure or the uttered speech. The final quality of the generations also suffers from using a basic CNN-based decoder that fails to capture the sharpness present in the source image and generates blurred output video. As a part of this work, we provide a detailed review of different approaches in Section 2.1.

In this chapter, we analyze the shortcomings of the current works and add key modules to our network. We introduce **A**udio-**V**isual **F**ace **R**eenactment **GAN** (**AVFR-GAN**), a novel architecture that uses both audio and visual cues to generate highly realistic face reenactments. We start with providing additional priors about the structure of the face in the form of a face segmentation mask and face mesh. We also provide corresponding speech to our algorithm to help it attend to the mouth region and improve lip synchronization. Finally, our pipeline uses a novel identity-aware face generator to improve the final outputs. Our approach generates superior results compared to the current state-of-the-art works, as shown in Section 2.3. We comprehensively evaluate our method against several baselines and report the quantitative performance based on multiple standard metrics. We also perform human evaluations to evaluate qualitative results in the same section. Our proposed method opens a host of applications, as discussed in Section 2.5, including one in compressing video calls. Our work achieves more than 7$\times$ improvement in visual quality when tested at the same compression levels using the recently released H.266 [11] codec.

Our contributions of this chapter are summarized as follows:

1. We use additional priors in the form of face mesh and face segmentation mask to preserve the geometry of the face.
2. We utilize additional input in the form of audio to improve the generation quality of the mouth region. Audio also helps to preserve lip synchronization, enhancing the viewing experience.
3. We build a novel carefully-designed identity-aware face generator to generate high-quality talking head videos in contrast to the high levels of blur present in the previous works.

2.1 Related Work

Talking head generation works can be broadly classified in three categories based on the type of input they use to generate a talking head: Text-driven [25, 50, 53], Audio-driven [13, 20, 28, 47, 54, 63, 65], and Video-driven [18, 39, 44, 56, 64] Talking Head Generation.

2.1.1 Text-driven Talking-head Generation

Text-driven natural image generation [37, 38] has recently seen a lot of progress in the computer vision community. Inspired by the recent success of GANs in generating static faces from text[55], Li *et al*. [25] proposed a method to use text for driving animation parameters of the mouth, upper face and head. Txt2Vid [50] converts the spoken language and facial webcam data into text and transmits it to achieve low-bandwidth video conferencing using talking head generation. However, this method relies heavily on the generated speech, altering the original speaker's voice, prosody, and head movements in the video call. It depends on the quality of the Speech-to-Text module, which introduces grammatical errors and language dependency. Text as a medium has very little information about the head and lip movements; thus, we consider the problem ill-posed.

2.1.2 Audio-driven Talking-head Generation

While text-driven methods suffer from a significant lack of adequate priors, we now move on to audio, a much more expressive and informative form of input. As the name suggests, audio-driven methods [13, 20, 28, 47, 54, 63, 65] use only audio to animate a static face image. The first set of works like You-said-that? [13], LipGAN [24] and Wav2Lip [36] achieved lip synchronization with given audio but failed to generate head movements in sync with the speech. These works used fully convolutional architectures and generated a single frame at a time without considering the temporal constraints. Eventually, a different class of works starting from Song *et al*. [47] in 2018 and Zhou *et al*. [63] in 2019, started using conditional Recurrent Neural Networks to model the temporal characteristics of a talking face. In 2020, Zhou *et al*. [65] published a landmark work that predicted dense flow from audio instead of directly generating the output video. The dense flow was then used to warp the source image to generate the final output. Several other well-known works like Emotional Video Portraits [20] add an additional emotion label as input to create the talking head in the desired emotion. However, all of these works lack fine-grained control of the talking head and often contain a loopy head motion, and thus cannot be directly used in many applications.

2.1.3 Video-driven Talking-head Generation

Finally, we move to video-driven methods, which use a driving video to get the motion and other facial features required to reenact a source image. Please note that the driving video and the source image may not have the same identity. Owing to the significant priors in driving video, the final generation quality of video-driven methods surpasses those of text-only and audio-only ones. The most influential work in this area, First-Order-Motion-Model (FOMM), was published by Siarohin *et al*. [44] in 2019. The key idea was to estimate the motion field from sparse keypoints detected in both source and driving frames. The motion field was used to calculate dense flow and warp the source frame in a latent space. Several other works [56, 18] followed the same principle and added supplementary components to improve the quality. Face-vid2vid [56] used keypoint information in a 3D space, taking care of head

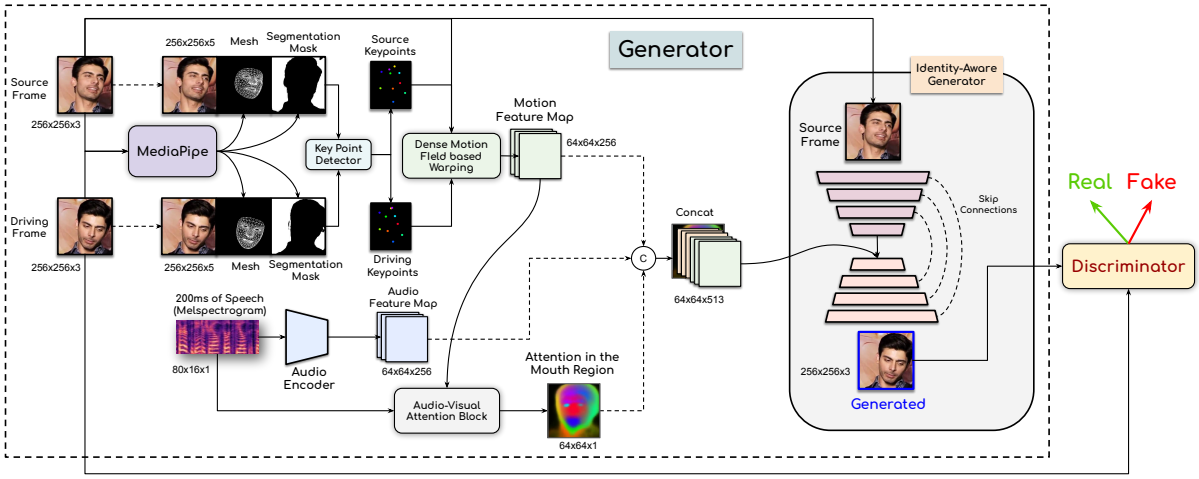

Figure 2.2 The overall pipeline of our proposed Audio Visual Face Reenactment network (AVFR-GAN) is given in this Figure. We take the source and driving images, along with their face mesh and segmentation masks to extract keypoints. An audio encoder extracts features from driving audio and use them to provide attention on lip region. The audio and visual feature maps are warped together and passed to the carefully designed Identity-Aware Generator along with extracted features of the source image to generate the final output.

rotation, among other things. DA-GAN[18] further added depth-aware attention to provide dense 3D facial geometry to guide the generation of motion fields. A similar approach in Motion-Representation-in-Articulated-Animation [45] uses key regions instead of keypoints to generate the warpable motion field. Approaches like ICface[51] provide a method to control the pose and expressions of a face image using head pose angles and action unit values. Recently, Zhang *et al.* [62] proposed using the three-dimensional morphable face model (3DMM) parameters to reenact a face image. They demonstrated that motion descriptor parameters for 3DMM can be derived from a driving video and, in turn, animate a static facial image.

To the best of our knowledge, PC-AVS [64] is the only work that uses audio and video to formulate a low-dimension pose and motion code. Unlike FOMM, PC-AVS does not predict motion fields to calculate dense flow and warp the source image. Instead, they try to train their network to learn motion in a latent space inherently. While this allows them to achieve state-of-the-art lip sync, the generated video's overall quality is considered inferior to works like DA-GAN [18]. In this work, we base our approach on FOMM's [44] principles and improve it with additional audio information. We also provide additional structural information to extract better geometries of the face. This allows us to use the best of both worlds and propose a novel network AVFR-GAN as described in the next section.

2.2 Audio-Visual Face Reenactment GAN

We present **A**udio-**V**isual **F**ace **R**eenactment GAN (**AVFR-Gan**), which takes a source image and a driving video plus audio to create high-quality talking head videos by preserving the source identity. As mentioned previously, we follow a similar strategy to that of FOMM [44] for our training pipeline. Instead of generating multiple frames in the form of a video, we handle the input in a frame-by-frame fashion. Our main goal is to estimate the motion between a source and a driving frame and then warp the source frame accordingly to generate an approximation of the driving frame. Our model can be broadly divided into a Generator M_{Gen} and a discriminator M_{Disc} as shown in Figure 2.2. We first discuss the individual components present inside the generator.

2.2.1 Additional Structural Priors to the Keypoint Detector

We start with selecting a source frame F_s and a driving frame F_d both of dimensions $h \times w$. During training, both of these frames are selected from the same video. We pass these frames through mediapipe [29] to generate a face mesh and a face segmentation map. We channel-wise concatenate the generated mesh and the segmentation mask with their respective images and create 5 channel versions of the same. We term the concatenated source and driving frames as I_s and I_d, respectively. We use these concatenated inputs to feed into our keypoint detector, M_{kp}. The addition of these priors helps us in providing the keypoint detector with more information about the respective structures of source and driving frames. Furthermore, the segmentation mask also provides the module with foreground and background information enabling the keypoints to be detected only from the foreground. We use the keypoint detector from FOMM [44] in our architecture. The keypoint detector M_{kp} detects K keypoints. More concretely, we can write,

$$\{X_{T,n}\}_{n=1}^{K} = M_{kp}(I_T), T \in s, d \tag{2.1}$$

The difference between the generated keypoints from the source and driving frames is used to calculate the motion field following FOMM. The motion field is then used to calculate dense flow and generate a warped feature map. We denote this feature map as Motion Feature Map, Enc_{motion} as it captures the motion between the source and the driving frames. The dimension of this feature map is kept to be $\frac{h}{4} \times \frac{w}{4} \times c$. We plot sample keypoints detected in specific frames in Figure 2.5 (left). Also, note that each keypoint has a specific region of interest in the generated motion field. We plot the heatmaps for each keypoint in Figure 2.5 (middle). The heatmaps show that the regions of interest for each keypoints correspond to specific facial features. For example, the dark blue keypoint attends to the mouth region, green attends to the jaw, and sky blue attends specifically to the eye regions. Interestingly both of the eyes are attended by the same keypoint.

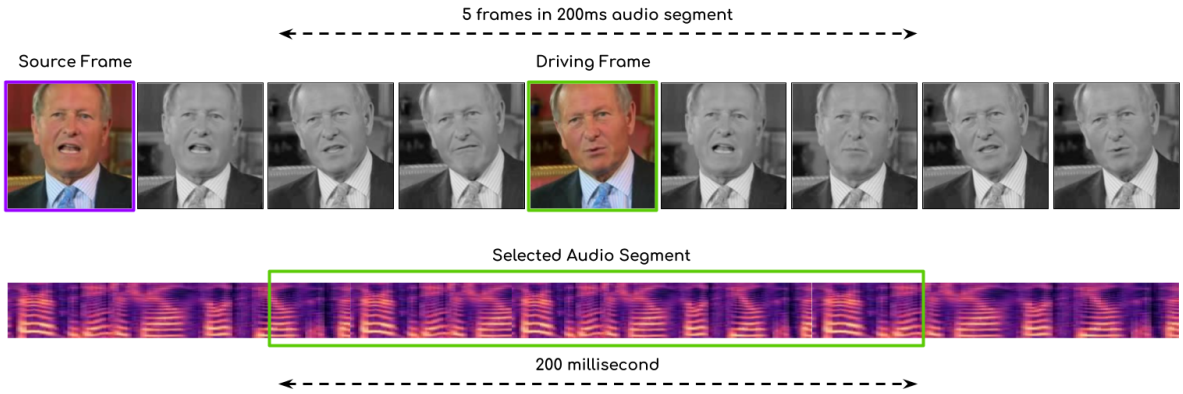

Figure 2.3 Illustration of Audio window selector mechanism. It generates a 200ms spectrogram such that the driving frame remains in the middle of the segment. In case of a 25 FPS video, a 200ms segment contains 5 frames.

2.2.2 Audio-conditioned Features

Audio (mainly speech in our case) is an essential source of information that often accompanies a talking-head video. We decided to use the speech from the driving video to improve the quality of mouth movements in the generated video. While works like MakeItTalk [65] have already generated head movements solely from audio, our goal is to only improve the mouth movements and transfer head motion directly from the driving video. Therefore, we follow the same strategy taken by lip-synchronization works like [13, 24, 36] to handle speech. We select the 200ms window of speech around our driving frame F_d such that F_d is the middle frame in the sampling window. A graphical representation of the audio window selection is given in Figure 2.3. We generate melspectrogram I_{mel} from the speech window and feed it to a 2D CNN-based encoder. The encoder contains a series of convolution blocks with upsampling layers. The audio encoder outputs a feature map, Enc_{aud}, of $\frac{h}{4} \times \frac{w}{4} \times c$ dimension. We concatenate $(Enc_{motion}, Enc_{aud})$ along with the attention map generated as described next.

2.2.3 Audio-Visual Attention

Apart from improving the lip synchronization in the generated video, we propose using audio to specifically attend to the speaker's mouth region, enhancing the fine-grained details like teeth in the generated video. To do this, we pass I_{mel} through an attention encoder generating an encoding Enc_{query} of dimensions $1 \times 1 \times c$. We then take Enc_{motion} of dimension $\frac{h}{4} \times \frac{w}{4} \times c$ and calculate the dot product at each location with Enc_{query}, generating a $\frac{h}{4} \times \frac{w}{4} \times 1$ matrix. We pass this through a Sigmoid layer to get the attention map Enc_{attn} as shown in Figure 2.4. A formal definition of this block is given in Equation 2.2.

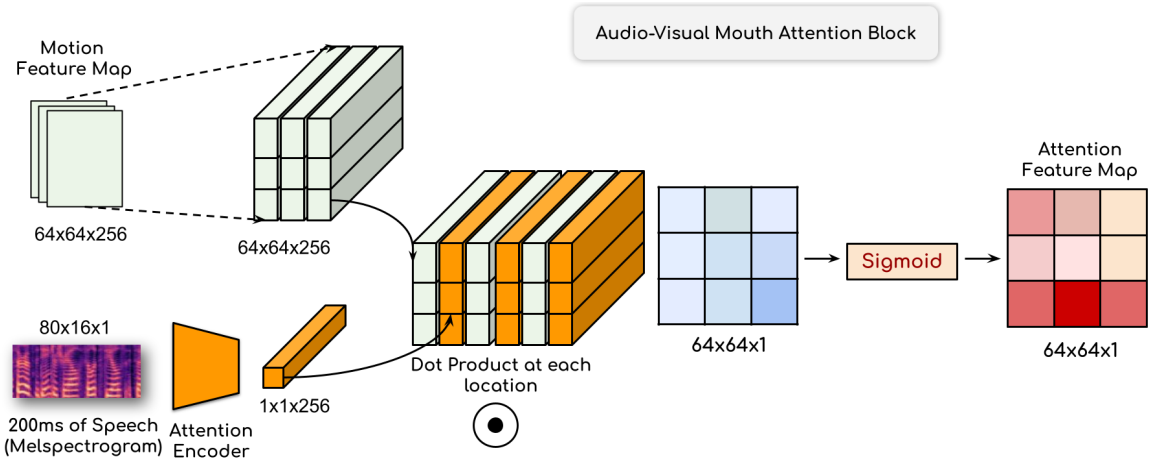

Figure 2.4 Illustration of Audio Visual Attention module. Attention is generated by taking the dot product between a learned audio feature and visual features in each location, followed by a Sigmoid activation.

$$Enc_{attn}(i,j) = Sigmoid(Enc_{query} \odot Enc_{motion}(i,j)),$$
$$i \in [1, \frac{w}{4}], j \in [1, \frac{h}{4}] \tag{2.2}$$

A visualization of the audio-visual attention can be found in Figure 2.5. As we can see, audio not only helps the model to attend to the mouth region but also helps the network attend to other regions like the eyes, which correlates to expressions from speech.

2.2.4 Identity-Aware Generator

We propose a novel generator to decode the concatenated feature vector. We analyze the current decoders used in FOMM [44], Face-Vid2Vid [56] and DA-GAN [18]. We realize that the pipelines followed by the current works fail to capture information from the source image directly. The network entirely depends on the warped features generated from the motion estimator to get the identity characteristics of the source speaker. Unfortunately, the warped features are forced to encode motion and fine-grained identity information, making it tougher to train. This ultimately causes the outputs to contain major artifacts and lose sharpness. We improve upon this and design an identity-aware face generator. We first concatenate Enc_{motion}, Enc_{con} and Enc_{attn} together to get the final warped features, generating Enc_{dec}. Instead of only feeding the warped features, we also feed in the source image F_s separately to the UNet-shaped [42] generator. The generator consists of an identity-encoder and a decoder. Both the encoder and decoder contain residual convolutional blocks inspired from Spatially Adaptive Normalization [35]. The source image F_s is first passed through an identity encoder to encode identity information. The output from the identity encoder is then concatenated with Enc_{dec} and

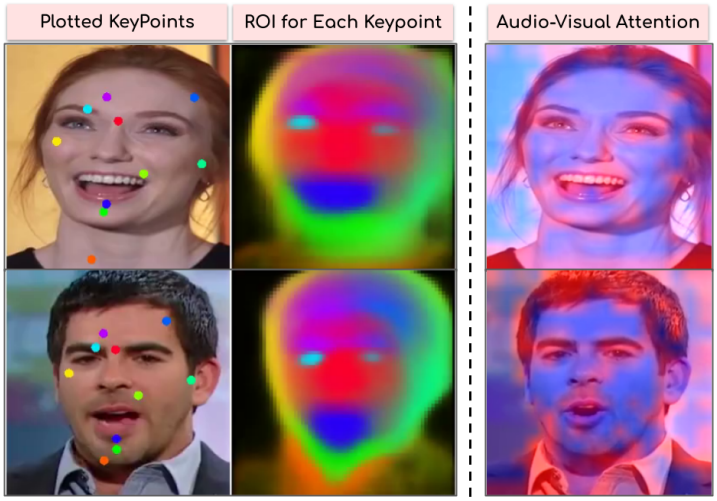

Figure 2.5 Illustration of keypoints detected (left), colour coded heatmap corresponding to each keypoint (centre) and the attention generated by our Audio-Visual Module (right). The ROI image shows that there are keypoints specific to the eye and mouth region. Attention image shows the important facial regions on which AVFR-Gan focuses.

finally passed through the matching decoder with appropriate skip connections between the encoder and decoder blocks. The final output from the generator is denoted by F_{gen}. Our generator produces the sharpest output compared to the current state-of-the-art, as shown in the subsequent sections.

2.2.5 Discriminator

To improve the quality of our generated outputs, we also employ a standard discriminator, which is trained in a GAN setup along with the rest of the network. Our discriminator M_{Disc}, consists of a stack of Conv2D layers each followed by either spectral normalization [32] or instance normalization [52]. Each convolution block is followed by a Leaky ReLU activation [30]. The discriminator predicts a real or fake label and is trained to maximize the following loss function L_{Disc} given in Equation 2.3.

$$\begin{aligned} \max_{M_{Disc}} L_{Disc} = \mathbb{E}_{x \sim p_{real}} \log M_{Disc}(x) + \\ \mathbb{E}_{F_{gen}} \log(1 - M_{Disc}(F_{gen})) \end{aligned} \tag{2.3}$$

2.2.6 Losses used to train the Generator

We use multiple loss functions similar to [44]. We use the $L1$ reconstruction loss between F_d and F_{gen}. We also use the LPIPs [61] perceptual similarity loss (denoted by L_{per}) to improve the perceptual quality of the generated outputs. Finally, we employ the equivarience constraints L_{eq} for generating consistent keypoints. Similar to the original FOMM paper, we use thin plate spline deformations to

generate keypoints and extend it to the jacobians as well. We refer the reader to [44] for information regarding these constraints. While training the generator we also minimize the discriminator loss given in Equation 2.3. Therefore, we present our final loss function, Equation 2.4.

$$\min_{M_{Gen}} L_{Gen} = ||F_d - F_{gen}||_1 + \\ L_{per} + L_{eq} + \mathbb{E}_{F_{gen}} \log(1 - M_{Disc}(F_{gen})) \tag{2.4}$$

2.2.7 Inference Setting

While we sample both F_s and F_d from the same video during training, our training strategy ensures that identity and motion information are well distilled. Therefore, our method allows for cross-identity face reenactment. During inference, we select a single image of a person as the source image F_s. Given a driving video of N frames, $V_{i...N}$, we pass each frame separately through our network along with F_s and the corresponding audio segment of V_i (denoted by A_i) to generate the final output as shown in Equation 2.5.

$$F^i_{Gen} = M_{Gen}(F_s, V_i, A_i), i \in 1...N \tag{2.5}$$

2.2.8 Implementation Details

In our experiments, we set $h = 256, w = 256$ and predict $K = 10$ keypoints for training all our models. The model is trained using the Adam optimizer[23] with a learning rate scheduler set at 60 and 90 epochs. The initial learning rate is set to be at 0.001. The training time taken by model on 4 NVIDIA RTX 3080Ti GPUs with a batch size of 10 is around 10 days. We train our model on the VoxCeleb [33] dataset, which contains 25 FPS videos. Thus, the 200ms audio window consists of 5 frames, of which the 3rd frame is selected as the driving frame F_d. Any other random frame from the same video is selected as F_s during training the network. We apply image transformations on some of the training images to make model more robust. We also use dataset repeater for increasing the size of dataset by a factor of 75 for VoxCeleb [33] dataset. The model is trained for 100 epochs.

2.3 Experiments and Results

We provide a comprehensive set of evaluations to measure the performance of our proposed method. We perform the quantitative assessment by following the standard benchmarks set by the previous works. We also perform extensive human evaluations to provide a qualitative assessment of the generated results.

2.3.1 Evaluation Set

We use the public test set of the VoxCeleb [33] dataset. The dataset contains videos of celebrities. All the videos are preprocessed to 256×256. The test set contains 465 number of videos of different identities making up a total of 76 minutes.

2.3.2 Evaluation Metrics

To provide an extensive evaluation of video reconstruction, we use several metrics to measure the performance of different works. We use the following metrics to measure various aspects of our generation. **L1**: It checks the average L1 distance between the generated and ground-truth video. **LMD**: Landmark Distance calculates the distance between detected keypoints of ground-truth and developed video using a pre-trained facial landmark detector[12]. Please note that this metric was denoted by Average Keypoint Distance in [44]. However, we renamed it Landmark Distance to avoid confusion with the keypoint detector module used in this work. **AED**: Average Euclidean Distance is used to evaluate the identity information. We use Openface[9] to find the feature vectors of generated and ground-truth video and then take the $L2$ distance between them. **PSNR**: Peak Signal to Noise Ratio is used to evaluate the reconstruction quality of the generated image compared to the ground truth image. **SSIM**: Structural Similarity Index evaluates the perceived changes in structural information of an image. We use it along with PSNR as it can also handle global illumination changes. **FID**: Fréchet Inception Distance is used to compare the distribution of generated images with the ground truth image using the features extracted from an InceptionV3 model [49]. **Sync**: Syncnet confidence score is used to measure the amount of lip sync [14].

2.3.3 Comparison with State-of-the-Art Methods

We compare our work with the current methods published for the same task. To have a fair comparison, we use the official pre-trained models of FOMM [2], MRAA [4], PC-AVS [6] and DA-GAN [1] from their respective open-source implementations. For Face-Vid2Vid, we use an unofficial implementation in [5]. All the pre-trained models and AVFR-GAN were trained on the same train split and evaluated on the test split of VoxCeleb[33] using two inference strategies defined below.

2.3.4 Same-identity Reenactment

We perform the face reenactment task where the source frame and the driving video are of the same person. In this setting, we take the first frame of any video as the source frame and consider the rest of the video as the driving video. The audio chunks corresponding to each driving frame are also fed to the network as input. In this case, we expect the generated output to be as close to the original video as possible. We can therefore calculate metrics like L1, LMD, PSNR, and SSIM, which requires ground truth. We also calculate AED, FID, and Sync metrics for the generated outputs from all the models. From Table 2.1, it is evident that our method outperforms all the other competing methods. The superior

Figure 2.6 Qualitative results on same-identity face reenactment. Upper row: Driving Video, Lower row: Generated Results

L1 and AED show that our model preserves identity information better. The improvement achieved by our model in terms of LMD indicates the improved structure of generated faces. Interestingly, our model generates improved eye movement in much more detail compared to the previous methods. We got state-of-the-art PSNR, SSIM, and FID scores, correlating with better visual quality. Finally, the sync quality achieved by our algorithm is superior to all the methods except PC-AVS, which performs slightly better in this metric.

	Same-id Reenactment							Cross-id Reenactment	
	L1↓	**PSNR↑**	**SSIM↑**	**FID↓**	**LMD↓**	**AED↓**	**Sync↑**	**FID↓**	**Sync↑**
FOMM[44]	0.046	28.890	0.740	11.04	1.294	0.142	5.17	11.93	3.17
Face-vid2vid [56]	0.062	29.160	0.690	11.47	1.620	0.153	4.96	10.81	4.19
MRAA [45]	0.040	23.351	0.64	11.36	1.280	0.135	3.10	15.61	3.96
PC-AVS [64]	0.081	23.750	0.620	14.32	1.843	0.180	**6.76**	16.78	**6.39**
DA-GAN [18]	0.036	31.220	0.804	9.10	**1.278**	0.129	5.01	9.40	4.71
AVFR-GAN (Ours)	**0.034**	**32.20**	**0.824**	**8.48**	1.280	**0.127**	5.45	**9.05**	4.99

Table 2.1 Comparison with state-of-the-art methods on Same-identity Reenactment and Cross-identity reenactment on VoxCeleb[33] dataset. ↑ indicates larger is better, and ↓ indicates smaller is better.

2.3.5 Cross-identity Reenactment

In this setting, we take a driving video for a different identity and animate a source image. The audio from the driving video is also given as input to the network, as usual. However, since the generated output does not mimic any specific ground truth, we use metrics that do not directly need the same. We use FID, which measures the distance between real and generated distributions and does not require one-to-one ground truths. We also use Sync to measure the quality of the lip sync in the generated video.

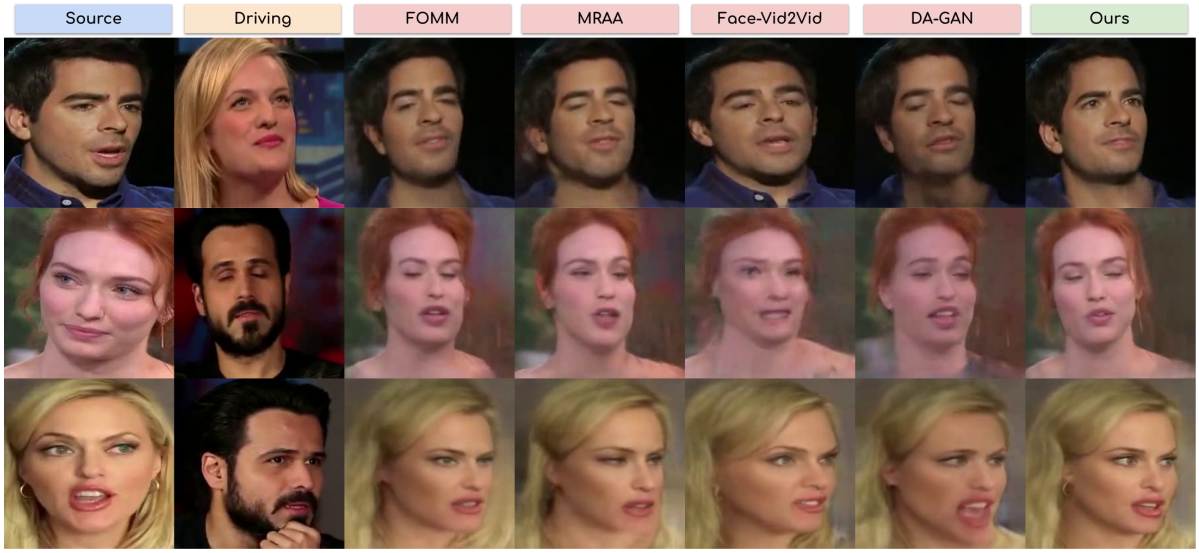

Figure 2.7 Qualitative comparison on Cross-identity reenactment. Our method gives fewer artifacts, preserves facial structure and handle motion in a better way.

As seen in Table 2.1, we achieve the best FID results and the second-best results in sync trailing only to PC-AVS.

2.3.6 Human Evaluations

Since our algorithm generates outputs directly meant for human consumption, we perform extensive human evaluations to ascertain the quality of the generations from our model from a human's perspective. We perform a study enrolling 20 users. Each user is shown generated samples from the state-of-that-art method along with Ours. The users are also shown the source image and the driving video. We select 30 samples from Cross-identity generations. Our user study shows corresponding results from each algorithm side by side, along with the source image and the driving video. The users are asked to rate each generated output based on three characteristics. The users rate the quality of 1. Head pose matching the driving videos, 2. Expressions matching the driving videos, 3. Identity preservation between the source image and the generated videos. The ratings are between 1 to 5, where 1 corresponds to the worst and 5 corresponds to the best. As seen in Table 2.2, our model consistently yields better results across all the criteria. Our model can enact a better head pose and match expressions of the driving video while preserving the source identities.

2.4 Ablation Study

Our proposed approach comprises the addition of several key priors and the use of a better image generator. We check the contribution of each of these novel blocks in this section. For setting a baseline

	HPMS↑	EMS↑	IPS↑
FOMM[44]	3.40	3.16	2.80
Face-vid2vid [56]	3.70	3.12	2.66
MRAA [45]	3.26	3.06	2.50
PC-AVS [64]	1.58	1.64	1.92
DA-GAN [18]	3.98	3.82	3.10
AVFR-GAN (Ours)	**4.56**	**4.22**	**3.94**

Table 2.2 User Study quantitative comparison. 'HPMS' represents Head Pose Matching Score, 'EMS' represents Expression Matching Score and 'IPS' represents Identity Preservation Score. ↑ shows higher is better.

(very similar to FOMM), we remove Face Mesh, Face Segmentation, Audio Encoders, and used a basic CNN-based decoder architecture[44, 18, 56]. We add one module at a time to this baseline and train them on the same train-test split. We first add only face mesh and face segmentation to the baseline. We separately also check the effect of adding audio to the baseline. Finally, we combine the structural priors and audio to train a model without the novel identity-aware generator. We calculate SSIM, FID, and Sync metrics and report them in Table 2.3.

	SSIM↑	FID↓	Sync↑
Baseline	0.74	11.04	5.17
+ Structural Prior	0.801	8.98	5.19
+ Audio Prior	0.79	8.69	**5.48**
+ IAG	0.812	8.51	5.13
AVFR-GAN	**0.824**	**8.48**	5.45

Table 2.3 Ablation Study. The baseline represents the model without face mesh, segmentation, audio, and identity-aware decoder. '+ Structural Prior' represents Baseline with face segmentation and face mesh. '+ Audio Prior' represents Baseline with Audio encoders. '+ IAG' represents Baseline with Identity Aware Generator. ↑ indicates larger is better, and ↓ indicates smaller is better.

As we observe clearly, the structural priors improve the SSIM significantly over baseline, while audio improves the lip sync quality. We also observe that audio improves the visual quality (measured using FID) of the generations marginally. Finally, the identity-aware face generator gives a significant boost in terms of visual quality improvement.

2.5 Applications

Our work opens up several applications in the digital industry. Our method can revolutionize multiple industries. We can potentially replace recording famous celebrities in a studio environment costing thousands of dollars; we can animate a single picture of them based on home-recorded driving videos. Similar advances can also be made in the education sector, where online lectures are an integral part of education. News readers can reduce their commute and present news from the comfort of their homes by animating their characters. We can also make video calls simpler in more than one way. We can replace the live video feed with a generated one reducing zoom fatigue. More importantly, this can lead to huge bandwidth reduction due to the compact keypoint-based representation, as already noted in [56].

2.5.1 Low-bandwidth Video Conferencing

Face reenactment methods can be easily extended for video compression. In the case of a video call between a sender and a receiver, we can first send a single high-resolution frame between the two and follow it up with sending keypoints detected by the keypoint detector for each frame. Our model can then generate the output frames at the receiver's end by considering the high-resolution frame as the source and keypoints from each of the driving frames, similar to the results shown in Figure 2.6. The 10 keypoints each consist of x and y coordinates and four jacobians, all of which are represented as float values. Therefore, the total bits required to represent a 256×256 frame using FP16 representation is $10 \times 6 \times 16 = 960$ bits. Therefore, the Bits-per-Pixel(BPP) achieved by our model is $\frac{960}{256 \times 256} = 0.014$. We use the latest H.266 codec [11] released in September of 2021 and compress the VoxCeleb test set at the same BPP. While the results generated by our algorithm achieve a FID of 8.48, the H.266 lags by a large margin at 58.32. This indicates the superior quality of the results generated using AVFR-GAN and provides a proof-of-concept for compressing video calls in future work.

2.6 Conclusion

In this work, we propose a novel face reenactment network, Audio-Visual Face Reenactment GAN. Our network uses audio-visual cues to reenact a source image according to a driving video. We provide the network with additional structural priors and speech to improve lip synchronization. The final output quality also benefits from a novel identity-aware generator. We believe these works will benefit and reduce manual effort in professional content creation.

2.7 Ethical Concerns

The improvement in the quality of the generative networks has also led to concerns over its potential misuse. We, therefore, urge the users of any such works to use it ethically. We also encourage users to clearly mark the generated videos with a watermark.

Chapter 3

Compressing Video Calls using Synthetic Talking Heads

3.1 Introduction

As we progress through the 21st century, the world continues to grow digitally and becomes more connected than ever! Video calls are a big part of this push and are a staple form of communication. The pandemic in 2020 led to a massive reduction in social interaction and fast-tracked its adoption. Universities and schools were forced to use video calls as the primary means of teaching, while for many, video calling remained the only way to connect with friends and family. While the number of video calls will continue to rise in the future, increasing bandwidth is a daunting task. Incidentally, over half the world's countries do not even have 4G services [1]! Therefore, introducing video compression schemes to reduce the bandwidth requirement is a need of the hour.

3.1.1 Traditional Video Compression Techniques

Compressing video information has fascinated researchers for nearly a century. The first works dealt with analog video compression and were released in 1929 [22]. A significant breakthrough in modern video compression was achieved by [31] using a DCT-based compression technique leading to the first practical applications. This was followed by the widely adopted H.264 [57] and H.265 [3] video codecs, which remain the most popular in industrial applications. The most recent codec to be released is H.266 [11]. However, we do not compare our work with H.266 due to the lack of availability of open-source implementations. Deep learning-based video compression techniques like [27, 43, 34, 40] have also been prevalent in the recent past. These techniques use autoencoder-like structures to encode video frames in a bottlenecked latent space and generate it back on the receiver's end. While such approaches have proven their effectiveness in multiple situations, they are generic and do not consider the high-level semantics of the video for compression.

[0] Webpage for the Paper: https://cvit.iiit.ac.in/research/projects/cvit-projects/talking-video-compression

[1] https://en.wikipedia.org/wiki/List_of_countries_by_4G_LTE_penetration

3.1.2 Talking Head Video Compression

Video calls, on the other hand, encompass a specific class of videos. They primarily contain videos of speakers and are popularly known as talking head videos. The inherent semantic information present in a talking head video involving the face structure, head movements, expressions on the face, etc., has long interested researchers in developing compression schemes targeted towards such specialized videos. Techniques like [34] transmit 68 facial landmarks for each frame, which synthesize the talking head at the receiver's end. In 2021, Wang et al. [56] proposed using face reenactment for video compression. They used 10 learned 3D keypoints instead of pre-defined face landmarks to represent a face in their work leading to significant compression. Each learned keypoint contains information regarding the structure of the face, rotation, translation, etc., and helps to warp a reference frame.

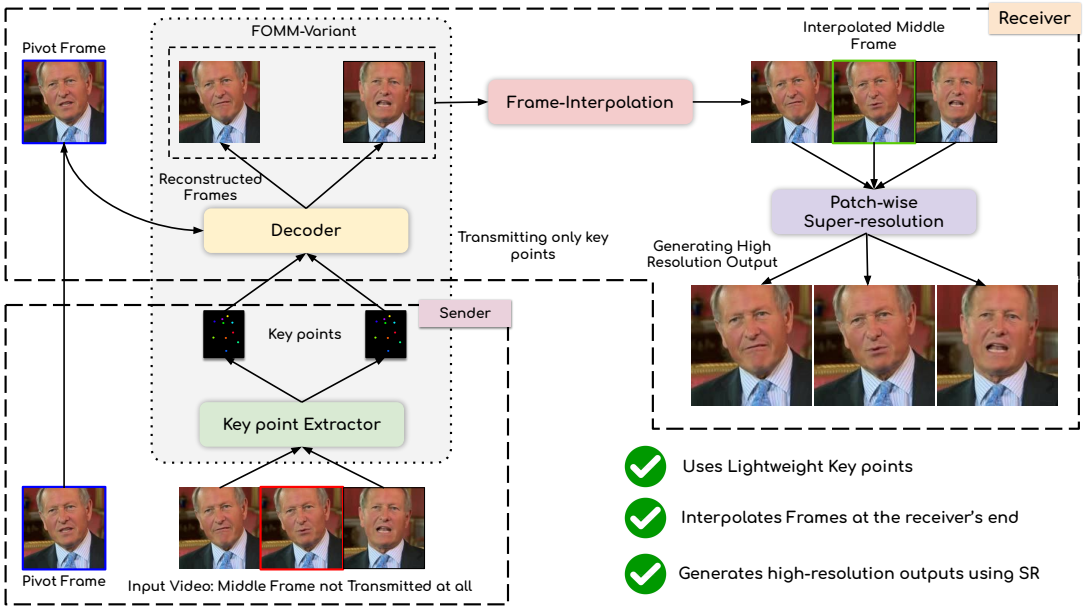

Figure 3.1 We depict the entire pipeline used for compressing talking head videos. In our pipeline, we detect and send keypoints of alternate frames over the network and regenerate the talking heads at the receiver's end. We then use frame interpolation to generate the rest of the frames and use super-resolution to generate high-resolution outputs.

3.1.3 Our Contributions

We explore this concept further in this work and propose several novel improvements. We first send a high-resolution frame (pivot frame) at the start of the video calls. For the rest of the frames, we use a modified version of [44] to detect keypoints in each of them and transmit them to the receiver. The keypoints are then used to calculate a dense flow that warps the pivot frame to recreate the original

video. While [44, 56] used 24 bytes to represent a single keypoint, we further propose to reduce this requirement to only 8 bytes. Next, we use a novel talking head frame-interpolater network to generate frames at the receiver's side. This allows us to send keypoints from fewer frames while rendering the rest of the frames using the interpolater network. We use a patch-wise super-resolution network to upsample the final outputs to arbitrary resolutions, significantly improving the generation's quality. In a lengthy video call sending a single pivot frame at the start of the video may lead to inferior results on significant changes in the background and head pose. Therefore, we also propose an algorithm to adaptively select and send pivot frames negating the effects of such changes. Overall, our approach allows for unprecedently low Bits-per-Pixel (BPP) value (bits used to represent a pixel in a video) while maintaining usable quality. We refer the reader to check our project web-page for numerous example results from our approach.

3.2 Background: Synthetic Talking Head Generation

Our work revolves around synthetic talking head generation. Therefore, we survey the different types of talking head generation works prevalent in the community. Talking head generation was first popularized in works like [48, 13, 19, 24, 36] which attempted to generate only the lip movements from a given speech. These works were effective for solutions that required preserving the original head movements in a talking head video while changing only the lip synchronization to a new speech. A separate class of works [65, 62, 54, 64] tried to generate the talking head video directly from speech without additional information. While these works can also potentially find their usage in video call compression, the head movements in the generated video do not match those of the original one, limiting its usage!

3.2.1 Face Reenactment

In face reenactment, a source image is animated using the motion from a driving video. The initial models for this class of works were speaker-specific [10, 58]. These models are specifically trained on a single identity and cannot generalize to different individuals. On the other hand, speaker agnostic models [44, 56, 64, 8] are more robust. They require a single image of any identity and a driving video (need not have the same identity) to generate a talking head of the source identity following the driving motion. We find face reenactment works to be well suited for talking head video compression. We propose to use the inherent characteristic of the problem and send a single high-quality frame that can be animated by the rest of the video at the receiver's end to generate the final output. The reenactment is driven by landmarks, feature warping, or latent embeddings. First-Order-Motion-Model (FOMM) proposed by Siarohin et al. [44] uses self-learned keypoints to represent the dense motion flow of driving video. Each keypoint consists of the coordinates and Jacobians representing the local motion field between the source image and the driving video. A global motion field is then interpolated from the

local motion field, and the source image is warped using the estimated motion field. Wang et al. [56] too similarly learn a motion field between the source image and the driving video. However, in their case, the keypoints are 3-dimensional, containing additional rotation and translation information.

3.3 Methodology

3.3.1 Overview of the Technique

As discussed previously, we start the video call by sending a pivot frame from the sender to the receiver and then animate it using the rest of the frames in the video call. We use a variation of the FOMM [44] model for achieving this task. Each keypoint in the FOMM model consists of 2D coordinates and Jacobians that possess additional region-specific information. Through experiments, we realize that Jacobians play an essential role in modeling complex motions. However, video calls are frontal face videos with relatively fewer head motions. Thus, we reduce the bits required to store each keypoint by removing Jacobians and transferring only the coordinates of keypoints for each frame over the network. We also propose a talking face frame interpolation algorithm inspired by [59] to generate intermediate frames in the video, reducing the number of frames for which keypoints needs to be transferred. Finally, we use a patch-based super-resolution network to generate arbitrary high-resolution outputs. To counter the instability caused by the removal of Jacobians when encountered with large head movements, we formulate a simple algorithm to send and replace pivot frames intermittently based on the difference in head pose and background between the current pivot frame and the driving video at the sender's side.

3.3.2 Formalizing the Compression Strategy

Let us assume we have $n + 1$ frames at the sender's end in our setup. We denote the frames by $f_0, f_1, f_2, ..., f_n$. We pick f_0 as our first pivot frame and transmit it to the receiver. The pivot frame is denoted by f_{pv}. We then pick alternate frames $f_1, f_3, f_5, ...$ and pass them through the learned keypoint detector of our FOMM-variant. The detected keypoints are denoted by $p_1, p_3, p_5, ...$ At the receiver's end, the decoder from the FOMM-variant uses the transmitted keypoints and f_{pv} to generate $f'_1, f'_3, f'_5, ...$ We use our frame-interpolater network to generate the intermediate frames, $f'_2, f'_4, ...$ We then apply our patch-based super-resolution network on all the frames on the receiver's end, $f'_1, f'_2, f'_3, f'_4, f'_5, ...$ to generate higher resolution versions of the same. Finally, for a significant difference in the head pose or background between the pivot frame f_{pv} and the i^{th} frame, f_i, we transmit f_i to the receiver making it the new pivot frame.

3.3.3 Modifying the First-Order-Motion-Model

We take inspiration from First Order Motion Model for Image Animation [44] for reenacting a face at the receiver's end. While the original version of FOMM [44] was not designed for compression in video calls, we re-purposed it for the task at hand and built a refined version of the model. In the original model, a keypoint detector detects 10 keypoints along with Jacobians in the neighborhood of each keypoint. The model detects these keypoints in both the source, and driving frames and a motion field is calculated between corresponding keypoints between the two frames. The dense flow calculated from this motion field is then used to warp the source frame using a decoder generating the final output. All the network components like the generator and the keypoint detector, are trained end-to-end allowing the keypoint detector to extract keypoints best suited for generating the most accurate result.

In this work, we remove the requirement of Jacobians and instead train a version of FOMM[2] requiring only coordinates of the keypoints to reconstruct a frame. This is motivated directly by our use-case of video call compression. Jacobians are 2×2 integer matrices for each of the 10 keypoints. By removing the Jacobians, we can represent a frame with only the (x, y) coordinates of the 10 keypoints saving a large amount of bandwidth. We find that removing the Jacobians does not affect the performance of our network on frontal-facing videos that are most encountered during a video call. We follow the same training methodology and losses as stated in [44] to train this modified version of the FOMM model. Once the model is trained, the keypoint extractor is deployed at the sender's end while the decoder part of the network is deployed at the receiver's end. At any point of the video call, the current pivot frame acts as the source frame, and the keypoints from the subsequent frames (which serve as the driving video) are used to warp the pivot frame animating it. A graphical representation of the process is given in Figure 3.1.

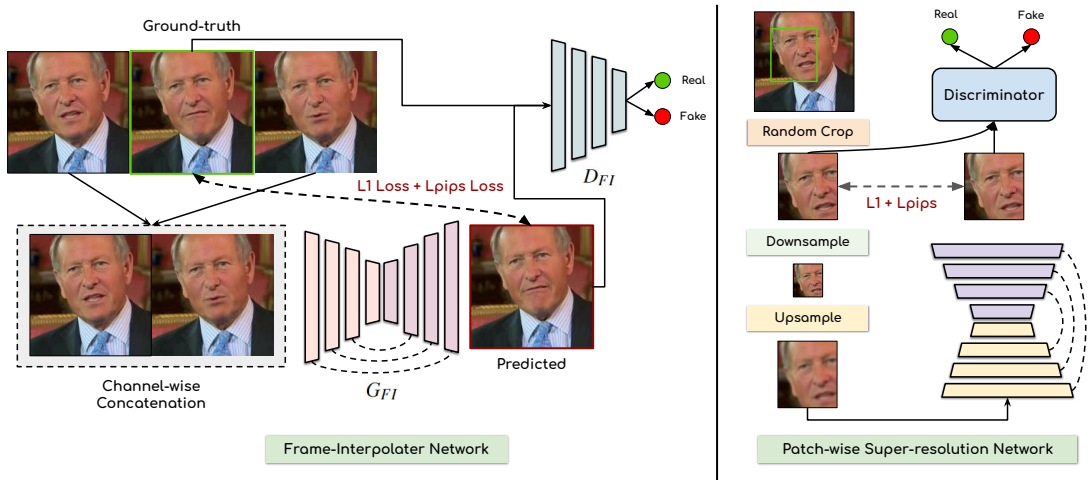

Figure 3.2 We depict the architectures of the frame-interpolation network and the Patch-wise Super-resolution Network.

[2] https://github.com/AliaksandrSiarohin/first-order-model

3.3.4 Frame Interpolation at the Receiver's End

To further reduce the bandwidth requirements and improve the compression ratio, we introduce a frame interpolation network motivated by recent advances in Face Enhancement works [59]. We use a standard GAN [17] architecture consisting of a Generator G_{FI} and a Discriminator D_{FI}. To ensure lesser model complexity, we decide against using 3D convolution layers and use standard 2D convolution in both networks. As shown in Figure 3.2, we train this network on videos in a self-supervised manner. During training, we sample consecutive windows of three frames, $\{v_i, v_{i+1}, v_{i+2}\}$ in a video. We then concatenate v_i and v_{i+2} channel wise creating the input to G_{FI}. The generator is tasked to generate v_{i+1}, which is used as the ground truth. The discriminator D_{FI} is trained to maximize the loss function given in Equation 3.2. We calculate three losses for the generator: the L1 reconstruction loss, the LPIPS [61] perceptual loss and finally, the discriminator's loss to train the generator G_{FI}. The loss and optimization functions used to train the generator are defined in Equation 3.3.

$$v_{i+1}^{gen} = G_{FI}(v_i||v_{i+2}) \tag{3.1}$$

$$\max_{D_{FI}} L_{disc}(D_{FI}, G_{FI}) = \mathbb{E}_{real}[\log D_{FI}(v_{i+1})] + \mathbb{E}_{fake}[\log(1 - D_{FI}(v_{i+1}^{gen}))] \tag{3.2}$$

$$\min_{G_{FI}} L_{gen} = L_{disc} + ||v_{i+1} - v_{i+1}^{gen}||_1 + LPIPS(v_{i+1}, v_{i+1}^{gen}) \tag{3.3}$$

3.3.5 Patch-wise Super-resolution Network

While users in the past were used to grainy webcam videos, the quality of the front cameras of cell phones, webcams, and other types of cameras has improved significantly. Maintaining the quality of the video calls is thus of utmost importance! Therefore, we train a GAN to enhance the quality and resolution of the generations. The architecture of this network closely resembles the frame interpolation network and is trained in a self-supervised manner. We also want our network to be able to arbitrarily super-resolve the output to any resolution. Therefore, instead of training the network on a fixed resolution of images, we train it using $k \times k$ cropped patches from the images. During training, we randomly sample frames from videos and take $k \times k$ random crops from them. We then bicubically downsample the patches by a random factor between $2 - 6$. We then create the input to the network by bicubically upsampling the downsampled patches back to their original resolution, i.e., $k \times k$. The network is tasked to remove the blur in the input patches introduced by bicubic upsampling. This network is also trained following a similar strategy to Equations 3.2 and 3.3. During inference, we bicubically upsample the whole image to any desired resolution. Using a sliding window, we then divide the image into $k \times k$ patches. Our network then super-resolves each patch separately to generate sharp, high-resolution outputs. A pictorial representation of the architecture is given in Figure 3.2.

3.3.6 Adaptive Pivot Frame Selection

Due to the lack of Jacobians in our keypoints, our network sometimes falters when faced with massive changes in head pose. While this is unlikely to happen in a dataset, it can be pretty standard when tested in a real-world video calling setup. We, therefore, propose a simple algorithm to adaptively change the pivot frame based on the difference in head pose and change in background. To detect the change in the head pose, we use an open-source codebase[3] and calculate the yaw, roll, and pitch in the pivot frame F_{pv} and any current frame F_i whose keypoints are to be transmitted. We empirically find thresholds of $\gamma_{yaw}, \gamma_{roll}, \gamma_{pitch}$ based on which we change the pivot frame to the current frame, i.e., $F_{pv} = F_i$ in case of a major shift. We also use Mediapipe [29] library to generate face segmentation masks to detect the background portions of a frame. We then use a pre-trained VGG-19 [46] network to generate embeddings for the backgrounds of both F_{pv} and F_i. A simple euclidean distance d_{bg} is calculated to determine the amount of background change. If a significant background change is determined using an empirical threshold, the pivot frame is replaced by the current frame.

3.3.7 Dataset & Implementation Details

We train our networks on the train set from the VoxCeleb dataset [33] with a learning rate of 0.001 using the Adam optimizer [23]. The resolution of all the videos is kept at 256×256 during training. The patch size used for training the Super-Resolution network is set to 64×64. During inference, we apply $2\times$ super-resolution achieving 512×512 resolution on the final generated videos. The thresholds that we select after experimentation are $\gamma_{yaw} > 15°, \gamma_{roll} > 15°, \gamma_{pitch} > 15°$. We select $d_{bg} > 0.05$ as the threshold for considering backgrounds as different. Please note that breaching either of the thresholds is considered a criterion for replacing the pivot frame.

3.4 Experiments and Results

3.4.1 Comparable Methods & Metrics used

We compare our work with two of the most famous and versatile video compression techniques, H.264 [57] and H.265 [3]. We vary the Constant Rate Factor (CRF) in both methods and generate results in various settings. We also compare our method with the original FOMM [44] and Face-Vid2Vid [56]. All the networks were trained on the same training set for a fair comparison. Apart from other comparable works, we also create baselines by removing different modules from our proposed pipeline. We report three visual quality metrics to measure the visual quality; PSNR, SSIM, and FID [49]. We also report the BPP to measure the compression level for each method. We use the test set from the VoxCeleb [33] for benchmarking all the approaches. Please note that the BPP is calculated based on 512×512 as the final resolution for all the methods.

[3] https://github.com/WIKI2020/FacePose_pytorch/

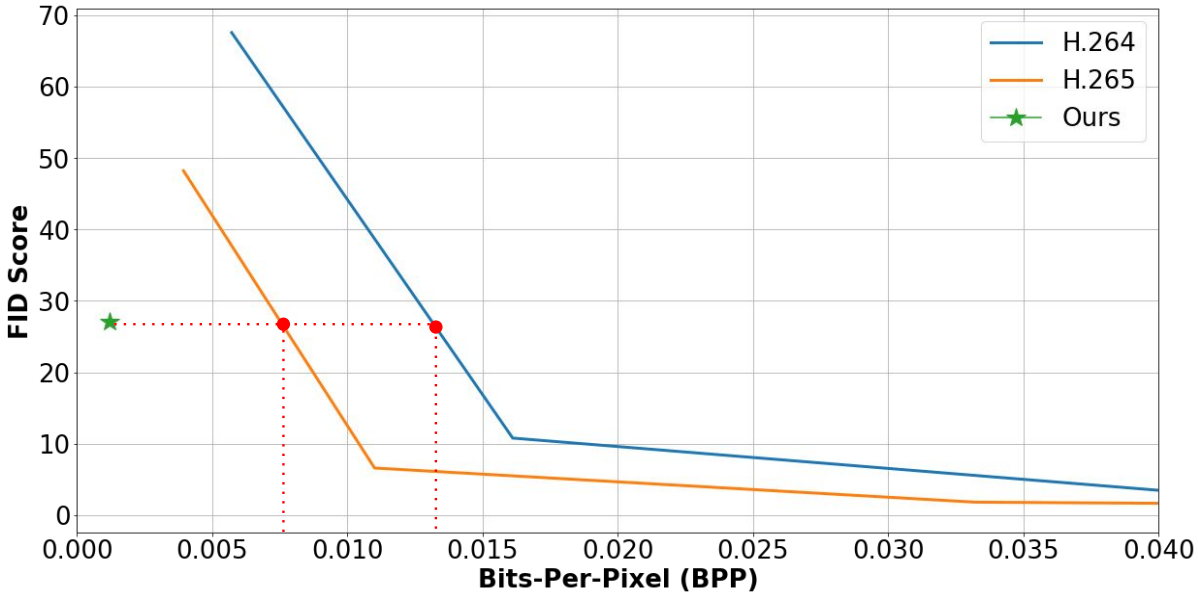

Figure 3.3 We calculate the change of FID with reducing compression, i.e., increasing BPP. We find that the FID score achieved by our network can only be achieved at a far lower level of compression for both H.264 and H.265.

3.4.2 Quantitative Results

We report quantitative scores in Table 3.1. For H.264 and H.265, we use the Constant Rate Factor (CRF) $= 51$ to generate the results at the least BPP possible. As we can see, even at the maximum compression levels of H.264 and H.265, our approach achieves less than $\frac{1}{3}$rd the BPP while generating visually appealing results. We also plot the variation of FID with changing BPP for H.264 and H.265 in Figure 3.3. We find that the FID achieved by our approach is only achievable at $5\times$ more BPP for H.265 and $10\times$ more BPP for H.264. We also achieve a much lower BPP than the original FOMM and Face-Vid2Vid [56] but can maintain quality. Finally, we stack up different modules from our approach one by one and then compare the results achieved in each combination. As observed in Table 3.1, adding each module in our approach reduced BPP while maintaining the quantitative metrics at a similar level.

3.4.3 Qualitative Results

We show multiple qualitative comparisons in Figure 3.4. As we can see, our method generates sharper results when compared to the prevalent compression techniques. Furthermore, we also ran our pipeline on real video calls publicly available on YouTube. These videos are far longer than the ones present in the test set. Figure 3.5 shows the impact of the adaptive pivot frame selection module and generates better outputs than the ones generated without using it.

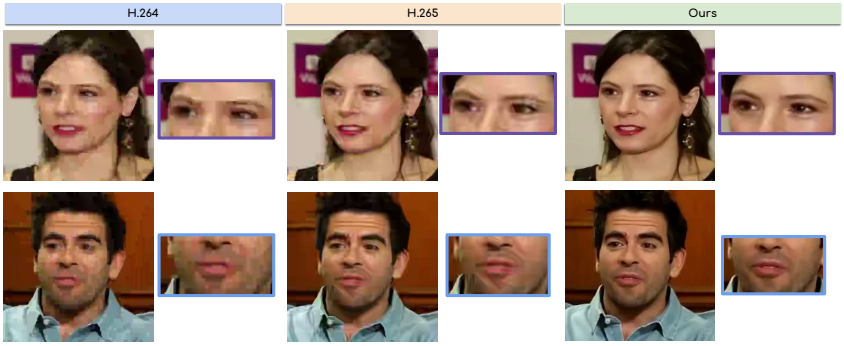

Figure 3.4 We compare our results with H.264 and H.265. Our method generates far sharper images with much less data.

Method	BPP↓	PSNR↑	SSIM↑	FID↓
FOMM [44]	0.029	26.81	0.79	26.81
Face-vid2vid [56]	0.016	24.37	0.80	25.19
H.264 [57]	0.0057	30.25	0.78	67.54
H.265 [3]	0.0039	**30.74**	0.80	48.20
keypoint Only (Ours)	0.0097	24.48	0.78	**20.58**
keypoint + Frame Interpolation (Ours)	0.0048	24.21	0.78	23.03
keypoint + Frame Interpolation + SR (Ours)	**0.0012**	26.73	**0.81**	27.81

Table 3.1 We compare our method with other state-of-the-art architectures as well as widely used techniques like H.264 and H.265. We observe our method to consistently have decent visual quality at much lower BPP.

3.5 Ablation Studies

We perform several ablation studies to understand the effectiveness of different hyperparameters we choose to achieve the best performance. We keep the pivot frame constant for all the experiments if not mentioned otherwise.

While we train our network to interpolate a single frame at a time, it can be easily used to interpolate more than one frame during inference by using the generated frames as input. We interpolate 2 frames and 3 frames at a time and report the scores in Table 3.2. As we see, interpolating more number of frames improves the BPP significantly but also leads to some loss in performance. However, the performance still remains within usable range and thus can be explored in cases where even more compression is required.

We also vary the $k \times k$ patch size used for training the super resolution network. We super resolve the output 2 times using different sized patches and report our findings in Table 3.3.

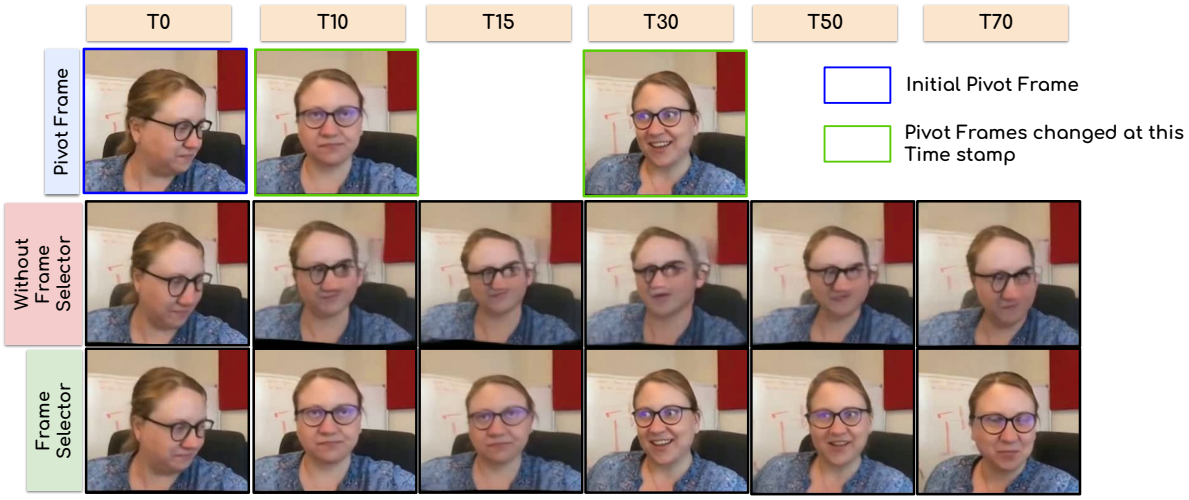

Figure 3.5 We use a lengthy real-world video call and mark frames for various time stamps (frame numbers in this case). Our goal is to understand the effect of the adaptive frame selector. In the above example, we select newer pivot frames at T10 and T30 owing to major head pose changes. This allows our network to continue generating sharp results.

#Int. frames	BPP↓	PSNR↑	SSIM↑	FID↓
0	0.0024	29.28	0.83	9.10
1	0.0012	28.49	0.82	12.42
2	0.0008	28.23	0.81	12.77
3	0.0006	27.73	0.79	12.92

Table 3.2 We vary the number of frames that are interpolated and report the scores achieved.

We select different thresholding parameters for our frame selection algorithm. We report the scores in Table 3.4. Using adaptive frame selection increases BPP due to the transfer of multiple pivot frames. We calculate the metrics using different thresholds for all the γ variables and d_{bg}. On an average, at our default setting of $\gamma > 15^{\circ}$ and $d_{bg} > 0.05$, we find a pivot frame change every 10 seconds. The shift is much less common on the bigger thresholds.

3.6 Conclusion

In this work, we propose to use the high-level semantics of a talking head video to create extreme compression schemes which can revolutionize video calling. Our work uses compact keypoints to transmit information about the talking head in each video frame. We also propose a frame interpolation network followed by super-resolution to arbitrary resolutions. Finally, a pivot frame selection algorithm

Patch Size	PSNR↑	SSIM↑	FID↓
128×128	27.37	0.81	19.12
64×64	27.47	0.80	20.16
32×32	26.18	0.79	20.89
16×16	25.34	0.78	21.17

Table 3.3 We vary the size of the patches taken by our SR network and report the scores in this table.

Method	BPP↓	PSNR↑	SSIM↑	FID↓
$d_{bg} > 0.05$	0.0029	27.37	0.81	19.12
$d_{bg} > 0.06$	0.0021	25.93	0.77	20.74
$d_{bg} > 0.07$	0.0016	24.72	0.73	23.17
$\gamma > 15^{\circ}$	0.0049	25.28	0.75	22.46
$\gamma > 30^{\circ}$	0.0031	24.02	0.71	26.63
$\gamma > 45^{\circ}$	0.0018	23.76	0.70	26.93

Table 3.4 We select different thresholds for our adoptive frame selection algorithm. Please note that γ here represents thresholds for all the three γ-values.

is used for long video calls helping our compression technique continue generating high-quality videos. In the future, we believe solving other aspects like ensuring its application on edge devices will be a prospective task.

Chapter 4

Conclusions

In this thesis, we explore the idea of face reenactment and talking head generation. We proposed an audio-visual face reenactment network that creates high-quality talking heads. We demonstrated the effectiveness of using audio-conditioned features to create better lip-sync and mouth movement. We also show the effectiveness of using face priors in the form of face mesh and segmentation mask to improve keypoint detection. Using a novel identity-aware generator helps reduce flickers and artifacts from the generated video. Our network performs better than the current state-of-the-art methods in terms of qualitative and quantitative comparison. We further explore the practical applications of our work, specifically video calling. We proposed a carefully designed pipeline with frame-interpolation, patch-based super resolution network and an adaptive frame selector network. The proposed pipeline gives promising results by achieving comparable visual quality like H.265 and H.265 while using 5x less bandwidth.

Our work opens a lot of practical applications and future research directions. It can be extended to a full-body reenactment, where the action of a single source image of a person can be modeled using a full-body driving video. This further leads to creating visual storytelling for not only faces but for the full body. Hence, making it possible to create an entire movie without any additional expensive vfx studio. One other interesting direction to work on is introducing temporal consistency in the generation process. Instead of creating one-to-one motion model using two images, the temporal information of a video can be used to model the motion better. The current work has a limitation in manually manipulating certain aspects of the face, like modifying emotions. Combining the current work with the editing capabilities of diffusion models can also be promising.

Our work has some shortcomings, especially in modeling the motion of non-face regions, such as the torso. It also fails to model the motion of accessories like a hat and generally treats them as a background. There are also some limitations in generating talking heads of animated characters which do not closely resemble human face structures. We believe future work can solve these problems and make the pipeline more robust.

We understand that there is a possibility of misusing the research to create fake news and identity theft. We strictly urge that users should follow ethical guidelines and use a watermark for every generated video, clearly showing that it is reenacted.

Related Publications

- **Madhav Agarwal**, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, C V Jawahar. Audio-visual face reenactment. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pages 5178–5187, January 2023.
- **Madhav Agarwal**, Anchit Gupta, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, C V Jawahar. Compressing video calls using synthetic talking heads. In 33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022. BMVA Press, 2022.
- Sai Niranjan Ramachandran, **Madhav Agarwal**, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, C V Jawahar. Understanding the Generalization of Pretrained Diffusion Models on Out-of-Distribution Data. IEEE/CVF International Conference on Computer Vision (ICCV), 2023. (Under Review) *(Not a part of thesis)*
- **Madhav Agarwal**, Ajoy Mondal, C.V. Jawahar. CDeC-Net: Composite Deformable Cascade Network for Table Detection in Document Images. 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 2021, pp. 9491-9498, doi: 10.1109/ICPR48806.2021.9411922. *(Not a part of thesis)*
- Ajoy Mondal, **Madhav Agarwal**, C.V. Jawahar. Dataset Agnostic Document Object Detection. Pattern Recognition, 142:109698, 2023. *(Not a part of thesis)*

Bibliography

[1] Depth-aware generative adversarial network for talking head video generation. https://github.com/harlanhong/CVPR2022-DaGAN1. 14

[2] First order motion model for image animation. https://github.com/AliaksandrSiarohin/first-order-model. 14

[3] High efficiency video coding. 19, 25, 27

[4] Motion representations for articulated animation. https://github.com/snap-research/articulated-animation. 14

[5] One-shot free-view neural talking head synthesis. https://github.com/zhanglonghao1992/One-Shot_Free-View_Neural_Talking_Head_Synthesis. 14

[6] Pose-controllable talking face generation by implicitly modularized audio-visual representation. https://github.com/Hangz-nju-cuhk/Talking-Face_PC-AVS. 14

[7] M. Agarwal, A. Gupta, R. Mukhopadhyay, V. Namboodiri, and C. Jawahar. Compressing video calls using synthetic talking heads. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022. 4

[8] M. Agarwal, R. Mukhopadhyay, V. P. Namboodiri, and C. V. Jawahar. Audio-visual face reenactment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5178–5187, January 2023. 4, 21

[9] B. Amos, B. Ludwiczuk, M. Satyanarayanan, et al. Openface: A general-purpose face recognition library with mobile applications. *CMU School of Computer Science*, 6(2):20, 2016. 14

[10] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh. Recycle-gan: Unsupervised video retargeting. In *Proceedings of the European conference on computer vision (ECCV)*, pages 119–135, 2018. 21

[11] B. Bross, Y.-K. Wang, Y. Ye, S. Liu, J. Chen, G. J. Sullivan, and J.-R. Ohm. Overview of the versatile video coding (vvc) standard and its applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):3736–3764, 2021. 6, 18, 19

[12] A. Bulat and G. Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1021–1030, 2017. 14

[13] J. S. Chung, A. Jamaludin, and A. Zisserman. You said that? In *British Machine Vision Conference*, 2017. 2, 6, 7, 10, 21

[14] J. S. Chung and A. Zisserman. Lip reading in the wild. In *Asian conference on computer vision*, pages 87–103. Springer, 2016. 14

[15] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1

[16] G. Fauville, M. Luo, A. Queiroz, J. Bailenson, and J. Hancock. Zoom exhaustion & fatigue scale. *Computers in Human Behavior Reports*, 4:100119, 2021. 5

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 1, 24

[18] 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*, pages 3397–3406, 2022. 2, 6, 7, 8, 11, 15, 17

[19] A. Jamaludin, J. S. Chung, and A. Zisserman. You said that?: Synthesising talking faces from audio. *International Journal of Computer Vision*, 127(11-12):1767–1779, 2019. 21

[20] X. Ji, H. Zhou, K. Wang, W. Wu, C. C. Loy, X. Cao, and F. Xu. Audio-driven emotional video portraits. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14080–14089, 2021. 2, 6, 7

[21] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 1

[22] R. D. Kell. Improvements relating to electric picture transmission systems, 1929. 19

[23] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 13, 25

[24] P. KR, R. Mukhopadhyay, J. Philip, A. Jha, V. Namboodiri, and C. Jawahar. Towards automatic face-to-face translation. In *Proceedings of the 27th ACM international conference on multimedia*, pages 1428–1436, 2019. 2, 7, 10, 21

[25] L. Li, S. Wang, Z. Zhang, Y. Ding, Y. Zheng, X. Yu, and C. Fan. Write-a-speaker: Text-based emotional and rhythmic talking-head generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1911–1920, 2021. 2, 6, 7

[26] D. Lorenz, L. Bereska, T. Milbich, and B. Ommer. Unsupervised part-based disentangling of object shape and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10955–10964, 2019. 6

[27] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao. Dvc: An end-to-end deep video compression framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11006–11015, 2019. 19

[28] Y. Lu, J. Chai, and X. Cao. Live speech portraits: real-time photorealistic talking-head animation. *ACM Transactions on Graphics (TOG)*, 40(6):1–17, 2021. 6, 7

[29] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019. 9, 25

[30] A. L. Maas. Rectifier nonlinearities improve neural network acoustic models. 2013. 12

[31] G. Mandyam, N. Ahmed, and N. Magotra. A dct-based scheme for lossless image compression. *Proceedings of SPIE - The International Society for Optical Engineering*, 02 1970. 19

[32] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 12

[33] A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*, 2017. xii, 13, 14, 15, 25

[34] M. Oquab, P. Stock, O. Gafni, D. Haziza, T. Xu, P. Zhang, O. Çelebi, Y. Hasson, P. Labatut, B. Bose-Kolanu, T. Peyronel, and C. Couprie. Low bandwidth video-chat compression using deep generative models. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2388–2397, 2021. 19, 20

[35] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019. 11

[36] K. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 484–492, 2020. 7, 10, 21

[37] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*, 2021. 7

[38] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069. PMLR, 2016. 7

[39] Y. Ren, G. Li, Y. Chen, T. H. Li, and S. Liu. Pirenderer: Controllable portrait image generation via semantic neural rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13759–13768, 2021. 6

[40] O. Rippel, S. Nair, C. Lew, S. Branson, A. G. Anderson, and L. Bourdev. Learned video compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3454–3463, 2019. 19

[41] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1

[42] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 11

[43] A. Sagheer, A. Farhan, and L. George. Fast intra-frame compression for video conferencing using adaptive shift coding. *International Journal of Computer Applications*, 81:29–33, 11 2013. 19

[44] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe. First order motion model for image animation. In *Conference on Neural Information Processing Systems (NeurIPS)*, December 2019. x, 2, 3, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 21, 22, 23, 25, 27

[45] A. Siarohin, O. Woodford, J. Ren, M. Chai, and S. Tulyakov. Motion representations for articulated animation. In *CVPR*, 2021. x, 5, 8, 15, 17

[46] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 25

[47] Y. Song, J. Zhu, D. Li, X. Wang, and H. Qi. Talking face generation by conditional recurrent adversarial network. *arXiv preprint arXiv:1804.04786*, 2018. 2, 6, 7

[48] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017. 21

[49] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 14, 25

[50] P. Tandon, S. Chandak, P. Pataranutaporn, Y. Liu, A. M. Mapuranga, P. Maes, T. Weissman, and M. Sra. Txt2vid: Ultra-low bitrate compression of talking-head videos via text. *arXiv preprint arXiv:2106.14014*, 2021. 2, 6, 7

[51] S. Tripathy, J. Kannala, and E. Rahtu. Icface: Interpretable and controllable face reenactment using gans. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020. 8

[52] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *ArXiv*, abs/1607.08022, 2016. 12

[53] L. Wang, W. Han, F. K. Soong, and Q. Huo. Text driven 3d photo-realistic talking head. In *Twelfth Annual Conference of the International Speech Communication Association*, 2011. 6

[54] S. Wang, L. Li, Y. Ding, C. Fan, and X. Yu. Audio2head: Audio-driven one-shot talking-head generation with natural head motion. *arXiv preprint arXiv:2107.09293*, 2021. 6, 7, 21

[55] T. Wang, T. Zhang, and B. Lovell. Faces a la carte: Text-to-face generation via attribute disentanglement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3380–3388, January 2021. 7

[56] T.-C. Wang, A. Mallya, and M.-Y. Liu. One-shot free-view neural talking-head synthesis for video conferencing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10039–10049, 2021. 2, 6, 7, 11, 15, 17, 18, 20, 21, 22, 25, 26, 27

[57] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560–576, 2003. 19, 25, 27

[58] W. Wu, Y. Zhang, C. Li, C. Qian, and C. C. Loy. Reenactgan: Learning to reenact faces via boundary transfer. In *Proceedings of the European conference on computer vision (ECCV)*, pages 603–619, 2018. 21

[59] T. Yang, P. Ren, X. Xie, and L. Zhang. Gan prior embedded network for blind face restoration in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 672–681, 2021. 22, 24

[60] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9459–9468, 2019. 6

[61] 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*, pages 586–595, 2018. 12, 24

[62] Z. Zhang, L. Li, Y. Ding, and C. Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3661–3670, 2021. 8, 21

[63] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang. Talking face generation by adversarially disentangled audio-visual representation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 9299–9306, 2019. 2, 6, 7

[64] H. Zhou, Y. Sun, W. Wu, C. C. Loy, X. Wang, and Z. Liu. Pose-controllable talking face generation by implicitly modularized audio-visual representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4176–4186, 2021. 2, 6, 8, 15, 17, 21

[65] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis, and D. Li. Makelttalk: speaker-aware talking-head animation. *ACM Transactions on Graphics (TOG)*, 39(6):1–15, 2020. 2, 6, 7, 10, 21