# Audio-driven Talking Video Frame Restoration

Harry Cheng, *Student Member, IEEE*, Yangyang Guo, Jianhua Yin, *Member, IEEE*, Haonan Chen, *Member, IEEE*, Jiafang Wang, *Member, IEEE*, Liqiang Nie, *Senior Member, IEEE*

***Abstract*—Talking video frames occasionally drop while streaming for reasons like network errors, which greatly hurts the online team collaboration and user experiences. Directly generating the dropped frames from the remaining ones is unfavorable since a person's lip motion is usually non-linear and thus hard to be restored when consecutive frames are missing. Nevertheless, the audio content provides strong signals for lip motion and is less likely to drop during transmitting. Inspired by this, as an initial attempt, we present the task of audio-driven talking video frame restoration in this paper, i.e., restoring dropped video frames by jointly leveraging the audio and remaining video frames. Towards the high-quality frame generation, we devise a cross-modal frame restoration network. This network aligns the complete audio content with video frames, precisely identifies and sequentially generates the dropped frames. To justify our model, we construct a new dataset, Talking Video Frames Drop, TVFD for short, consisting of 2.5K video and 144K frames in total. We conduct extensive experiments over TVFD and another publicly accessible dataset - Voxceleb2. Our model obtains significantly improved performance as compared to other state-of-the-art competitors.**

***Index Terms*—Frame Restoration, Frame-Dropped Video, Cross-Modal Learning, Dynamic Programming, Generative Adversial Network.**

## I. Introduction

ONLINE videotelephony services (e.g., video conference, live streaming, and online education) thrive with the catastrophic burst of COVID-19 to meet the communication needs under social distancing. For instance, Zoom has reached more than 300 million daily meeting participants[1], and more than 2 million viewers watch live videos simultaneously in Twitch[2]. Despite its significance in teamwork and entertainment, one pivotal problem of the online talking video services is that some video frames often unexpectedly drop when people are talking (an example can be found in Figure 1). As a result, the user experience is largely affected, leading to losing active users (AU). Several factors contribute to frame dropping, including but not limited to network error, camera problem, and software malfunction.

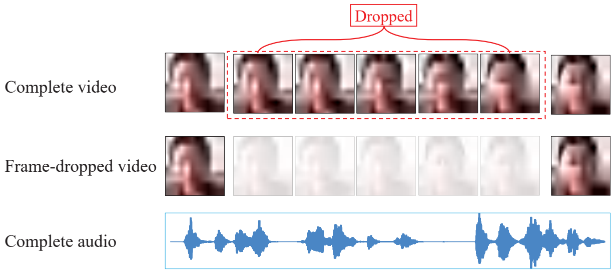


Fig. 1: Examples of frame dropping in video conferences. Under normal conditions, seven consecutive frames with the complete audio should be transmitted to other conference participants. However, five of them are dropped due to unknown reasons. Consequently, other participants can only receive the woman's voice while her face abnormally remains static.

As compared to video frames [1], the audio signal is less likely to drop during transmitting because of its unsophisticated coding format and easy-to-compression [2]. Service providers are thus motivated to ensure the audio transmitting primarily. Moreover, Nagrani *et al.* [3] revealed the high correlation between audio voices and human faces in visual frames. In light of this, we present the task of Audio-driven Talking Video Frame Restoration, dubbed as ATVFR, which aims to restore dropped frames with the complete audio track and remaining video frames. This task is distinguished from other related ones like talking face generation [4], audio-visual lip synchronization [5], and video-based frame interpolation [6], because of its practicality in the online talking video applications. In particular, the first two tasks mainly target at generating a precise lip motion, while the last one inevitably requires consecutive video frames and neglects the audio signals despite their importance.

ATVFR is challenging due to the following facts: 1) Identifying the dropped video frames is arduous, as frame dropping is quite random during the video service. 2) It is hard to align lip motions with the audio. Because the continuous deformation of the participant's lips relates to the audio content and the personal characteristic at the same time [4]. And 3) in addition to preserving the facial details, the background around the face should be unblurred. Nevertheless, it is rather complicated to implement, as the motions and illumination are non-linear and intricate in the real world.

In this paper, to address the aforementioned challenges, we present a Cross-modal fRame rEstorATion nEtwork, CREATE for short. An overview of our model is shown in Figure 2. It mainly involves two key networks, Ali-Net and FE-Net, short for alignment network and flow estimation network, respectively. In particular, Ali-Net employs two VGG-M-based [7] models to process video and audio inputs separately and identify which frames have been dropped. The FE-Net

Manuscript received June 20, 2021; revised August 15, 2021; accepted October 2, 2021. This work was supported in part by the Shandong Provincial Natural Science Foundation under Grants ZR2019JQ23 and ZR2019QF001, in part by the National Natural Science Foundation of China under Grants U1936203 and 61802231, and in part by new AI project towards the integration of education and industry in QLUT. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaochun Cao. *(Corresponding author: Liqiang Nie.)*

Harry Cheng, Yangyang Guo, Jianhua Yin, and Liqiang Nie are with School of Computer Science and Technology, Shandong University, Qingdao, 266000, China (e-mail: xaCheng1996@gmail.com, guoyang.eric@gmail.com, jhyin@sdu.edu.cn, nieliqiang@gmail.com).

Haonan Chen and Jiafang Wang are with Tmall Genie-AI, Alibaba Cloud Intelligence Business Group, Alibaba Group, Hangzhou, 310000, China (e-mail: haolan.chn@alibaba-inc.com, jiafang.wjf@alibaba-inc.com).

[1] https://blog.zoom.us/90-day-security-plan-progress-report-april-22/.

[2] https://www.twitch.tv/directory.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMM.2021.3118287, IEEE Transactions on Multimedia

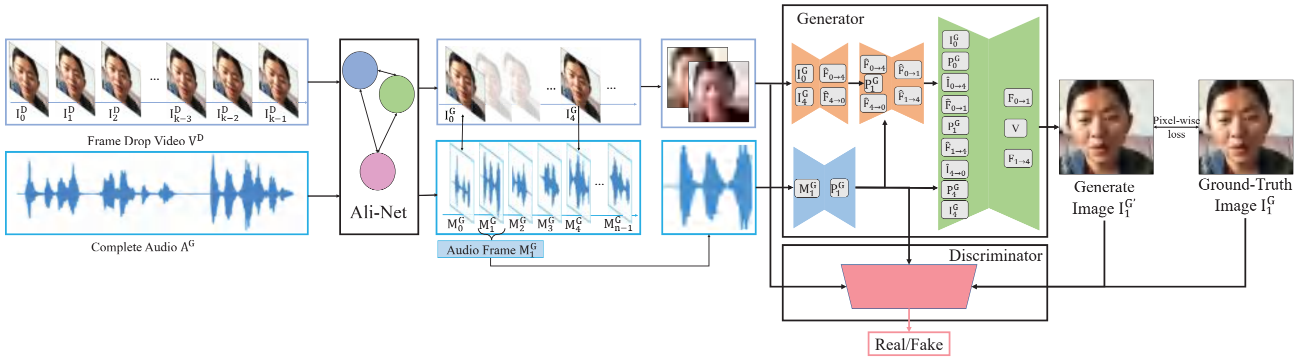

$I_0^D$ $I_1^D$ $I_2^D$ ... $I_{k-3}^D$ $I_{k-2}^D$ $I_{k-1}^D$
Frame Drop Video $V^D$
Complete Audio $A^G$
Ali-Net
$I_0^G$ ... $I_4^G$ ...
$M_0^G$ $M_1^G$ $M_2^G$ $M_3^G$ $M_4^G$ ... $M_{n-1}^G$
Audio Frame $M_1^G$
Generator
$I_0^G$ $\hat{F}_{0\to4}$ $I_4^G$ $\hat{F}_{4\to0}$
$\hat{F}_{0\to4}$ $P_1^G$ $\hat{F}_{4\to0}$ $\hat{F}_{0\to1}$ $\hat{F}_{1\to4}$
$M_1^G$ $P_1^G$
$I_0^G$ $P_0^G$ $\hat{I}_{0\to4}$ $\hat{F}_{0\to1}$ $P_1^G$ $\hat{F}_{1\to4}$ $\hat{I}_{4\to0}$ $P_4^G$ $I_4^G$
$F_{0\to1}$ V $F_{1\to4}$
Generate Image $I_1^{G'}$
Pixel-wise loss
Ground-Truth Image $I_1^G$
Discriminator
Real/Fake

Fig. 2: Schematic illustration of our proposed CREATE model. Ali-Net on the left takes the frame-dropped video and the complete audio as inputs with separate network streams. The alignment between these two modalities is performed, and the dropped frames are consequently identified. Afterward, FE-Net is devised to generate the dropped frames based upon the complete audio and remaining frames via a GAN model.

is utilized to estimate the optical flow among input frames and generate the dropped ones based on generative adversarial networks (GANs). Thereafter, we leverage the audio content in the generator and discriminator of FE-Net to fine-tune the optical flow, making the restored video frames more consistent with the audio. To facilitate the validation of our proposed model, we further construct a new dataset, namely TVFD - Talking Video Frames Drop, which is the first of its kind and large-scale, comprising over 2.5K videos and 144K frames in total. To be more specific, we invite 25 volunteers, and each of them is required to face up to the built-in device cameras and make meaningful utterances without noise. Moreover, each video is ensured to contain one visible face and motionless background only. It is worth mentioning that the videos are split according to the person IDs, which implies that the training, validation, and testing sets contain separate persons.

We conduct extensive experiments over the newly constructed TVFD dataset and a widely used dataset Voxceleb2 [8]. The experimental results demonstrate that our CREATE model yields superior performance than baselines pertaining to frame drop identification, with an average deviation of 9.1% performance improvement. Moreover, regarding the dropped frame generation, CREATE delivers higher-quality frames in terms of lip movements and background sharpening.

The main contributions of this work are threefold:

- To the best of our knowledge, we are the first on defining the task of audio-driven talking video frame restoration in literature. Apart from the promising application in the online talking video services, ATVFR is also applicable to many other video-related tasks, such as video transmission.
- We devise a novel cross-modal frame restoration model to restore the dropped frames, and the experimental results demonstrate the superiority of our model over several state-of-the-art baselines.
- We construct a new large-scale, high-quality dataset - TVFD, to facilitate the development of audio-driven talking video frame restoration.

The rest of this paper is structured as follows. Section II briefly reviews the related literature. We introduce the construction of the TVFD dataset in Section III. Section IV elaborates the definition of the ATVFR task and the proposed model. The experimental settings and results are respectively introduced in Section V and Section VI, followed by the conclusion and future work of this paper in Section VII.

## II. Related Work

In this section, we study three literature streams related to ATVFR, i.e., frame interpolation [9], talking face generation [10], and audio-visual synchronisation [11].

**Frame Interpolation** generates intermediate frames between consecutive frames to form coherent videos both spatially and temporally. Related existing methods along with this line can be roughly classified into two groups: flow-based and flow-free. Approaches in the first group are introduced to obtain the intermediate frames via bi-directional optical flows [6]. For example, DAIN [12] improves the flow estimation using a depth-aware flow projection layer, and the frames can then be generated through an adaptive warping operation. Xu *et al.* [13] exploited curvilinear trajectory and variable velocity to estimate flows and developed a flow reversal approach for frame synthesis. As to the flow-free methods, they often utilize spatially-adaptive convolution instead of optical flow to estimate the intermediate frames [14]. For instance, Choi *et al.* [15] employed a PixelShuffle operator equipped with the channel-wise attention to replace the optical flow computation module. Compared with flow-based approaches, flow-free ones usually achieve higher efficiency. However, directly applying frame interpolation methods into ATVFR is sub-optimal. Since these models only utilize the visual information while ignoring the audio cues, making it infeasible to achieve accurate lip motion generation.

**Talking Face Generation** targets at generating a face video with precise lip motions [16]. Existing methods can be divided into single-modality and multi-modality generation models based on the modalities they exploited for generation. The former ones leverage the visual information only, which is then employed to transfer the source facial expression to the target one [17]. The GAN approaches conditioned on images [18] or landmarks [19] are frequently used in these models. As to the multi-modality one, both visual and audio information is

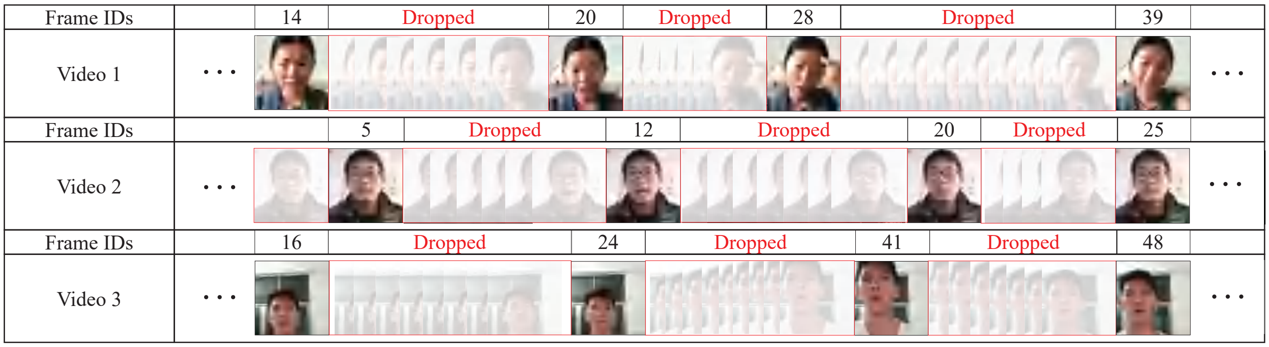


Fig. 3: Some examples from the newly collected TVFD dataset. The dropped frames are blurred and marked with red boxes. It is worth noting that our videos may contain sufficient head movements. Note that there exist complete and complex head movements in videos like video 1 and 3.

considered. For example, DAVS [4] presents to disentangle the audio-visual representation with an adversarial training approach. External information is often introduced to this task. ATVGnet [20] converts the audio content to landmarks [21] for better face video generation, and Wen *et al.* [22] extended it to 3D face model [23]. LipGAN [5] and Wav2Lip [24] focus only on the lip synchronization, leaving the upper face masked, to generate precise lip motions. In addition, some approaches such as PC-AVS [25] are also presented to generate head motions from a source pose video. To sum up, talking face generation shows a great advantage in tuning the lip movements, while generating the motions beyond the lip area is often arduous [26].

**Audio-Visual Synchronisation** is the task of retrieving a matched audio segment from a set of audio candidates based on a video clip. Among the initial efforts, SyncNet [11] employs convolutional neural networks (CNNs) to learn a joint embedding space of videos and audios, wherein the audio segment can be effectively retrieved. Inspired by SyncNet, metric learning [27] and multi-way matching strategy [28] are adopted to determine the relevance level between audios and videos. Nagrani *et al.* [29] proposed to separately learn the linguistic content and speaker identity with a content loss. The natural cross-modal synchronization between face and audio is then exploited. Afouras *et al.* [30] introduced a self-supervised learning method, where the voice sources are firstly localized and grouped, followed by the optical flow modeling to aggregate the audio-visual information over time. Although these methods have achieved assured success, they are all applied to retrieve a video or an audio segment. This is different from ATVFR, where each video frame should be taken into consideration.

## III. Preliminaries

In this section, we firstly elaborate the newly collected Talking Video Frames Drop dataset. Thereafter, a general data processing technique is given to extract both the audio and video frame features.

### A. Dataset Collection

To facilitate the study of this task, we collect a new large-scale dataset - TVFD, composed of 2,532 videos annotated by 25 volunteers with average frames of 57.1 per video. In the process of collecting data, annotators are required to record videos without any noise. Moreover, each video contains one visible face and motionless background only. Note that the videos are split into training, validation, and testing sets according to the person IDs, which is helpful to evaluate the generalization capability of models. The statistics of this dataset are summarized in Table I.

TABLE I: Statistics of TVFD.

| | |
|---|---|
| #videos | 2,532 |
| #video frames | 144,511 |
| #average frames per video | 57.1 |
| #train/val/test videos | 2,045/253/234 |
| average duration (s) per video | 1.9 |
| frame rate | 30 |
| frame drop rate | 0.87 |
| resolution | 480p |
| audio sample rate | 16kHz |
| MFCC dims | 13 |
| Spectrogram dims | 128 |

The privacy agreement limits the access to the real-world user talking videos, which drives us to collect a helpful dataset with crowdsourcing approaches. To simulate the video frame drop information in the real-world situation, during the video recording, we employed specially-tuned cameras and drivers provided by the Alibaba Group. These equipments enable us to comprehensively analyze the packet loss and the camera occupancy of online terminals, which are useful to reproduce the frame dropping process. Subsequently, we acquire the original complete videos and corresponding frame-dropped videos at the same time. We then deeply analyze the frame-dropped videos and find that the frame drop rate is about $0.87$, which means that only about $1$ frame can be successfully transmitted per $10$ frames. Some sampled videos with the dropped counterparts are illustrated in Figure 3.

### B. Data Preprocessing

Inspired by previous studies [28], [31], the audio contents are preprocessed in three steps. Firstly, each second of audio is sequentially divided into 100 segments with a 10ms sliding window (each segment is a 25ms audio). Secondly, $p$ segments are sampled evenly, where the value of $p$ is based on the video frame rate in the dataset, namely 30 for TVFD.

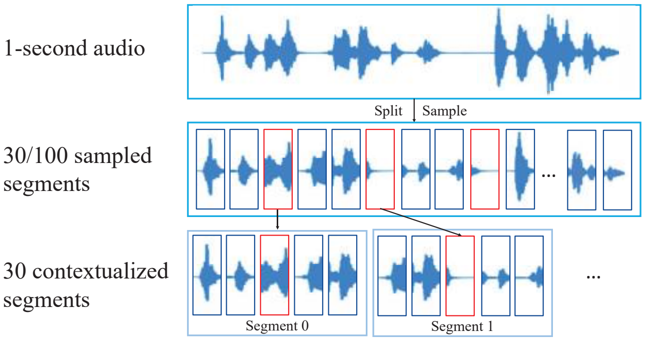


Fig. 4: Process of the audio segment generation.

Lastly, we extract two adjacent segments before and after the current sampled segment and compose a more contextualized audio segment (five segments included). Figure 4 illustrates the process of audio segmentation. For the video frame, we identify the face through the dlib tool[3] and resize each frame to 256 × 256.

In this way, the audio, the frame-dropped video, the complete video, and the generated video can be respectively represented as:

$$\begin{cases} \mathcal{A} = \{\mathbf{M}_0, \mathbf{M}_1, \dots, \mathbf{M}_{n-2}, \mathbf{M}_{n-1}\}, \\ \mathcal{V}^D = \{\mathbf{I}_0^D, \mathbf{I}_1^D, \dots, \mathbf{I}_{k-2}^D, \mathbf{I}_{k-1}^D\}, \\ \mathcal{V} = \{\mathbf{I}_0, \mathbf{I}_1, \dots, \mathbf{I}_{n-2}, \mathbf{I}_{n-1}\}, \\ \hat{\mathcal{V}} = \{\hat{\mathbf{I}}_0, \hat{\mathbf{I}}_1, \dots, \hat{\mathbf{I}}_{n-2}, \hat{\mathbf{I}}_{n-1}\}, \end{cases} \tag{1}$$

where $k \leq n$ is the number of remaining frames in $\mathcal{V}^D$. The symbols $\mathbf{I}_i^D$, and $\mathbf{I}_i$ respectively denote three-channel RGB images with a resolution of 256 × 256, and $\mathbf{M}_i$ represents a segment from audio $\mathcal{A}$.

## IV. Proposed Method

In this paper, we present a cross-modal frame restoration network, dubbed as CREATE, to tackle the task of audio-driven talking video frame restoration. In what follows, we first present the method formulation, and then the two critical networks of CREATE: alignment network (Ali-Net) and flow estimation network (FE-Net), are sequentially introduced.

### A. Method Formulation

We firstly provide the formal definition of ATVFR in this subsection. Specifically, given a frame-dropped video $\mathcal{V}^D$ and a complete audio $\mathcal{A}$, ATVFR targets at generating a video $\hat{\mathcal{V}}$ based on $\mathcal{V}^D$ and $\mathcal{A}$:

$$\hat{\mathcal{V}} = \phi(\mathcal{V}^D, \mathcal{A}), \tag{2}$$

where $\phi$ constitutes a function with a parameter set $\theta$, and $\hat{\mathcal{V}}$ should be sufficiently similar to the ground-truth video $\mathcal{V}$:

$$\theta_\phi^* = \arg\min_{\theta_\phi}(L(\hat{\mathcal{V}}, \mathcal{V})), \tag{3}$$

where $\theta^*$ denotes the optimal parameters; $L$ is an empirical loss function.

To implement this, we devise CREATE method, which could be stated as:

$$\hat{\mathcal{V}} = f_{fe}(f_{ali}(\mathcal{V}^D, \mathcal{A}), \mathcal{A}), \tag{4}$$

where $f_{ali}(\cdot,\cdot)$ and $f_{fe}(\cdot,\cdot)$ denote the Ali-Net and FE-Net, respectively. Based on $\mathcal{A}$, $\mathcal{V}$, and $\hat{\mathcal{V}}$, we can calculate the loss according to the content of Section IV and obtain the following goals as:

$$\theta^* = \arg\min_{\theta}(L_T(\mathcal{V}, \mathcal{A}) + L_G(\hat{\mathcal{V}}, \mathcal{V})), \tag{5}$$

where $L_T$ and $L_G$ are the loss functions utilized in the Ali-Net and FE-Net, respectively; $L_T$ constraints Ali-Net to identify which frames in $\mathcal{V}$ are dropped, and $L_G$ measures the quality of $\hat{\mathcal{V}}$; $\theta$ is the set of parameters. We expect to obtain a set of parameter combinations $\theta^*$ that minimizes $L_T + L_G$.

### B. Alignment Network

With the preprocessing of video frames and audios, the Ali-Net demonstrated in Figure 5 is utilized to identify which frames in the complete video $\mathcal{V}$ are dropped. Specifically, given the frame-dropped video $\mathcal{V}^D$ and audio $\mathcal{A}$, Ali-Net obtains a position sequence $\mathbf{s} = \{s_0, s_1, \dots s_{k-1}\}$, where $s_i$ represents the position of the frame $\mathbf{I}_i^D$ in $\mathcal{V}$. Take Figure 2 as an example, Ali-Net calculates $s_0 = 0$ and $s_1 = 4$ based on $\mathcal{V}^D$ and $\mathcal{A}$, denoting that $\mathbf{I}_0^D$ and $\mathbf{I}_1^D$ are actually $\mathbf{I}_0$ and $\mathbf{I}_4$ in $\mathcal{V}$, respectively. To implement this, we design two modules in Ali-Net, i.e., cross-modal feature extraction module and dynamic position retrieval module. The former is utilized to extract the features from inputs in different modalities and map these representations into a common latent space. The latter retrieves the position sequence $\mathbf{s}$ based on the extracted features at once with an effective dynamic programming strategy.

**Cross-modal feature extraction module**. The inputs to this module are composed of the complete video $\mathcal{V}$ and the audio $\mathcal{A}$. To extract the video and audio features, we design two sub-networks [32], Video-Ali and Audio-Ali, as illustrated in Figure 6.

To enhance the feature extraction, we firstly obtain an interfering audio $\mathcal{A}^I$ by randomly shuffling the segments of $\mathcal{A}$. In this way, the corresponding segments $\mathbf{M}_i$ and $\mathbf{M}_i^I$ in $\mathcal{A}$ and $\mathcal{A}^I$ would share the same personal characteristics, yet with inconsistent audio contents. In light of this, we could minimize the influence of irrelevant factors, such as timbre and tone.

We then employ the triplet loss [33] function as follows:

$$L_T = \max\{d(h(\mathbf{I}_i), h(\mathbf{M}_i)) - d(h(\mathbf{I}_i), h(\mathbf{M}_i^I)) + r, 0\}, \tag{6}$$

where $\mathbf{I}_i$ is the video frame from the complete video $\mathcal{V}$, $\mathbf{M}_i$ denotes audio segments from $\mathcal{A}$ and $\mathbf{M}_i^I$ is the artificially constructed interfering audio $\mathcal{A}^I$. Meanwhile, the function $h(\cdot)$ denotes the procedure of feature extraction of Video-Ali and Audio-Ali, $r$ represets a threshold which is a hyperparameter and set to 1.0 in our experiments, and $d(\cdot,\cdot)$ is defined as:

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2, \tag{7}$$

[3] http://dlib.net/.

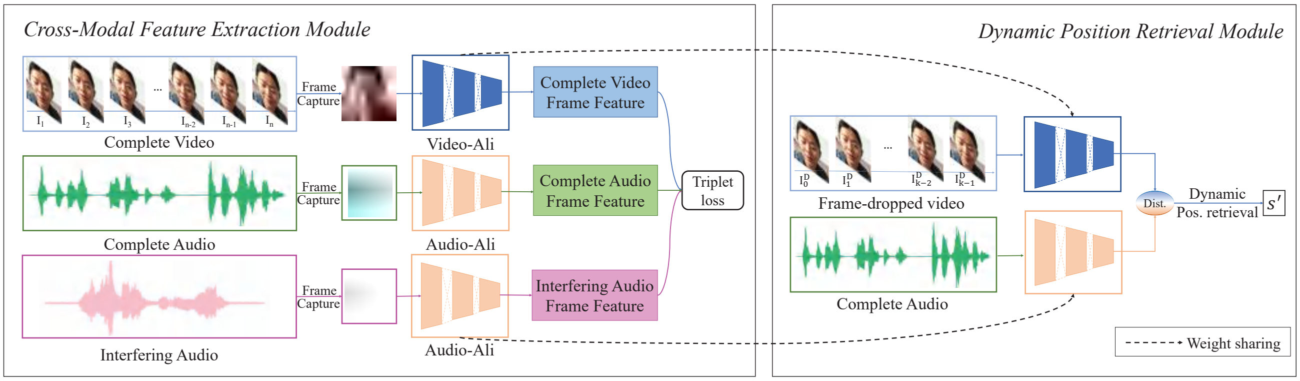
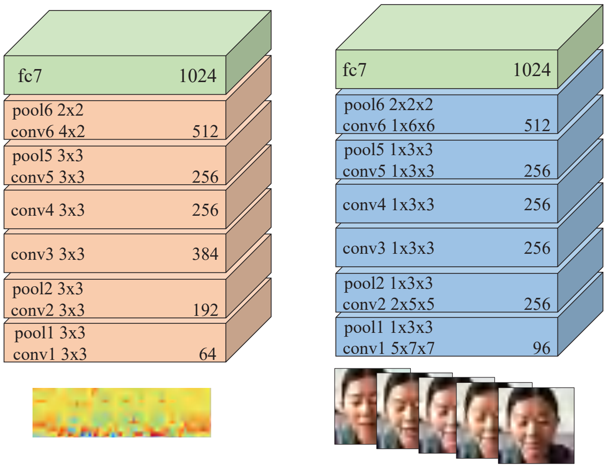


Fig. 5: Network architecture of Ali-Net.



(a) Audio (b) Video

Fig. 6: Trunk architecture of (a) Audio-Ali and (b) Video-Ali.

which indicates the distance in the common space between input $\mathbf{M}_i$ and $\mathbf{I}_i$.

The triplet loss function in Eqn.(6) refers the video frame $\mathbf{I}_i$ as an anchor, and the corresponding audio segments $\mathbf{M}_i$ and $\mathbf{M}_i^I$ as positive and negative samples, respectively. By means of calculating the difference between $d(h(\mathbf{I}_i), h(\mathbf{M}_i))$ and $d(h(\mathbf{I}_i), h(\mathbf{M}_i^I))$, Ali-Net pulls the matched video frame $\mathbf{I}_i$ and the positive audio segment $\mathbf{M}_i$ closer while pushes the unmatched negative segment $\mathbf{M}_i^I$ further away [34]. The hyperparameter $r$ works as a threshold to make the positive and negative samples more separable. The main objective of this module is to learn two skilled feature extraction sub-networks, i.e., Video-Ali and Audio-Ali, which are favorable in feature extraction and feature space mapping.

**Dynamic position retrieval module**. In this module, we apply an effective dynamic programming strategy to predict the position sequence. We first employ the well-trained Video-Ali and Audio-Ali to consume the frame-dropped video $\mathcal{V}^D$ and the complete audio $\mathcal{A}$, respectively. We then calculate the distance between any pair of the visual-audio features $\mathbf{I}_i^D$ and $\mathbf{M}_j$ with the distance function in Eqn.(7). Consequently, a distance matrix $\mathbf{D}$ can be obtained, where each element $\mathbf{D}_{ij}$ represents the distance between $\mathbf{I}_i^D$ and $\mathbf{M}_j$:

$$\mathbf{D}_{ij} = \left\| h(\mathbf{I}_i^D) - h(\mathbf{M}_j) \right\|_2 . \tag{8}$$

Subsequently, we need to determine which video frames are matched with the audio segments based on the distance matrix. A popular approach is to introduce a threshold $\beta$. For instance, $\mathbf{D}_{ij} < \beta$ implies that $\mathbf{I}_i^D$ and $\mathbf{M}_j$ are matched. However, this approach has some flaws which are difficult to resolve. One example is shown in Figure 7(a), with the ground truth position sequence $\mathbf{s} = \{0, 2, 3, 4\}$. If we use a fixed threshold $\beta = 5$ in this case, the predicted sequence will be $\mathbf{s}' = \{0, None, 5, 2\}$, which is not satisfied. Beyond the fixed threshold, we can also employ dynamic $\beta$ to make $\mathbf{s}'$ closer to the ground truth. However, complex tuning and increased computational cost come along, leading to sub-optimal performance.

The key issue of the threshold approach is that it ignores the order of the video frames in $\mathcal{V}^D$. Considering the situation that $\mathbf{I}_0^D$ matches $\mathbf{M}_2$, $\mathbf{I}_1^D$ will not match $\mathbf{M}_1$, no matter whether $\mathbf{D}_{11}$ is smaller than $\beta$. The possible reason is that the frame $\mathbf{I}_0^D$ is prior to $\mathbf{I}_1^D$, and the matched audio segment of $\mathbf{I}_1^D$ cannot appear in front of $\mathbf{M}_2$ (i.e., the index of $\mathbf{M}$ should be larger than 2). Therefore, for all video frames $\{\mathbf{I}_0^D, \mathbf{I}_1^D, \mathbf{I}_2^D...\mathbf{I}_{k-1}^D\}$, it is expected to explore an ordered sequence of audio segments $\{\mathbf{M}_0^{'}, \mathbf{M}_1^{'}, \mathbf{M}_2^{'}...\mathbf{M}_{k-1}^{'}\}$ that minimizes the sum of their corresponding distances:

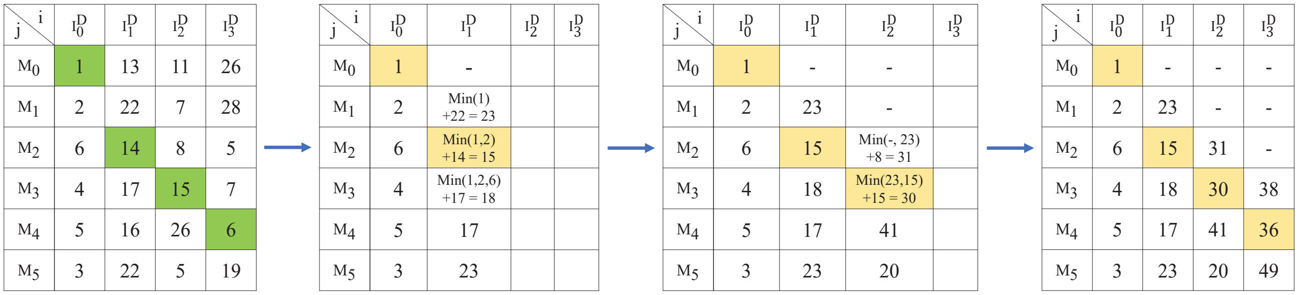
$$d_{sum} = \sum_{i=0}^{k-1} d(\mathbf{I}_i^D, \mathbf{M}_i^{'}). \tag{9}$$

To this end, we propose a dynamic programming method, which can be formulated as:

$$\mathbf{DP}(i,j) = \begin{cases} \mathbf{D}_{ij}, & i = 0, \\ \mathbf{D}_{ij} + \min\{\mathbf{DP}(i-1, 0\colon j)\}, & i > 0, \end{cases} \tag{10}$$

where $\mathbf{DP}$ is a new matrix constructed via the distance matrix. Each value $\mathbf{DP}(i,j)$ denotes the smallest $d_{sum}$ between video frames $\{\mathbf{I}_0^D, \mathbf{I}_1^D, ...\mathbf{I}_i^D\}$ and audio segments $\{\mathbf{M}_0, \mathbf{M}_1...\mathbf{M}_j\}$ if $\mathbf{I}_i^D$ matches $\mathbf{M}_j$.

Figure 7 shows an example of calculating the $\mathbf{DP}$ matrix. Specifically, Figure 7(a) is the distance matrix, with the ground truth position sequence $\mathbf{s} = \{0, 2, 3, 4\}$. Figure 7(b) - (d) illustrate the computational process, where each $\mathbf{DP}(i,j)$ is derived by the previous state. Taking $\mathbf{I}_2^D$, whose calculation procedure is shown in Figure 7(c), as an example, $\mathbf{DP}(2,0)$ and $\mathbf{DP}(2,1)$ are null since two frames, i.e., $\mathbf{I}_0^D$ and $\mathbf{I}_1^D$ have



| i<br>j | $I_0^D$ | $I_1^D$ | $I_2^D$ | $I_3^D$ |
|---|---|---|---|---|
| $M_0$ | 1 | 13 | 11 | 26 |
| $M_1$ | 2 | 22 | 7 | 28 |
| $M_2$ | 6 | 14 | 8 | 5 |
| $M_3$ | 4 | 17 | 15 | 7 |
| $M_4$ | 5 | 16 | 26 | 6 |
| $M_5$ | 3 | 22 | 5 | 19 |

(a) Distance Matrix.

| i<br>j | $I_0^D$ | $I_1^D$ | $I_2^D$ | $I_3^D$ |
|---|---|---|---|---|
| $M_0$ | 1 | - | | |
| $M_1$ | 2 | Min(1)<br>+22 = 23 | | |
| $M_2$ | 6 | Min(1,2)<br>+14 = 15 | | |
| $M_3$ | 4 | Min(1,2,6)<br>+17 = 18 | | |
| $M_4$ | 5 | 17 | | |
| $M_5$ | 3 | 23 | | |

(b) DP Process($I_1^D$).

| i<br>j | $I_0^D$ | $I_1^D$ | $I_2^D$ | $I_3^D$ |
|---|---|---|---|---|
| $M_0$ | 1 | - | - | |
| $M_1$ | 2 | 23 | - | |
| $M_2$ | 6 | 15 | Min(-, 23)<br>+8 = 31 | |
| $M_3$ | 4 | 18 | Min(23,15)<br>+15 = 30 | |
| $M_4$ | 5 | 17 | 41 | |
| $M_5$ | 3 | 23 | 20 | |

(c) DP Process($I_2^D$).

| i<br>j | $I_0^D$ | $I_1^D$ | $I_2^D$ | $I_3^D$ |
|---|---|---|---|---|
| $M_0$ | 1 | - | - | - |
| $M_1$ | 2 | 23 | - | - |
| $M_2$ | 6 | 15 | 31 | - |
| $M_3$ | 4 | 18 | 30 | 38 |
| $M_4$ | 5 | 17 | 41 | 36 |
| $M_5$ | 3 | 23 | 20 | 49 |

(d) DP Matrix.

Fig. 7: One running example based upon our dynamic strategy. From left to right, (a) is the distance matrix. (b) and (c) represent the computational process for each column illustrated in Eqn.(10), respectively, and (d) is the **DP** matrix.

already been computed. For $\mathbf{DP}(2,2)$, it can be determined by:

$$\mathbf{DP}(2,2) = \mathbf{D}_{22} + \min\{\mathbf{DP}(1,0\colon 2)\}, \tag{11}$$

where $\mathbf{D}_{22}$=8, and $\min\{\mathbf{DP}(1,0\colon 2)\} = \min\{-,23\}$. The calculation of $\mathbf{DP}(2,3)$, $\mathbf{DP}(2,4)$, and $\mathbf{DP}(2,5)$ follows analogous dictates.

Consequently, $\mathbf{DP}$ matrix, as shown in Figure 7(d), is obtained. The rightmost column denotes the optimal $d_{sum}$ between $\{\mathbf{I}_0^D, \mathbf{I}_1^D, \mathbf{I}_2^D ... \mathbf{I}_{k-1}^D\}$ and different audio segment sets. We seek the minimum value $\mathbf{DP}(3,4)$, search the path from the $\mathbf{DP}(3,4)$ to $\mathbf{DP}(0,0\colon k)$ backward, and finally acquire a sequence $\mathbf{s}^t = \{\mathbf{DP}(3,4), \mathbf{DP}(2,3), \mathbf{DP}(1,2), \mathbf{DP}(0,0)\}$. In light of this, we predict a position sequence $\mathbf{s}' = \{0,2,3,4\}$ from $\mathbf{s}^t$, which is consistent with the correct sequence.

### C. Flow Estimation Network

Based on the output frame sequence $\mathbf{s}$ from Ali-Net, for any two consecutive frames $\mathbf{I}_i^D$ and $\mathbf{I}_{i+1}^D$, it is expected that $x$ ($s_{i+1}$-$s_i$-1) frames are dropped in this case. We then leverage the FE-Net to restore these $x$ frames. For simplicity, we define $l = s_i$ and $r = s_{i+1}$, and then $\mathbf{I}_i^D$ and $\mathbf{I}_{i+1}^D$ could also be expressed as $\mathbf{I}_l$ and $\mathbf{I}_r$, respectively.

**Generator Pipeline.** As shown in Figure 8, the input of the generator of FE-Net is composed of two frames $\mathbf{I}_l$, $\mathbf{I}_r$, and an audio segment $\mathbf{M}_t$, where $l < t < r$. The dropped frame $\mathbf{I}_t$ therefore can be restored based on the optical flows $\mathbf{F}_{t\to l}$ and $\mathbf{F}_{t\to r}$. Towards this end, we construct four U-Net-based [35] networks, namely $audio\_net$, $flow\_net$, $refine\_net$, and $inter\_net$, to sequentially generate the dropped frames.

Firstly, $audio\_net$ is applied to process $\mathbf{M}_t$ to extract the audio feature $\mathbf{P}_t$. We then utilize $flow\_net$ to estimate the bi-directional flow $\mathbf{F}_{l\to r}$ and $\mathbf{F}_{r\to l}$ based on $\mathbf{I}_l$ and $\mathbf{I}_r$. To approximate the intermediate optical flow $\hat{\mathbf{F}}_{t\to l}$ and $\hat{\mathbf{F}}_{t\to r}$, $refine\_net$ is employed to fuse and tune the bi-directional flows. It is well accepted that the audio can strongly support the non-linear lip movements generation [5], [24], $\mathbf{P}_t$ is thus taken as an external channel to $\mathbf{F}_{l\to r}$ and $\mathbf{F}_{r\to l}$ in $refine\_net$. To this end, these features constitute the input feature map and the output features are referred to two new flows $\hat{\mathbf{F}}_{t\to l}$ and $\hat{\mathbf{F}}_{t\to r}$. This method can take advantage of the reconstruction capability of U-Net. $\hat{\mathbf{F}}_{t\to l}$ and $\hat{\mathbf{F}}_{t\to r}$ could then be employed to respectively generate the intermediate frame $\hat{\mathbf{I}}_{t\to l}$ and $\hat{\mathbf{I}}_{t\to r}$:

$$\begin{cases} \hat{\mathbf{I}}_{t\to l} = g(\mathbf{I}_l, \hat{\mathbf{F}}_{t\to l}), \\ \hat{\mathbf{I}}_{t\to r} = g(\mathbf{I}_r, \hat{\mathbf{F}}_{t\to r}), \end{cases} \tag{12}$$

where $g(\cdot,\cdot)$ is a backward warping function which can be implemented with the bilinear interpolation [36], [37]. Finally, we integrate the outputs, i.e., $\mathbf{I}_l$, $\mathbf{P}_l$, $\hat{\mathbf{F}}_{l\to r}$, $\hat{\mathbf{I}}_{t\to l}$, $\mathbf{P}_t$, $\hat{\mathbf{I}}_{t\to r}$, $\hat{\mathbf{F}}_{r\to l}$, $\mathbf{P}_r$, $\mathbf{I}_r$ together, and input them to the $inter\_net$ for learning the optical flows $\mathbf{F}_{t\to l}$ and $\mathbf{F}_{t\to r}$, and a visualization matrix $\mathbf{W}$. It is worth noting that $\mathbf{P}_l$ and $\mathbf{P}_r$ are features of $\mathbf{M}_l$ and $\mathbf{M}_r$, respectively. $\mathbf{W} \in [0,1]$ is a weight matrix with the same size as $\mathbf{I}_l$ and $\mathbf{I}_r$. Each value $\mathbf{W}(i,j)$ represents the contribution of each pixel in $\mathbf{I}_l$ and $\mathbf{I}_r$ to the restored frame $\hat{\mathbf{I}}_t$. The function of obtaining $\hat{\mathbf{I}}_t$ is defined as follows:

$$\begin{aligned} \hat{\mathbf{I}}_t = {} & (1-v)\mathbf{W} \odot g(\mathbf{I}_l, \mathbf{F}_{t\to l}) \\ & + v(1-\mathbf{W}) \odot g(\mathbf{I}_r, \mathbf{F}_{t\to r}), \end{aligned} \tag{13}$$

where $v = \frac{t-l}{x+1}$. A normalization factor is applied to normalize $\hat{\mathbf{I}}_t$,

$$\hat{\mathbf{I}}_t = \frac{1}{\mathbf{Z}} \odot \hat{\mathbf{I}}_t, \tag{14}$$

where $\mathbf{Z} = (1-v)\mathbf{W} + v(1-\mathbf{W})$.

**Loss Functions for the Generator.** The overall pixel-wise loss function $L_G$ in the generator is a linear combination of four terms:

$$L_G = \lambda_r L_r + \lambda_p L_p + \lambda_w L_w + \lambda_s L_s, \tag{15}$$

where $\lambda_r$, $\lambda_p$, $\lambda_p$, and $\lambda_s$ denote the weights of these four losses. In the next, the detailed function of these losses is sequentially provided.

*Reconstruction loss* $L_r$. Similar to [6], it models the reconstruction quality of the intermediate frames with:

$$L_r = \frac{1}{x} \sum_{k=1}^{x} \left\| \mathbf{I}_k - \hat{\mathbf{I}}_k \right\|_1 . \tag{16}$$

*Perceptual loss* $L_p$. It is employed to make frames sharper:

$$L_p = \frac{1}{x} \sum_{k=1}^{x} \left\| \phi\left(\mathbf{I}_k\right) - \phi\left(\hat{\mathbf{I}}_k\right) \right\|_2 , \tag{17}$$

where $\phi$ denotes $conv4\_3$ from a pretrained VGG16 model [38].

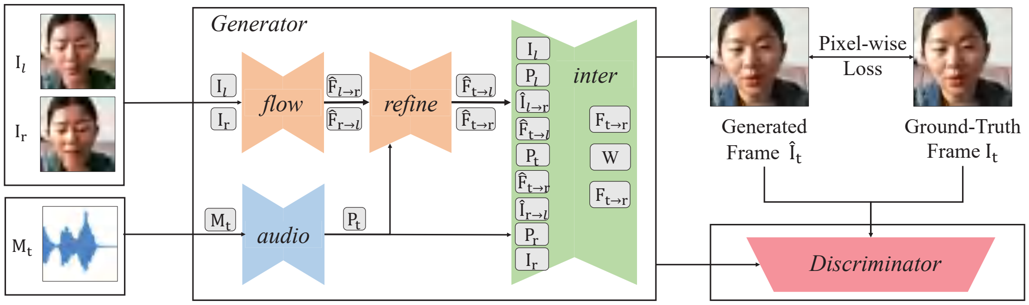


Fig. 8: Illustration of the FE Net structure.

*Warping loss* $L_w$. This loss function is used to model the the quality of the computed optical flow, which is defined as:

$$\begin{aligned} L_w = & \left\|\mathbf{I}_l - g\left(\mathbf{I}_r, \mathbf{F}_{l\to r}\right)\right\|_1 + \left\|\mathbf{I}_r - g\left(\mathbf{I}_l, \mathbf{F}_{r\to l}\right)\right\|_1 \\ & + \frac{1}{x}\sum_{k=1}^{x} \left\|\mathbf{I}_k - g(\mathbf{I}_l, \mathbf{F}_{k\to l})\right\|_1 \\ & + \frac{1}{x}\sum_{k=1}^{x} \left\|\mathbf{I}_k - g(\mathbf{I}_r, \mathbf{F}_{k\to r})\right\|_1 . \end{aligned} \quad (18)$$

*Smoothing loss* $L_s$. It is built to encourage neighboring pixels to have similar flow values:

$$L_s = \left\|\nabla \mathbf{F}_{l\to r}\right\|_1 + \left\|\nabla \mathbf{F}_{r\to l}\right\|_1 . \quad (19)$$

**Discriminator.** The discriminator we adopt predicts the probability of whether the restored frame and audio are paired or not, resulting in the following $L_D$ of FE-Net:

$$\begin{aligned} L_D(G, D) = & \mathbb{E}[\log D(\mathbf{I}, \mathbf{P})] + \\ & \mathbb{E}[\log(1 - D(G(\mathbf{I}, \mathbf{P}), \mathbf{P})]. \end{aligned} \quad (20)$$

## V. Experiment Settings

### A. Datasets

We extensively evaluated our model on two datasets - TVFD and Voxceleb2. Voxceleb2 is collected from YouTube videos, which contains over 1 million utterances of 6,112 celebrities. The audio settings of Voxceleb2 are the same as TVFD, while the video frame rate is 25, and the resolution is 224 × 224. It is worth noting that Voxceleb2 is not originally suitable for the ATVFR task. We, therefore, leveraged the same strategy to produce the frame drop information following TVFD. In addition, we split each dataset into training, validation, and testing sets with a ratio of 8:1:1. Note that the data partition criterion is based on person IDs to ensure that each person appears only in one set. This is useful to evaluate the model generalization capacity.

### B. Parameter settings

We implemented our model with the Pytorch toolkit[4]. And the Adam optimizer [39] is adopted with a fixed learning rate of $2\times10^{-4}$. All model parameters are initialized using random normal distribution with a mean of 0 and a standard deviation of 0.02. The model is trained up to 10 epochs with a mini-batch size of 1, i.e., the model takes one video as input at each time. The threshold $r$ used in Eqn.(6) is set 1.0 as default. We followed the same settings as reported in [6] regarding the different loss weights in Eqn.(15) for a fair comparison, where the weights are set as $\lambda_r = 1.0$, $\lambda_p = 0.005$, $\lambda_p = 0.8$, and $\lambda_s = 1.0$.

### C. Evaluation Metrics

We employed the commonly used reconstruction metrics Peak Signal-to-Noise Ratio (PSNR) [40], Structural Similarity Index Measure (SSIM) [41], and Multiscale Structural Similarity Index Measure(MS-SSIM) [42] to evaluate the generalization quality of the restored frames. Besides, for Ali-Net, we calculated the average and median of the difference between frame positions in the frame-dropped video and the prediction from Ali-Net to measure the ability to identify the dropped frames.

### D. Compared Baselines

We compared our model with seven state-of-the-art baselines regarding the metrics PSNR, SSIM, and MS-SSIM. Thereinto, two baselines are about talking face generation: namely, AVTGNET [20] and DAVS [4], which generate frames based mainly on the audio. LipGAN [5] and Wav2Lip [24] are proposed to synthesize video frames by jointly exploring video and audio, primarily focusing on the generation of lip motions. And the remaining three baselines, i.e., super-slomo [6], super-slomo with audio, and RIFE [43], are models from the frame interpolation.

- **AVTGNET** [20] devises a cascade GAN approach to generate the talking face video, where the audio is firstly transferred to landmarks and then converted to video frames. In addition, a dynamically adjustable pixel-wise loss with the attention mechanism and the regression-based discriminator are applied to enhance the model sharpening ability.
- **DAVS** [4] integrates the identity-related and speech-related information by learning disentangled audio-visual representation. It applies an associative-and-adversarial training process to disentangle these information and generate video frames.
- **LipGAN** [5] is also termed as "Face-to-Face Translation". This model could translate a video of a person speaking in language A into a target language B with realistic lip synchronization. It is one variant of GAN and generates face images conditioned on the audio.

[4] https://pytorch.org/.

TABLE II: Performance comparison between CREATE and baselines.

| Model | TVFD | | | Voxceleb2 | | |
|---|---|---|---|---|---|---|
| | PSNR ↑ | SSIM ↑ | MS-SSIM ↑ | PSNR ↑ | SSIM ↑ | MS-SSIM ↑ |
| AVTGNET | 14.593 | 0.444 | 0.510 | 19.951 | 0.700 | 0.737 |
| DAVS | 18.981 | 0.692 | 0.729 | 21.672 | 0.719 | 0.748 |
| LipGAN | 24.157 | 0.804 | 0.864 | 22.975 | 0.723 | 0.767 |
| Wav2Lip | 24.639 | 0.817 | 0.880 | 23.261 | 0.747 | 0.806 |
| Super-slomo | 26.379 | 0.856 | 0.881 | 24.726 | 0.785 | 0.833 |
| Super-slomo with audio | 26.625 | 0.859 | 0.901 | 25.125 | 0.792 | 0.839 |
| RIFE | 26.791 | 0.860 | 0.914 | 25.461 | 0.810 | 0.861 |
| **CREATE** | **26.935** | **0.863** | **0.924** | **25.813** | **0.823** | **0.870** |

TABLE III: Results of Ali-Net and baselines on TVFD.

| Model | Average ↓ | Median ↓ |
|---|---|---|
| AVE-Net | 14.620 | 11 |
| Multi-way matching | 12.360 | 9 |
| Ali-Net$_{threshold}$ | 12.377 | 10 |
| Ali-Net | **11.293** | **8** |

- **Wav2Lip** [24] is an upgraded version of lipGAN. This method employs a powerful lip-sync discriminator to enhance the generation process.
- **Super-slomo** [6] is a flow-based frame interpolation model. This model applies U-Net to estimate the optical flows between two frames, calculates the intermediate flows by weight, and synthesizes the frame based on the intermediate flows.
- **Super-slomo with audio** is a variant of Super-slomo implemented in this paper. Considering that Super-slomo does not support multi-modal inputs, we, therefore, augmented the input channels of U-Net with another audio input. The audio features are then cascaded with the optical flow to restore high-quality video frames.
- **RIFE** [43] leverages IFNet to directly estimate the intermediate flows from frames instead of linearly combining them. RIFE is supervised by ground truth optical flow in an end-to-end fashion based on a leakage distillation loss.

## VI. Experiment Results

In this section, we demonstrate and analyze the obtained experimental results. In particular, we attempt to answer the following research questions:

- **RQ1:** Can our model outperform several state-of-the-art baselines?
- **RQ2:** Are the two key networks of CREATE - Ali-Net and FE-Net, separately effective in this task?
- **RQ3:** Why the proposed method outperforms other baselines qualitatively?
- **RQ4:** Is our model practical in online talking video services on both robustness and real-time requirements?
- **RQ5:** How does our method perform under actual user test settings?

### *A. Performance Comparison (RQ1)*

To ensure a fair comparison, we input the position sequence $\mathbf{s}$ obtained by Ali-Net to baselines to identify which frames should be generated. The experimental results are displayed in Table II. From this table, we have the following observations:

- Traditional talking face generation models (i.e., AVTGNET and DAVS) perform inferior to other methods. The reason might be that these models consider only the lip motions while ignoring other movements and the background information. Meanwhile, the performance of these two models on Voxceleb2 is better than that of TVFD. We attributed this phenomenon to the fact that the face regions in Voxceleb2 are more strict than TVFD, namely, having less background information. Besides, the range of head motions is much smaller in Voxceleb2.
- By contrast, LipGAN and Wav2Lip, which can effectively integrate video and audio features simultaneously, outperform AVTGNET and DAVS. We speculated that these two approaches generate videos based on the remaining frames, thereby ensuring particular background.
- The frame interpolation baselines achieve promising performance due to the fact that they rely on the warped frames and hence obtain accurate background such as illumination information. However, unlike the talking face generation methods, the performance of these frame interpolation models declines on Voxceleb2 (although they are still significantly better than ATVGNET and DAVS). The reason is that the image quality of Voxceleb2 is slightly lower than TVFD, which hurts the performance of these three models and CREATE.
- Compared with all the baselines, our proposed model CREATE achieves superior performance regarding the PSNR, SSIM, and MS-SSIM metrics. It is verified that our model can learn the interactions between visual and audio features more effectively; thereby, the generated face motions become more accurate, and the sharpening of the background is barely affected.

### *B. Ablation Study (RQ2)*

CREATE mainly involves two critical networks, namely Ali-Net and FE-Net. In view of this, we presented the results from these two networks separately in this subsection.

**Ali-Net.** We employed the Average and Median metrics to evaluate Ali-Net and baselines. These two metrics are computed based on the original frame positions of the frames in $\mathcal{V}^D$ and the predictions from the models. Specifically, assuming that the predicted position sequence obtained by Ali-Net from a frame-dropped video is $\mathbf{s}' = \{1, 2, 3\}$, and the ground truth position sequence is $\mathbf{s} = \{2, 3, 4\}$, then we

TABLE IV: Performance comparison between FE-Net and baselines with the ground truth position sequence.

| Model | TVFD | | | Voxceleb2 | | |
|---|---|---|---|---|---|---|
| | PSNR ↑ | SSIM ↑ | MS-SSIM ↑ | PSNR ↑ | SSIM ↑ | MS-SSIM ↑ |
| AVTGNET | 14.593 | 0.444 | 0.510 | 19.951 | 0.700 | 0.737 |
| DAVS | 18.981 | 0.692 | 0.729 | 21.672 | 0.719 | 0.748 |
| LipGAN | 24.164 | 0.801 | 0.863 | 22.974 | 0.724 | 0.769 |
| Wav2Lip | 24.642 | 0.819 | 0.884 | 23.263 | 0.746 | 0.808 |
| Super-slowmo | 28.686 | 0.906 | 0.942 | 26.700 | 0.812 | 0.870 |
| Super-slomo with audio | 28.967 | 0.909 | 0.937 | 26.722 | 0.820 | 0.872 |
| RIFE | 30.365 | 0.916 | 0.951 | 26.961 | 0.832 | 0.880 |
| **FE-Net** | **30.791** | **0.923** | **0.962** | **27.413** | **0.853** | **0.897** |

obtained the prediction bias as:

$$Ali_b = \sum_{i=0}^{n-1} |s_i - s'_i|, \tag{21}$$

where $s_i$ and $s'_i$ are the corresponding value in $\mathbf{s}$ and $\mathbf{s}'$, respectively, and $Ali_b = 3$ in this case. Applying the above calculations to all frame-dropped videos, we could naturally calculate the average and median according to all the obtained $Ali_b$ values, and these two results are the metrics to evaluate the ability to identify the dropped frames. The smaller these two metrics are, the closer the predicted sequence $\mathbf{s}'$ to ground truth $\mathbf{s}$ is, and the better the model performs.

We used three baselines, i.e., AVE-Net [27], multi-way matching model [31], and Ali-Net$_{threshold}$ in this experiment. The former two are state-of-the-art audio-visual synchronization approaches, and the latter is constructed by introducing a threshold strategy to Ali-Net to demonstrate the necessity of dynamic position retrieval module. We conducted this experiment on the TVFD dataset. Table III displays that Ali-Net can significantly surpass the baselines over the two metrics. In particular, Ali-Net outperforms the multi-way matching model with 9.1% on Average and 11.1% on Median. In addition, it can also be observed that the Ali-Net with threshold strategy does not perform well, which further highlights the superiority of the dynamic programming strategy.

**FE-Net.** To evaluate the restoration capability, we employed the ground truth position sequence as input to FE-Net and baselines rather than the outputs from Ali-Net. We leveraged DAVS, AVTGNET, LipGAN, Wav2Lip, super-slomo, super-slomo with audio, and RIFE as baselines. All settings are the same as Section VI-A.

As can be seen from Table IV, our method can significantly outperform all the baselines. Note that the performance of FE-Net and the three frame interpolation models (i.e., super-slomo, super-slomo with audio, and RIFE) is greatly boosted compared to the full model in Table II. The reason is that we utilized the ground truth position sequence in this case. In contrast, the other two talking face generation baselines, i.e., AVTGNET and DAVS, are the opposite, since their video input has only one frame and does not actually need position sequences. The performance of LipGAN and Wav2Lip fluctuates very slightly since the lack of video frame problem is still not addressed.

### C. Qualitative Results (RQ3)

In this subsection, firstly, we evaluated our method CREATE and two baselines, i.e., DAVS and RIFE, on the TVFD dataset, and demonstrated some generated samples in Figure 9. As can be observed in Figure 9(a), DAVS generates favorable lip motions while other facial expressions such as closed eyes are ignored. Moreover, we zoomed in the background of the samples in Figure 9(b) and found that the background generated by DAVS is relatively blurred. In Figure 9(c), it can be seen that the lip motions of RIFE rarely change, which can be attributed to the lack of audio content. Compared with RIFE, the lip motions are more acceptable in DAVS. However, the generated frame colors are very different from the ground truth. One possible reason is that DAVS does not utilize illumination information when generating frames. In general, CREATE leverages all the lip motions, illumination, and other facial expressions, enabling itself to deliver better results.

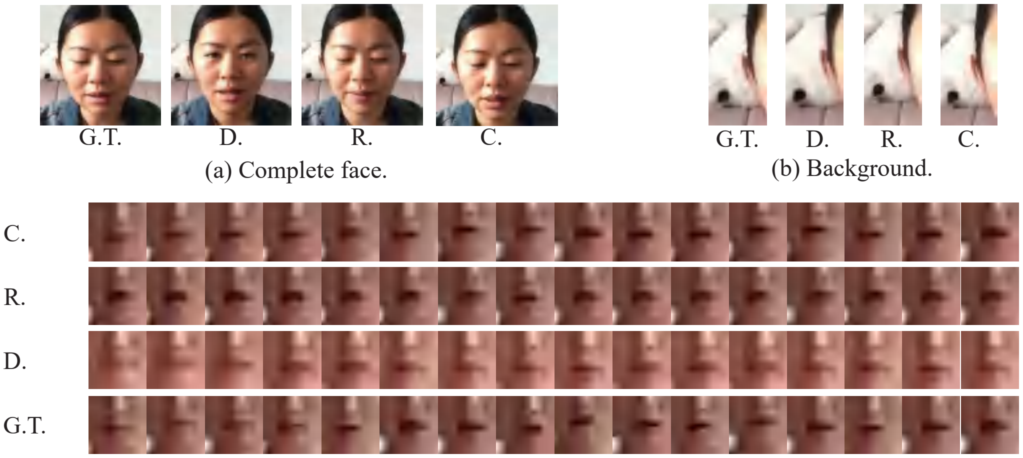
In addition, we would like to explore why LipGAN and Wav2Lip perform relatively less favorably on this task. To this end, we randomly selected 32 consecutive frames generated by Wav2Lip and compared them with CREATE and ground truth on the Voxceleb2 dataset. As shown in Figure 10, we can observe that:

- In Figure 10(a), the illumination of the Wav2Lip frames is satisfactory, and the lip motions generated by Wav2Lip are moderately consistent with ground truth. This is because Wav2Lip primarily focuses on editing a small region around the lips, and the other regions are directly copied from the original image.
- Except for the shape of lips, Wav2Lip performs worse on the upper face, especially in the region of head motions and blinking. As can be observed from Figure 10(b), Wav2Lip generates frames in a loop. It can be seen more clearly in the enlarged images of Figure 10(c) that when the lady in the ground truth frames opens her eyes, the eyes generated by Wav2Lip are closed, which seriously affects the performance of Wav2Lip. By contrast, our method, CREATE, can circumvent this problem.

### D. Robustness Analysis (RQ4)

To further verify the robustness of CREATE, we employed different frame drop rates in the TVFD dataset. Specifically, the frame drop rate utilized in Section VI-A, VI-B, and VI-C is 0.87, i.e., about seven frames dropped of every eight ones. And we explored the capability of CREATE at more diverse frame drop rates.

To roughly delineate the frame drop rate boundaries, we conducted a user study, recorded the comfort score, and demonstrated the results in Figure 11. It can be observed

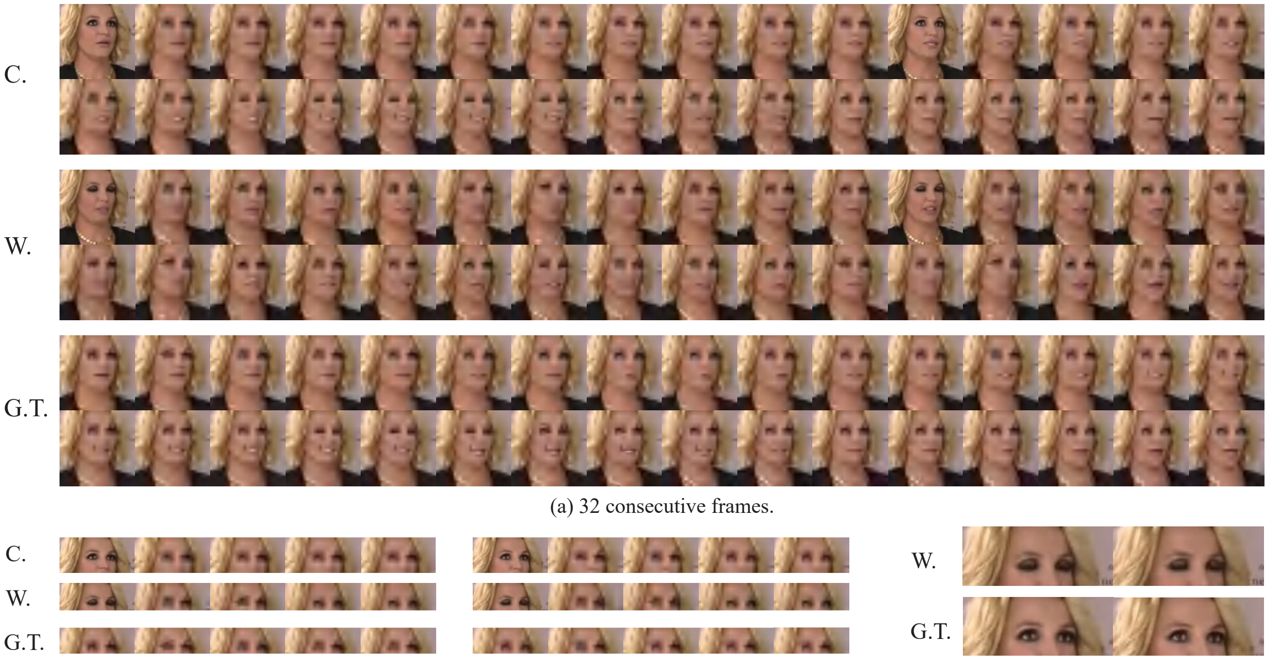


(a) Complete face.

(b) Background.

(c) Lip motions of 16 consecutive frames.

Fig. 9: Ground truth (G.T.) and generation results from baselines DAVS (D.), RIFE(R.), and our method CREATE (C.).



(a) 32 consecutive frames.

(b) The upper face in frames 1 to 5 and frames 12 to 16.

(c) Enlarge the face in frames 1 and 12.

Fig. 10: Ground truth (G.T.) and generation results from baseline Wav2Lip(W.) and our method CREATE (C.).

that the users' viewing experience starts to drop sharply when the rate reaches $\frac{1}{2}$, and the score tends to converge once we increase the drop rate to $\frac{15}{16}$.

Through controlling the cameras and applications, we adjusted the frame drop rate from $\frac{1}{2}$ to $\frac{15}{16}$. The videos were fed directly into the trained model as a new testing set, and we compared the performance of CREATE with three frame interpolation baselines in Figure 12(a) and Figure 12(b). The observations are as follows:

- The performance of all the models degrades consistently with the increase of the frame drop rate.
- When frame drop rate is $\frac{1}{2}$, the dropped frames can be well restored by CREATE and RIFE, which implies that high-quality frames could be generated without the audio information when few frames are dropped.
- It is challenging to restore high-quality frames for all the methods when the frame drop rate becomes $\frac{15}{16}$, since most visual information is dropped in this case.
- Our FE-Net performs the best over all the baselines pertaining to each frame drop rate. This proves the robustness of our method under various drop rate settings.

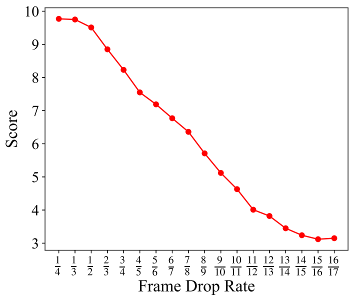
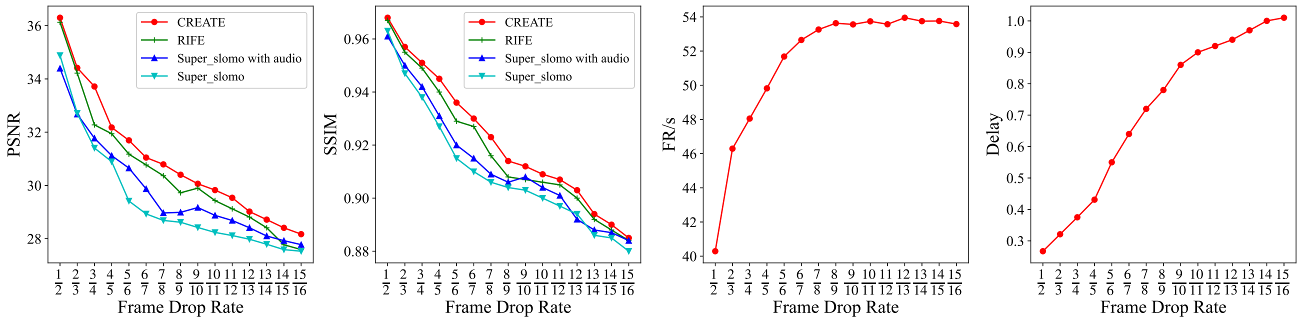


Fig. 11: The scores of viewing comfort of different frame drop rates.



(a) (b) (c) (d)

Fig. 12: Performance comparison between CREATE and three baselines with respect to the frame drop rate.

### *E. Real-time Analysis (RQ4)*

In order to verify the functionality of our approach in real-world scenarios, we applied our model to the online service provided by Alibaba Group on a small scale. In this experiment, a video and an audio stream is the input to CREATE. The audio is continuous, while the video stream may drop frames. We reported the frame restoration per second (FR/s) in Figure 12(c). The key observations are as follows:

- CREATE can restore about 53 frames per second in our experimental environment (a single RTX2080Ti), which can effortlessly meet the real-time requirements when the frame drop rate is 0.87 (the video is 30 fps).
- With the increase of frame drop rate, the speed of generating frames is getting faster. One possible reason is that the time of I/O and preprocessing (such as audio feature extraction and image reading) is saved. And when the frame drop rate reaches no less than $\frac{7}{8}$, the FR/s begins to be stable since I/O is no longer the leading cause for time-consuming.

It is worth noting that our model requires at least two frames to restore the dropped frames. In order to ensure that the model works properly, we set a buffer for the video stream to store the acquired video frames. This strategy can cause time delay to the video transmitting. Figure 12(d) shows the delay with respect to different frame drop rates. We can observe that when the frame drop rate is 0.87, the delay is 0.7 seconds, which is quite acceptable in practical applications.

In addition to restoring the dropped frames online, our model can also be applied to some offline applications. For example, users may use some personal terminals with low-quality hardware to record and upload videos. The CPU can be overloaded, and the frame drop rate is thus raised unexpectedly. In this case, our model works to restore these videos, freeing users from multiple recordings and improving user experiences.

### *F. User Study (RQ5)*

In addition to the quantitative evaluations, we also performed user studies to validate the effectiveness of the proposed method. To this end, we invited 41 volunteers of different genders, ages, and occupations to conduct the blind test. The volunteers were asked to watch 20 videos, which could be ground truth or synthesized videos via methods of DAVS, RIFE, Wav2Lip, and our CREATE. Furthermore, the method category is kept blind to volunteers. For each video, the volunteers were required to score the videos based on the following indicators:

TABLE V: Actual user test among Ground truth(G.T.), DAVS, RIFE, Wav2Lip, and CREATE.

| Model | Authenticity | Continuity | Overall perception |
|---|---|---|---|
| G.T. | 9.77 | 9.49 | 9.53 |
| DAVS | 3.91 | 8.58 | 6.72 |
| RIFE | 9.18 | 8.68 | 7.90 |
| Wav2Lip | 4.74 | 5.71 | 5.63 |
| CREATE | 9.29 | 8.83 | 8.61 |

- **Authenticity** indicates the faithfulness level of each given video, including whether the video is recorded by a natural person (not synthesized) and whether the background in the video is unblurred. A score of 10 means that this video is recorded by a natural person in a real environment.
- **Continuity** evaluates the smoothness of each video, especially the facial movements. The difference between this metric and the authenticity is that it does not consider the reality of faces. For example, the facial movements of characters in animation, such as Zootopia[5], are fluid, while the characters do not exist in the real world. A score of 10 demonstrates that the smoothness of the video is on par with ordinary videos.
- **Overall perception** is employed to measure the overall viewing experience of the video. A score of 10 indicates that the video is "comfortable" for the subject, while a lower score expresses some disappointing factors, such as lags and jitter.

The overall results are demonstrated in Table V, where the observations are three-fold:

- The optical-flow-based approaches, i.e., RIFE and CREATE, excel in terms of authenticity, which is also observed in Section VI-C. In contrast, the other two methods are less favorable. Volunteers can easily distinguish DAVS as less realistic due to its blurred background and almost non-existent head motions. Wav2lip, on the other hand, was identified as a synthetic video by most subjects due to its looping facial movements and severe jitter.

[5] https://en.wikipedia.org/wiki/Zootopia.

- In terms of continuity, Wav2lip performs worse than the other three methods due to its cyclic head motions.
- There is a particular gap between the ground truth and the synthesized videos regarding the overall perception. Volunteers' feedback demonstrates that DAVS, RIFE, and Wav2Lip affect the users' viewing experience because of the stillness of heads, unreasonable lip motions, and jitter caused by circular frames, respectively. CREATE, on the other hand, is the closest to real video in volunteers' ratings. One possible reason is that CREATE effectively leveraged the multi-modal information to generate videos.

## VII. Conclusion and Future Work

In this paper, we present the task of audio-driven talking video frame restoration, which is practical and effective in online talking video service applications. To address the challenging task, a cross-modal frame restoration model is proposed to identify and then restore the dropped frames, which jointly leverages the remaining video frames and the audio signals. Regarding the two key networks in this model, Ali-Net aligns the video and audio and then identifies the dropped frames via a dynamic position retrieval strategy; FE-Net generates the dropped frames based upon the optical flow features. To justify our proposed model, we build a new large-scale and high-quality dataset. We perform extensive experiments on this new dataset and another benchmark dataset. The results demonstrate that our model achieves state-of-the-art performance when compared to a series of baselines. Moreover, the robustness is also verified, and it thus meets the real-time requirements of online talking video service.

This work opens a new research revenue to restore the dropped frames from both videos and audios. Nevertheless, notably, this challenge is far from being settled, as the results leave a certain space for future studies. Following efforts could be devoted to two aspects: 1) building an end-to-end framework to perform the two sub-tasks, i.e., frame drop identification and generation. And 2) exploring flow-free and more efficient feature extraction approaches.

## References

[1] T. Feng, H. Sun, Q. Qi, J. Wang, and J. Liao, "Vabis: Video adaptation bitrate system for time-critical live streaming," in *IEEE Transactions on Multimedia*, 2020, pp. 2963–2976.

[2] A. Kaur, P. Sircar, and A. Banerjee, "Interpolation of lost frames of a video stream using object based motion estimation and compensation," in *Proceedings of Annual IEEE India Conference*, 2008, pp. 40–45.

[3] A. Nagrani, S. Albanie, and A. Zisserman, "Seeing voices and hearing faces: Cross-modal biometric matching," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8427–8436.

[4] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang, "Talking face generation by adversarially disentangled audio-visual representation," in *Proceedings of Conference on Artificial Intelligence*, 2019, pp. 9299–9306.

[5] P. K. R, R. Mukhopadhyay, J. Philip, A. Jha, V. Namboodiri, and C. V. Jawahar, "Towards automatic face-to-face translation," in *Proceedings of ACM International Conference on Multimedia*, 2019, pp. 1428–1436.

[6] H. Jiang, D. Sun, V. Jampani, M. Yang, E. G. Learned-Miller, and J. Kautz, "Super slomo: High quality estimation of multiple intermediate frames for video interpolation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9000–9008.

[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proceedings of British Machine Vision Conference*, 2014, pp. 1–12.

[8] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proceedings of Conference of the International Speech Communication Association*, 2018, pp. 1086–1090.

[9] A. Gupta, A. Aich, and A. K. Roy-Chowdhury, "ALANET: adaptive latent attention network for joint video deblurring and interpolation," in *ACM International Conference on Multimedia*, 2020, pp. 256–264.

[10] H. Zhu, H. Huang, Y. Li, A. Zheng, and R. He, "Arbitrary talking face generation via attentional audio-visual coherence learning," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 2020, pp. 2362–2368.

[11] J. S. Chung and A. Zisserman, "Lip reading in the wild," in *Proceedings of Asian Conference on Computer Vision*, 2016, pp. 87–103.

[12] W. Bao, W. Lai, C. Ma, X. Zhang, Z. Gao, and M. Yang, "Depth-aware video frame interpolation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3703–3712.

[13] X. Xu, L. Si-Yao, W. Sun, Q. Yin, and M. Yang, "Quadratic video interpolation," in *Proceedings of Conference on Neural Information Processing Systems*, 2019, pp. 1645–1654.

[14] S. Niklaus and F. Liu, "Context-aware synthesis for video frame interpolation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1701–1710.

[15] M. Choi, H. Kim, B. Han, N. Xu, and K. M. Lee, "Channel attention is all you need for video frame interpolation," in *Proceedings of Conference on Artificial Intelligence*, 2020, pp. 10 663–10 671.

[16] L. Xie and Z.-Q. Liu, "Realistic mouth-synching for speech-driven talking face using articulatory modelling," in *IEEE Transactions on Multimedia*, 2007, pp. 500–510.

[17] Q. Wang, H. Fan, G. Sun, W. Ren, and Y. Tang, "Recurrent generative adversarial network for face completion," in *IEEE Transactions on Multimedia*, 2021, pp. 429–442.

[18] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe, "First order motion model for image animation," in *Proceedings of Conference on Neural Information Processing Systems*, 2019, pp. 7135–7145.

[19] E. Zakharov, A. Shysheya, E. Burkov, and V. S. Lempitsky, "Few-shot adversarial learning of realistic neural talking head models," in *Proceedings of IEEE International Conference on Computer Vision*, 2019, pp. 9458–9467.

[20] L. Chen, R. K. Maddox, Z. Duan, and C. Xu, "Hierarchical cross-modal talking face generation with dynamic pixel-wise loss," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7832–7841.

[21] P. Gao, K. Lu, J. Xue, L. Shao, and J. Lyu, "A coarse-to-fine facial landmark detection method based on self-attention mechanism," in *IEEE Transactions on Multimedia*, 2021, pp. 926–938.

[22] X. Wen, M. Wang, C. Richardt, Z. Chen, and S. Hu, "Photorealistic audio-driven video portraits," in *IEEE Transactions on Visualization and Computer Graphics*, 2020, pp. 3457–3466.

[23] X. Tu, J. Zhao, M. Xie, Z. Jiang, A. Balamurugan, Y. Luo, Y. Zhao, L. He, Z. Ma, and J. Feng, "3d face reconstruction from a single image assisted by 2d face images in the wild," in *IEEE Transactions on Multimedia*, 2021, pp. 1160–1172.

[24] K. R. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. Jawahar, "A lip sync expert is all you need for speech to lip generation in the wild," in *Proceedings of ACM International Conference on Multimedia*, 2020, pp. 484–492.

[25] 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 Conference on Computer Vision and Pattern Recognition*, 2021.

[26] C. Luo, J. Zhang, J. Yu, C. W. Chen, and S. Wang, "Real-time head pose estimation and face modeling from a depth image," in *IEEE Transactions on Multimedia*, 2019, pp. 2473–2481.

[27] R. Arandjelovic and A. Zisserman, "Objects that sound," in *Proceedings of European Conference on Computer Vision*, 2018, pp. 451–466.

[28] S. Chung, J. S. Chung, and H. Kang, "Perfect match: Improved cross-modal embeddings for audio-visual synchronisation," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 3965–3969.

[29] A. Nagrani, J. S. Chung, S. Albanie, and A. Zisserman, "Disentangled speech embeddings using cross-modal self-supervision," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 6829–6833.

[30] T. Afouras, A. Owens, J. S. Chung, and A. Zisserman, "Self-supervised learning of audio-visual objects from video," in *Proceedings of European Conference on Computer Vision*, 2020, pp. 208–224.

[31] S. Chung, J. S. Chung, and H. Kang, “Perfect match: Self-supervised embeddings for cross-modal retrieval,” in *IEEE Journal of Selected Topics in Signal Processing*, 2020, pp. 568–576.

[32] Y. Guo, Z. Cheng, L. Nie, Y. Liu, Y. Wang, and M. Kankanhalli, “Quantifying and alleviating the language prior problem in visual question answering,” in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 75–84.

[33] E. Hoffer and N. Ailon, “Deep metric learning using triplet network,” in *Proceedings of Similarity-Based Pattern Recognition*, 2015, pp. 84–92.

[34] Y. Guo, L. Nie, Z. Cheng, F. Ji, J. Zhang, and A. Del Bimbo, “Adavqa: Overcoming language priors with adapted margin cosine loss,” in *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI*, 2021.

[35] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Proceedings of Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.

[36] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, “View synthesis by appearance flow,” in *Proceedings of European Conference on Computer Vision*, 2016, pp. 286–301.

[37] Z. Liu, R. A. Yeh, X. Tang, Y. Liu, and A. Agarwala, “Video frame synthesis using deep voxel flow,” in *Proceedings of IEEE International Conference on Computer Vision*, 2017, pp. 4473–4481.

[38] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proceedings of International Conference on Learning Representations*, 2015, pp. 1–14.

[39] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proceedings of International Conference on Learning Representations*, 2015, pp. 1–15.

[40] S. Niklaus and F. Liu, “Softmax splatting for video frame interpolation,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5436–5445.

[41] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” in *IEEE Transactions on Image Process*, 2004, pp. 600–612.

[42] W. Zhou, S. E. P., and B. A. C., “Multiscale structural similarity for image quality assessment,” in *IEEE Asilomar Conference on Signals, Systems Computers*, 2003, pp. 1398–1402.

[43] Z. Huang, T. Zhang, W. Heng, B. Shi, and S. Zhou, “Rife: Real-time intermediate flow estimation for video frame interpolation,” *arXiv preprint arXiv:2011.06294*, 2020.

**Harry Cheng** received the B.Eng. and master degree from Shandong University, China, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Shandong University, under the supervision of Liqiang Nie. His research interests include information retrieval and multi-modal computing.

**Yangyang Guo** is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Shandong University. He will work as a research fellow with the National University of Singapore. He has published several papers in top conferences and journals such as ACM MM, IEEE TKDE. He has served as a Regular Reviewer for journals, including IEEE TMM, IEEE TKDE, ACM ToMM.

**Jianhua Yin** received the Ph.D. degree in computer science and technology from Tsinghua University, Beijing, China, in 2017. He is currently an Associate Professor with the School of Computer Science and Technology, Shandong University, Jinan, China. He has published several papers in the top venues, such as ACM TOIS, IEEE TKDE, ACM MM, ACM SIGKDD, ACM SIGIR, and IEEE ICDE. His research interests mainly include data mining and machine learning applications.

**Haonan Chen** received the B.Sci. degree in Instrument Science and Engineering from Zhejiang University , Hangzhou, China, in 2014 and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2020. He is currently a Engineer-Algorithm-Applied Algorithm of Alibaba Group-Alibaba Cloud Intelligence Business Group-Tmall Genie-AI. His research interests include deep learning, pattern recognition, biometrics (mainly face recognition) and multimodal visual and audio algorithm.

**Jiafang Wang** received the Master degree in Institute of UAV Systems and Control from Zhejiang University, Hangzhou, China, in 2017. He was a Engineer-Algorithm-Applied Algorithm of NetEase Artificial Intelligence Division, China, from 2017 to 2018. He is currently a Engineer-Algorithm-Applied Algorithm of Alibaba Group-Alibaba Cloud Intelligence Business Group-Tmall Genie-AI. His research interests include AR/VR, Gesture Recognition, biometrics (mainly face recognition) and multimodal visual and audio algorithm.

**Liqiang Nie** received the B.Eng. degree from Xi’an Jiaotong University and the Ph.D. degree from the National University of Singapore (NUS). After Ph.D., he continued his research in NUS as a Research Fellow for more than three years. He is currently a professor with the School of Computer Science and Technology, Shandong University. Meanwhile, he is the Adjunct Dean with the Shandong AI Institute. His research interests lie primarily in multimedia computing and information retrieval. He has published around 100 papers in the top conferences or journals, with more than 10,000 Google Scholar citations. He was granted several awards like SIGIR and ACM MM best paper nomination award, ACM SIGMM Rising Star, DAMO Academy Young Fellow, AI 2000 the most influential scholar in artificial intelligence and MIT TR35 China. He is an AE of IEEE TKDE, TMM, ACM ToMM and Information Science. He serves as a PC Chair of ICIMCS 2017, PCM 2018, ChinaMM 2020 and an Area Chair of ACM MM 2018-2021.