# *Do You Really Mean That?* Content Driven Audio-Visual Deepfake Dataset and Multimodal Method for Temporal Forgery Localization

Zhixi Cai*, Kalin Stefanov*, Abhinav Dhall†* and Munawar Hayat*
{zhixi.cai,kalin.stefanov,munawar.hayat}@monash.edu, abhinav@iitrpr.ac.in
*Monash University, Australia
†Indian Institute of Technology Ropar, India

2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA) | 978-1-6654-5642-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/DICTA56598.2022.10034605

***Abstract*—Due to its high societal impact, deepfake detection is getting active attention in the computer vision community. Most deepfake detection methods rely on identity, facial attributes, and adversarial perturbation-based spatio-temporal modifications at the whole video or random locations while keeping the meaning of the content intact. However, a sophisticated deepfake may contain only a small segment of video/audio manipulation, through which the meaning of the content can be, for example, completely inverted from a sentiment perspective. We introduce a content-driven audio-visual deepfake dataset, termed Localized Audio Visual DeepFake (LAV-DF), explicitly designed for the task of learning temporal forgery localization. Specifically, the content-driven audio-visual manipulations are performed strategically to change the sentiment polarity of the whole video. Our baseline method for benchmarking the proposed dataset is a 3DCNN model, termed as Boundary Aware Temporal Forgery Detection (BA-TFD), which is guided via contrastive, boundary matching, and frame classification loss functions. Our extensive quantitative and qualitative analysis demonstrates the proposed method's strong performance for temporal forgery localization and deepfake detection tasks.**

## I. Introduction

Advances in computer vision and deep learning methods (e.g. Autoencoders [1] and Generative Adversarial Networks [2]) have enabled the creation of very realistic fake videos, known as *deepfakes*[1]. There are various ways of creating deepfakes, including voice cloning [3], [4], face reenactment [5], [6], and face swapping [7], [8]. Highly realistic deepfakes are a potential tool for spreading harmful misinformation, given our increasing online presence. This success in generating high-quality deepfakes has raised serious concerns about their role in shaping people's beliefs, with some scholars suggesting that deepfakes are a "threat to democracy" [9], [10], [11], [12]. As an example of the potentially harmful effect of deepfakes, consider the recent work [13] that uses a video of the former United States President Barack Obama to showcase a novel face reenactment method. In this work, the lip movements of Barack Obama are synchronized with another person's speech, resulting in high quality and realistic video in which the former president appears to say something he never did. Given the recent surge in synthesized fake video content on the Internet, it has become increasingly important to identify deepfakes with more accurate and reliable methods.

[1]In the text, *deepfake* and *forgery* are used interchangeably.

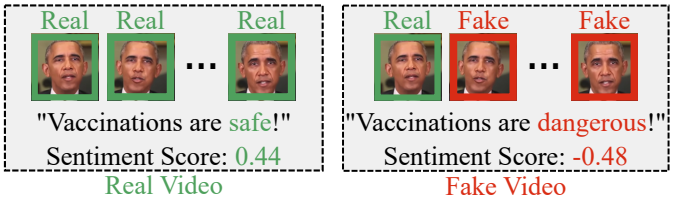


Fig. 1: **Content-driven audio-visual manipulation.** On the left is a real video with the subject saying "Vaccinations are safe". On the right is an audio-visual deepfake created from the real video based on the change in perceived sentiment where "safe" is changed to "dangerous". Green-edge and red-edge images are real and fake frames, respectively. *Through subtle audio-visual manipulation, the whole meaning of the video content has changed.*

This has led to the release of several benchmark datasets [14], [15], [16] and methods [17] for fake content detection. These fake video detection methods aim to correctly classify any given input video as either *real* or *fake*. This suggests that the major assumption behind those datasets and methods is that fake content is present in the entirety of the video/audio signal; that is, there is some form of manipulation throughout the content. And current state-of-the-art deepfake detection methods [18], [19], [20] achieve impressive results on this problem using the largest benchmark datasets.

However, fake content might constitute only a small part of an otherwise long real video, as was initially suggested in [28]. Such short modified segments have the power to alter the meaning and sentiment of the original content completely. For example, consider the manipulation illustrated in Figure 1. The real video might represent a person saying "Vaccinations are safe", while the fake includes only a short modified segment; for example, "safe" is replaced with "dangerous". Hence, the meaning and sentiment of the fake video differ significantly from the real one. If done precisely, this type of coordinated manipulation can sway public opinion (e.g. when employed for media of a famous person as the example with Barack Obama) in a particular direction, for example, based on target sentiment polarity. Given the discussed central assumption behind current datasets and methods, the state-of-the-art deepfake detectors might not perform well on this type of manipulation.

This paper tackles the important task of detecting content altering fake segments in videos. The literature review on benchmark datasets for deepfake detection indicates that there is no dataset suitable for this task, that is, a dataset that

TABLE I: **Comparison of the proposed dataset with other publicly available deepfake datasets.** *Cla: Classification*, *SL: Spatial Localization*, *TFL: Temporal Forgery Localization*, *FS: Face Swapping*, and *RE: ReEnactment*.

| Dataset | Year | Tasks | Manipulated Modality | Manipulation Method | #Subjects | #Real | #Fake | #Total |
|---|---|---|---|---|---|---|---|---|
| DF-TIMIT [14] | 2018 | Cla | V | FS | 43 | 320 | 640 | 960 |
| UADFV [21] | 2019 | Cla | V | FS | 49 | 49 | 49 | 98 |
| FaceForensics++ [15] | 2019 | Cla | V | FS/RE | - | 1,000 | 4,000 | 5,000 |
| Google DFD [22] | 2019 | Cla | V | FS | - | 363 | 3,068 | 3,431 |
| DFDC [16] | 2020 | Cla | AV | FS | 960 | 23,654 | 104,500 | 128,154 |
| DeeperForensics [23] | 2020 | Cla | V | FS | 100 | 50,000 | 10,000 | 60,000 |
| Celeb-DF [24] | 2020 | Cla | V | FS | 59 | 590 | 5,639 | 6,229 |
| WildDeepfake [25] | 2021 | Cla | - | - | - | 3,805 | 3,509 | 7,314 |
| FakeAVCeleb [26] | 2021 | Cla | AV | RE | 600+ | 570 | 25,000+ | 25,500+ |
| ForgeryNet [27] | 2021 | SL/TFL/Cla | V | Random FS/RE | 5400+ | 99,630 | 121,617 | 221,247 |
| LAV-DF (Ours) | 2022 | TFL/Cla | AV | Content-driven RE | 153 | 36,431 | 99,873 | 136,304 |

consists of content-driven manipulations. Therefore, this paper describes the process of creating such a large-scale dataset that will enable further research in this important direction. In addition, we propose a novel multimodal method for precisely predicting the boundaries of fake segments based on visual and audio information. The **main contributions** of our work are as follows,

1) We introduce a new large-scale public audio-visual dataset called *Localized Audio Visual DeepFake*.
2) We propose a new multimodal method called *Boundary Aware Temporal Forgery Detection*.

## II. Related Work

**Deepfake Datasets.** The body of research in deepfake detection is driven by seminal datasets curated with different manipulation methods. A summary of the relevant datasets is presented in Table I. Korshunov and Marcel [14] curated one of the first deepfake datasets, DF-TIMIT, where face-swapping was performed on VidTimit [29]. Down the lane, other important datasets such as UADFV [30], FaceForensics++ [15], and Google DFD [22] were introduced. Due to the complexity of face manipulation and limited availability of open-source face manipulation techniques, these datasets are fairly small in size [24]. Facebook released a large-scale dataset DFDC [16] in 2020 for the task of deepfake classification. Multiple face manipulation methods generated 128,154 videos, including real videos of 3000 actors. DFDC has become a mainstream benchmark dataset for the task of deepfake detection. With the progress in both audio and visual deepfake manipulation, post DFDC, several new datasets including Celeb-DF [24], DeeperForensics [23], and WildDeepFake [25] were introduced. All these datasets are designed for the binary task of deepfake classification and focus primarily on visual manipulation detection [28]. In 2021, OpenForensics [31] dataset was introduced for spatial detection, segmentation and classification. Recently, FakeAVCeleb [26] was released, focusing on both face-swap and face-reenactment methods with manipulated audio and video. ForgeryNet[27] is the latest contribution to the growing list of deepfake detection datasets. This large-scale dataset is also centered around video-only identity manipulation and is suitable for video/image classification and spatial/temporal forgery localization tasks.

All previous datasets provide face manipulations that occur in most of the frames of the video [28]. Only the latest one, ForgeryNet, provides examples of the important problem of temporal forgery localization since it includes random face-swapping applied to parts of some videos. However, the manipulations present in that dataset are only identity modifications that do not necessarily alter the meaning of the content. Our content-driven manipulation dataset addresses this important gap.

**Deepfake Detection.** Deepfake detection methods draw inspiration from observations of artifacts such as different eye colors and unnatural blink and lip-sync issues in deepfake videos. These binary classification approaches are based on both traditional machine learning methods (e.g. EM [32] and SVM [21]) and deep learning methods (e.g. 3DCNN[33], GRU[34] and ViT [20], [19], [18]). Previous methods [35], [36] also aim to detect temporal inconsistencies in deepfake content and recently, several audio-visual deepfake detection methods such as MDS [28] and M2TR [37] were proposed. The methods above are classification centric and do not focus on temporal localization. The only exception is the MDS, shown to work for localization tasks, however, the method is designed primarily for classification. The proposed dataset and method are specifically designed for temporal localization of manipulations.

**Temporal Localization.** Given that the task of temporal forgery localization is similar to the task of temporal action localization, previous work in this area is important. Benchmark datasets in this domain include THUMOS [38] and ActivityNet [39] and the proposed methods can be grouped into two categories: 2-step approaches which first generate segment proposals and then perform multi-class classification to evaluate the proposals [40], [41], [42] and 1-step approaches which directly generate the final segment predictions [43], [44], [45]. For temporal forgery localization, there are no classification requirements for the foreground segments; the background is always real, and the foreground segments are always fake. Therefore, boundary prediction and 1-step approaches are more relevant for our task. Bagchi et al. [46] divided the approaches to segment proposal estimation in temporal action localization into two main categories: methods based on anchors and methods based on predicting the boundary probabilities. As for the anchor-based, these methods mainly use sliding windows in the video, such as S-CNN [47], CDC [48], TURN-TAP [49] and CTAP [50]. As for the methods predicting the boundary probabilities, Lin et al. [51]

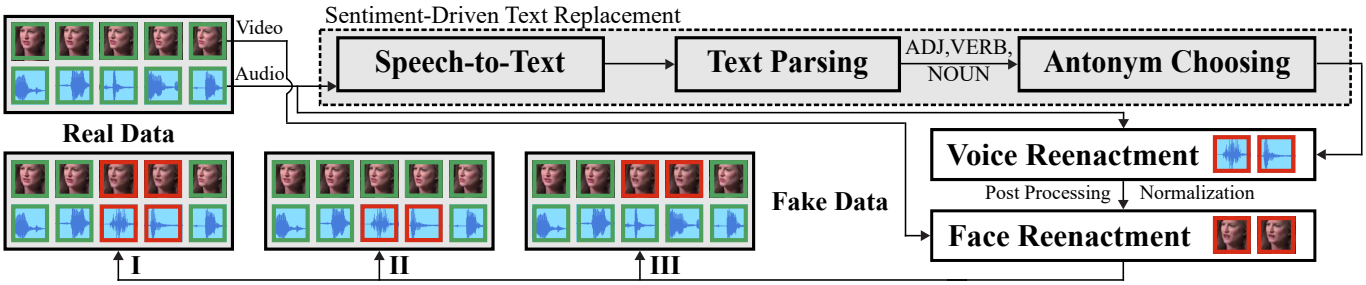


Fig. 2: **Generation pipeline of the proposed dataset**. The green-edge audio and video frames are the real data, and the red-edge audio and video frames are the generated fake data. The real audio-based transcript is used to decide the location and content to be replaced based on the largest change in sentiment. The chosen antonyms are used as input for generating fake audio with voice cloning. The post-processing and normalization are applied to the audio to maintain the consistency of the loudness between the generated audio and real audio in the neighborhood. The generated audio is used as input for facial reenactment. Three categories of data are generated: *I. Fake Audio and Fake Video, II. Fake Audio and Real Video* and *III. Real Audio and Fake Video*. The details on dataset generation are discussed in Section III.

introduced BSN. The method can utilize the global information to overcome the problem that anchor-based methods cannot generate precise and flexible segment proposals. Based on BSN, BMN [52] and BSN++ [53] were introduced for improved performance. It is worth noting that all these methods are unimodal, which is not optimal for the task of temporal forgery localization. The importance of multimodality was demonstrated recently by AVFusion [46].

**Proposed Approach.** For the task of temporal forgery localization, both the audio and visual information are important, in addition to the required precise boundary proposals. In this paper, we introduce a multimodal method based on boundary probabilities and compare its performance with BMN [52], AGT [45], MDS [28] and AVFusion [46].

## III. Proposed Dataset

The proposed dataset Localized Audio Visual DeepFake (LAV-DF) is a large audio-visual deepfake dataset. The main steps in creating the dataset are 1) Sourcing the real videos, 2) Processing the real videos to manipulate their transcripts, and 3) Audio and video synthesis. The deepfake generation is based on the hypothesis that changing relevant words in a transcript can lead to a change in its perception, and in particular, this can be accomplished by changing the sentiment of the transcript. Therefore, the manipulation strategy is to replace strategic words with their antonyms, which leads to a significant change in the sentiment of the statement. The data generation pipeline is illustrated in Figure 2.

**Data Sourcing.** The real videos are sourced from the VoxCeleb2 [54] dataset, a facial video dataset with over 1 million utterance videos of over 6000 speakers. The faces in the videos are tracked and cropped with the facial detector in [55] at 224× 224 resolution. The original dataset contains videos with different duration, spoken language, and voice loudness. Only English-speaking videos are chosen using the confidence score from the Google Speech-to-Text service. The same service generates the transcripts, which are used for manipulation.

### A. Data Generation

**Transcript Manipulation.** After sourcing the real videos, the next step is to analyse a video's transcript denoted by $D = \{d_0, d_1, \cdots, \boldsymbol{d_m}, \cdots, d_n\}$, where $d_i$ denotes word tokens and $n$ is the number of tokens. The aim is to find the tokens to be replaced in $D$ such that the sentiment of the transcript changes the most. In other words, to goal is to create a transcript $D' = \{d_0, d_1, \cdots, \boldsymbol{d'_m}, \cdots, d_n\}$, composed of most of the tokens of $D$ with the exception of a few tokens being replaced. The replacement token $d'_m$ is selected from a set $\hat{d_m}$ of antonyms of $d_m$. The sentiment analyzer in NLTK [56] is used to estimate the sentiment value $S(D)$ of a transcript. For each token $d_i$ in a transcript $D$, we find the replacement as follows,

$$\tau = \underset{d_i \in D, d'_i \in \hat{d_i}}{\operatorname{argmax}} |S(D) - S(D')|$$

We find all the replacements in a transcript $D$ as follows,

$$\theta = \underset{\{\tau_m\}_{m=1}^{M}}{\operatorname{argmax}} |\sum_{i=1}^{M} \Delta S(\tau_i)|$$

where $\Delta S(\tau_i)$ is the sentiment difference with the replacement $\tau_i$ and $M$ is the maximum number of replacements in the transcript. For videos shorter than 10 seconds, there is up to 1 replacement; otherwise, there are up to 2 replacements. Figure 3 (a) illustrates the change in sentiment distribution after the manipulations and Figure 3 (b) presents the histogram of $|\Delta S|$, suggesting that the sentiment of most transcripts was successfully changed.

**Audio Generation.** The next step is to generate the corresponding audio in the speaker's style. Several recent adaptive text-to-speech (TTS) methods [4], [57], [58] which can generate the speech style of a person who is not in the training dataset were evaluated. Based on the better performance, SV2TTS [4] is chosen as the final method for audio generation. The SV2TTS comprises three modules 1) An encoder for extracting style embedding of the reference speaker, 2) Tacotron 2 [59] based spectrogram generated using the replacement tokens and the speaker style embedding, and 3) WaveNet [60] based vocoder for generating realistic audio using the spectrogram. The pre-trained SV2TTS is used for generating the fake audio segments which are later loudness normalized using the corresponding real audio neighbors.

**Video Generation.** The generated fake audio is used as an input for generating the corresponding fake video frames. Wav2Lip [6] facial reenactment is used for this task as it has been shown to have better output quality than previous methods [61], [62], and has better generalization and robustness

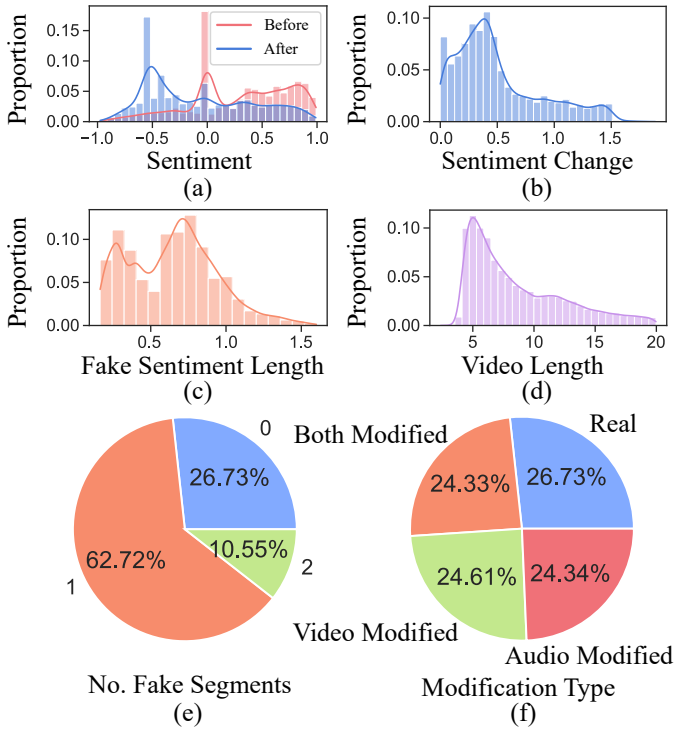


Fig. 3: **Summary of the proposed dataset.** (a) Distribution of sentiment scores, (b) Distribution of sentiment changes, (c) Distribution of fake segment lengths, (d) Distribution of video lengths, (e) Proportion of fake segments, and (f) Proportion of modifications.

to unseen scenarios. It is worth noting that newer methods that achieve better video synthesis quality are not suitable for our task. For example, AD-NeRF [63] is not designed for zero-shot generation of unseen identities, and ATVGNet [64] reenacts the face based on a static reference image, which causes pose inconsistencies on the boundary between fake and real segments. Wav2Lip takes a reference video and target audio as input, generating an output video in which the person in the reference video speaks the target audio content with synced lips. The pre-trained Wav2Lip model is used and the generated fake video segments are up-scaled to a resolution of $224 \times 224$. In the final step, the generated fake audio and video segments are synchronized and used to replace the original audio and video segments.

Similar to [65] the proposed dataset includes three variations of deepfake data,

1) **Fake Audio** and **Fake Video.** The audio and corresponding video are generated for replacement tokens.
2) **Fake Audio** and **Real Video.** Only the audio is generated for replacement tokens and the corresponding real video is length-normalized.
3) **Real Audio** and **Fake Video.** Only the video is generated for replacement tokens and the length of the fake video is normalized to match the real audio.

### B. Dataset Statistics

The dataset contains 136,304 videos, of which 36,431 are completely real, and 99,873 have fake segments, with 153 unique identities. We split the dataset into 3 identity-independent subsets for training (78703 videos of 91 identities), validation (31501 videos of 31 identities), and testing (26100 videos of 31 identities). The summary of the dataset is shown in Figure 3. The total number of fake segments is 114,253, with duration in the range [0-1.6] seconds and an average length of 0.65 seconds, where 89.26% of the segments are shorter than 1 second. The maximum video length is 20 seconds, and 69.61% of the videos are shorter than 10 seconds. As for the modality modification types, the amount of the 4 types (i.e. video-modified, audio-modified, both-modified, real) is approximately equal. In most videos (62.72%), there is 1 fake segment, and in some videos (10.55%), there are 2.

## IV. Proposed Method

The proposed method called Boundary Aware Temporal Forgery Detection (BA-TFD) is illustrated in Figure 4. The first step of the method is to extract features from the input data $X = \{V, A\}$, where $V$ is the video and $A$ is the audio.

### A. Feature Encoders

**Video Encoder.** The goal of the video encoder is to learn frame-level spatio-temporal features from the input video $V$ using a 3DCNN. For that purpose, we designed the video encoder $E_v$ to take the whole video $V \in \mathbb{R}^{C \times T \times H \times W}$ as input, where $T$ is the number of frames, $C$ is the number of channels, and $H$ and $W$ are the height and width of the frame. The output of the $E_v$ are the frame-level features $F_v \in \mathbb{R}^{C_f \times T}$, where $C_f$ is the features dimension. $E_v$ is composed of 4 blocks, each containing multiple 3D convolutional layers with kernel size $3 \times 3 \times 3$ and a final max-pooling layer.

**Audio Encoder.** The goal of the audio encoder is to learn features from the input audio $A$ using a 2DCNN. In addition, the learned audio features are temporarily aligned with the learned video frame-level video features. The first step is to generate the spectrogram $A' \in \mathbb{R}^{F_m \times T_a}$ of the audio signal in log-space, where $T_a$ is the temporal dimension, and $F_m$ is the length of mel-frequency cepstrum features. In the second step, we designed the audio encoder $E_a$ to take the spectrogram $A'$ as input. The output of the $E_a$ are the audio frame features $F_a \in \mathbb{R}^{C_f \times T}$, where $C_f$ is the features dimension. $E_a$ is composed of multiple 2D convolutional layers with kernel size $3 \times 3$ and a final max-pooling layer to reduce the temporal dimension $T_a$ to $T$.

### B. Loss Functions

**Contrastive Loss.** We hypothesize that content modification in one or more modalities will result in miss-synchronization between the modalities (i.e. video and audio), and contrastive loss has been shown [66], [28] to be a powerful objective for similar tasks. Our method uses the audio and video features learned from real videos as positive pairs. The audio and video features learned from videos with at least one modified modality are considered negative pairs. For the positive pairs, the contrastive loss minimizes the difference between the

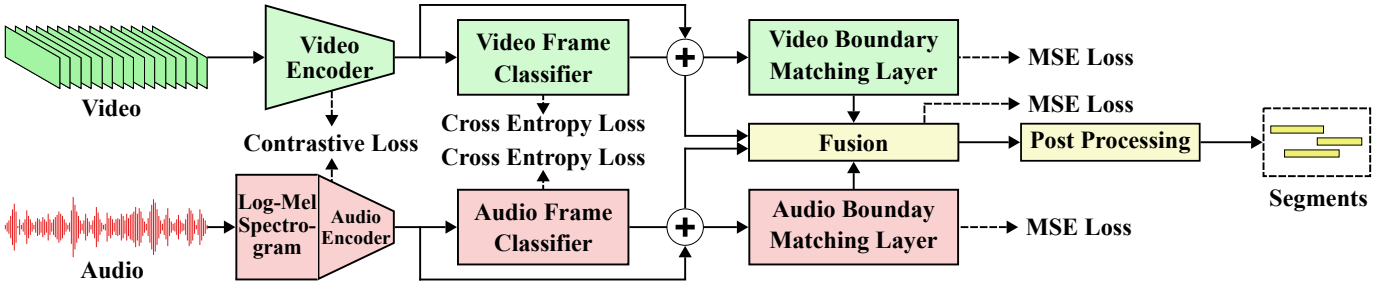


Fig. 4: **Structure of the proposed method.** The video encoder uses raw video as input. The audio encoder uses spectrograms extracted from raw audio. $\oplus$ denotes concatenation. During inference, post-processing is applied to generate segments from the output of the fusion module. The details on different components of the method are discussed in Section IV.

modalities, while for negative pairs, the contrastive loss keeps that margin larger than $\delta$,

$$L_c = \frac{1}{NC_fT}\sum_{i=1}^{N} y_i d_i^2 + (1-y_i)\max(\delta - d_i, 0)^2$$

$$d_i = ||F_{vi} - F_{ai}||_2$$

**Frame Classification Loss.** Since we have access to the frame-level features $F_v$ and $F_a$, we utilize the labels and train the encoders to extract powerful and robust features that capture different deepfake artifacts. For that purpose, we designed two frame-level logistic regression classifiers $FC_v$ and $FC_a$ using $F_v$ and $F_a$ as input. The classifiers consist of 1D convolutional layers and predict the label $\hat{Y}$ as real or fake for each frame and each modality. The classifiers are trained with cross-entropy loss,

$$L_f = -\frac{1}{2NT}\sum_{m\in\{a,v\}}\sum_{i=1}^{N}\sum_{j=t}^{T} H(\hat{Y}_{mij}, Y_{mij})$$

$$H(\hat{Y}, Y) = Y\log\hat{Y} + (1-Y)\log(1-\hat{Y})$$

$$Y_m = \eta_m Y + (1-\eta_m)Y_0$$

where $N$ is the number of samples in the dataset, $T$ is the number of frames, $m$ is the modality (i.e. audio $a$ or video $v$), $\eta_m$ specifies whether modality $m$ is modified, and $Y_0$ is the label for real videos.

**Boundary Matching Loss.** The ground truth boundary maps are generated following the procedure in [52]. Given the fusion boundary map $\hat{M}$, video boundary map $\hat{M}_v$ and audio boundary map $\hat{M}_a$ predicted by the model we use mean squared error as boundary matching loss for $\hat{M}$, $\hat{M}_v$ and $\hat{M}_a$. The fusion boundary matching loss is,

$$L_b = \frac{1}{NDT}\sum_{i=1}^{N}\sum_{j=1}^{D}\sum_{k=1}^{T}(\hat{M}_{ijk} - M_{ijk})^2$$

where, $N$ is the number of samples in the dataset, $D$ is the number of all possible proposal durations and $T$ is the number of frames. The modality boundary matching loss is similar to the frame classification loss,

$$L_{bm} = \frac{1}{2NDT}\sum_{m\in\{v,a\}}\sum_{i=1}^{N}\sum_{j=1}^{D}\sum_{k=1}^{T}(\hat{M}_{mijk} - M_{mijk})^2$$

$$M_m = \eta_m M + (1-\eta_m)M_0$$

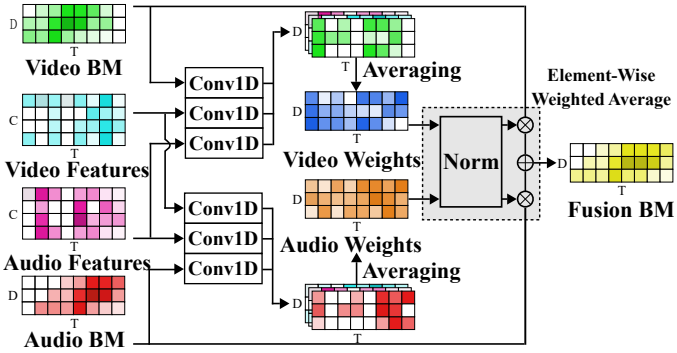


Fig. 5: **Structure of the fusion module.** The gray block normalizes the video and audio weights predicted from the 1D convolutional layers and applies element-wise weighted average. $\oplus$ denotes element-wise addition and $\otimes$ denotes element-wise multiplication. *BM: boundary map.*

where, $m$ is the modality (i.e. video $v$ or audio $a$), $\eta_m$ specifies whether modality $m$ is modified, and $M_0$ is the ground truth boundary map for real videos.

**Overall Loss.** The overall loss is defined as follows,

$$L = \lambda_c L_c + \lambda_f L_f + \lambda_b L_b + \lambda_{bm} L_{bm}$$

where, $\lambda_c$, $\lambda_f$, $\lambda_b$ and $\lambda_{bm}$ are weights for different losses.

### C. Multimodal Fusion

The predictions of $FC_v$ and $FC_a$ are concatenated with the features $F_v$ and $F_a$, and used by two boundary matching layers $B_v$ and $B_a$ [52]. The goal is to predict the boundary maps $\hat{M}_v \in \mathbb{R}^{D\times T}$ and $\hat{M}_a \in \mathbb{R}^{D\times T}$ for the video and audio, where $T$ is the number of frames and $D$ is the maximum duration of the fake segments. The fusion module, illustrated in Figure 5, uses the $\hat{M}_v$, $\hat{M}_a$, $F_v$ and $F_a$ as input. For the video modality, the $\hat{M}_v$, $F_v$ and $F_a$ are used to calculate the video weights $W_v \in \mathbb{R}^{D\times T}$ and for the audio modality, the $\hat{M}_a$, $F_a$ and $F_v$ are used to calculate the audio weights $W_a \in \mathbb{R}^{D\times T}$. In the final step, we perform element-wise weighted average and calculate the fusion boundary map prediction $\hat{M} \in \mathbb{R}^{D\times T}$,

$$\hat{M} = \frac{W_v\hat{M}_v + W_a\hat{M}_a}{W_v + W_a}$$

where all operations are element-wise.

### D. Inference

During inference, the model uses the video and audio as input and generates a fusion boundary map $\hat{M}$. The boundary map represents the confidence for all proposals in the video

TABLE II: **Temporal forgery localization results on the full set (see Section V for details) of the proposed dataset.** The visual-only version of the proposed method uses the output from the video boundary matching layer (see Figure 4 for details), showing the performance when using only the video modality.

| Method | AP@0.5 | AP@0.75 | AP@0.95 | AR@100 | AR@50 | AR@20 | AR@10 |
|---|---|---|---|---|---|---|---|
| MDS [28] | 12.78 | 01.62 | 00.00 | 37.88 | 36.71 | 34.39 | 32.15 |
| AGT [45] | 17.85 | 09.42 | 00.11 | 43.15 | 34.23 | 24.59 | 16.71 |
| BMN [52] | 24.01 | 07.61 | 00.07 | 53.26 | 41.24 | 31.60 | 26.93 |
| BMN (I3D) | 10.56 | 01.66 | 00.00 | 48.49 | 44.39 | 37.13 | 31.55 |
| AVFusion [46] | 65.38 | 23.89 | 00.11 | 62.98 | 59.26 | 54.80 | 52.11 |
| BA-TFD (visual-only) (Ours) | 58.55 | 28.60 | 00.16 | 62.49 | 58.77 | 53.86 | 50.29 |
| BA-TFD (multimodal) (Ours) | **76.90** | **38.50** | **00.25** | **66.90** | **64.08** | **60.77** | **58.42** |

TABLE III: **Temporal forgery localization results on the subset (see Section V for details) of the proposed dataset.** The visual-only version of the proposed method uses the output from the video boundary matching layer (see Figure 4 for details), showing the performance when using only the video modality.

| Method | AP@0.5 | AP@0.75 | AP@0.95 | AR@100 | AR@50 | AR@20 | AR@10 |
|---|---|---|---|---|---|---|---|
| MDS [28] | 23.43 | 03.48 | 00.00 | 58.53 | 56.68 | 53.16 | 49.67 |
| AGT [45] | 15.69 | 10.69 | 00.15 | 49.11 | 40.31 | 31.70 | 23.13 |
| BMN [52] | 32.32 | 11.38 | 00.14 | 59.69 | 48.17 | 39.01 | 34.17 |
| BMN (I3D) | 28.10 | 05.47 | 00.01 | 55.49 | 54.44 | 52.14 | 47.72 |
| AVFusion [46] | 62.01 | 22.77 | 00.11 | 61.98 | 58.08 | 53.31 | 50.52 |
| BA-TFD (visual-only) (Ours) | 83.55 | 41.88 | 00.24 | 65.79 | 62.30 | 57.95 | 55.34 |
| BA-TFD (multimodal) (Ours) | **85.20** | **47.06** | **00.29** | **67.34** | **64.52** | **61.19** | **59.32** |

and is very dense (i.e. there are many duplicated proposals). Similar to BSN [51], we utilize post-processing with Soft Non-Maximum Suppression (S-NMS) [67] to eliminate the duplicated proposals.

## V. Experiments

We have performed extensive benchmarking of the proposed dataset via several state-of-the-art methods including, BMN [52], AGT [45], AVFusion [46], and MDS [28]. Apart from our proposed dataset, we also validate our method for classification on DFDC [16] dataset.

**Dataset Preparation and Evaluation Protocol.** To compare with visual-only methods, we prepare a subset of the test set where the audio-only modified data is removed which is denoted as *subset*. The original test set is denoted as *full set* in the experiments. Unlike temporal action localization methods [42], [45] that are using only average precision, we follow the protocol proposed in ForgeryNet [27] and use both average precision (AP) and average recall (AR) as the evaluation metrics for the quantitative comparison. For AP, we follow the protocol of ActivityNet [39] to set the IoU thresholds to 0.5, 0.75 and 0.95. For AR, as the number of fake segments is small, we set the number of proposals to 100, 50, 20 and 10 with the IoU thresholds [0.5:0.05:0.95]. Our method can also be used for deepfake detection (i.e. classification) task. We use area under the curve (AUC) for evaluation of the deepfake classification.

**Implementation Details.** The proposed method is implemented in PyTorch [68]. For hyperparameters, we set $\lambda_c = 0.1$, $\lambda_f = 2$, $\lambda_b = 1$, $\lambda_{bm} = 1$ and $\delta = 0.99$. For comparison, we trained BMN [52], AGT [45], AVFusion [46] and MDS [28] for temporal forgery localization task. In addition, to evaluate the usefulness of the proposed method, we compare with MDS, EfficientViT [18] and other methods on classification task. We followed the original settings for BMN, AGT, MDS and EfficientViT, and used encoding concatenation fusion for AVFusion. For the methods that require pre-trained features, we trained them end-to-end with trainable encoder. For comparison, we also trained BMN with I3D features [69] (i.e. fixed encoder). For the models which require S-NMS [67] post-processing, we used the validation set to search for optimal parameters for post-processing. Final evaluation and results are based on the test set. For DFDC, we consider the whole fake video as one fake segment. For evaluation, we used 2 methods to generate the classification output for our method 1) Using the highest confidence of the predicted segments as the confidence of the video being fake and 2) Training a MLP classifier using the confidences of predicted segments. We chose evaluation method 1) for our dataset and method 2) for DFDC based on performance.

## VI. Results

**Temporal Forgery Localization.** We compare our method on the full set of the proposed dataset with the latest methods for temporal action localization and deepfake detection. From Table II, our method achieves the best performance, which is 76.9 for AP@0.5 and 66.9 for AR@100. Unlike temporal action localization datasets, in our dataset there is a single label for the fake segments, so it is reasonable that the AP score is relatively high. The multimodal MDS method is not designed for temporal forgery localization tasks and predicts only fixed length segments (i.e. cannot predict the precise boundaries), hence the scores for that method are low. As for AGT and BMN, the scores are low because they are visual-only unimodal methods and cannot detect the fake segments in videos where only the audio is modified. We also evaluated the performance of our visual-only unimodal method, which shows worse results than the multimodal version and AVFusion. In addition, the results show that when the video encoder is trained with data from the proposed dataset, BMN performs significantly better than using I3D features. We also evaluated the same methods on the subset of the proposed dataset. From Table III, the performance of the visual-only methods is improved, and for our method, the visual-only

TABLE IV: **Temporal forgery localization results on the full set (see Section V for details) of the proposed dataset.** The contribution of different loss terms in the proposed method (see Section IV for details).

| Loss Function | AP@0.5 | AP@0.75 | AP@0.95 | AR@100 | AR@50 | AR@20 | AR@10 |
|---|---|---|---|---|---|---|---|
| $L_f$ | 40.50 | 29.74 | 00.13 | 60.51 | 60.50 | 60.47 | 59.90 |
| $L_c, L_f$ | 40.92 | 31.23 | 00.74 | 64.71 | 64.71 | 64.36 | 62.79 |
| $L_b$ | 53.16 | 11.91 | 00.02 | 53.99 | 50.94 | 47.74 | 45.55 |
| $L_{bm}, L_b$ | 54.70 | 15.50 | 00.04 | 56.64 | 53.57 | 49.46 | 45.85 |
| $L_f, L_{bm}, L_b$ | 76.50 | **39.92** | 00.18 | 66.69 | 63.71 | 60.07 | 57.76 |
| $L_c, L_f, L_{bm}, L_b$ | **76.90** | 38.50 | **00.25** | **66.90** | **64.08** | **60.77** | **58.42** |

score improves from 58.55 (AP@0.5) to 83.55 (AP@0.5) and the margin between the unimodal and multimodal versions is decreased from 18.35 (AP@0.5) to 1.65 (AP@0.5). Overall, our method still ranks first, which demonstrates it's superior performance for temporal forgery detection.

**Deepfake Classification.** We also compare our method with previous deepfake detection methods on the full set of the proposed dataset and a subset of DFDC. On our dataset, our method (**0.990**) outperforms F3Net [70] (0.520), MDS (0.828) and EfficientViT (0.965). As for the subset of DFDC, the performance of our method (**0.846**) is better than previous methods such as Meso4 [71] (0.753), FWA [72] (0.727) and [73] (0.844) and is close to MDS (0.916). It is worth noting that, our method is not designed and trained for classification task with classification loss. It is trained for temporal forgery localization and then the segment outputs are summarized as a whole video label prediction. Therefore, the performance of our method on DFDC drops as compared to the state-of-the-art classification method MDS. On the other hand, previous deepfake detection methods assume that fake videos are entirely fake, so their performance (e.g. the frame-based approach of F3Net) is reduced on our dataset. In summary, our method still performs well on classification task and has potential to reach the state-of-the-art performance.

**Impact of Loss Functions.** From Table IV, all loss terms have positive effect on the performance of the proposed model. The results suggest that the frame classification loss contributes the most to the method performance.

**Failure Analysis.** The output of the proposed method can be noisy for cases that contain very short video manipulations ($\leq 0.5$ sec) and the corresponding real audio. For such short video-only manipulations, if the visual transition from real to fake and then back to real is smooth, it may lead to noisy output.

## VII. Conclusion

This work introduces and investigates a novel problem related to content-driven deepfake generation and detection. To this end, we propose a new dataset in which the audio and video are modified at specific locations based on the change in sentiment of the content. We also propose a new method for temporal forgery localization in such partially modified videos. The conducted experiments show that our method achieves better performance than previous relevant state-of-the-art methods.

**Ethical Concerns.** The proposed dataset potentially might have a negative social impact. Since the individuals in the dataset are celebrities, the content in the dataset may be used for unethical purposes such as making fake rumours. Also, the dataset generation pipeline can be used to create fake videos. To encounter the potential negative impact of our work, we prepared a license for public usage of the dataset and proposed the method.

**Limitations.** This work has some limitations 1) The audio reenactment method used in the dataset does not always generate the reference style, 2) The resolution of the dataset is constrained on the basis of source videos and 3) The high score of classification results indicates the necessity of improving the video reenactment method.

**Future Work.** Major improvement in the future will be increasing the dataset with new token insertion, substitution and deletion of existing tokens and converting statements into questions.

## References


[1] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," Tech. Rep., 1985.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[3] Y. Wang, R. J. Skerry-Ryan, D. Stanton, and et al., "Tacotron: Towards End-to-End Speech Synthesis," *arXiv:1703.10135 [cs]*, 2017.

[4] Y. Jia, Y. Zhang, R. J. Weiss, and et al., "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *NeurIPS*. Curran Associates Inc., 2018, pp. 4485–4495.

[5] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "MoCoGAN: Decomposing Motion and Content for Video Generation," in *CVPR*, 2018, pp. 1526–1535.

[6] 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 *ACM MM*, 2020, pp. 484–492.

[7] I. Korshunova, W. Shi, J. Dambre, and L. Theis, "Fast Face-Swap Using Convolutional Neural Networks," in *ICCV*, 2017, pp. 3677–3685.

[8] Y. Nirkin, Y. Keller, and T. Hassner, "FSGAN: Subject Agnostic Face Swapping and Reenactment," in *ICCV*, 2019, pp. 7184–7193.

[9] O. Schwartz, "You thought fake news was bad? Deep fakes are where truth goes to die," *The Guardian*, 2018.

[10] J. Brandon, "There Are Now 15,000 Deepfake Videos on Social Media. Yes, You Should Worry." *Forbes*, 2019.

[11] I. Sample, "What are deepfakes – and how can you spot them?" *The Guardian*, 2020.

[12] D. Thomas, "Deepfakes: A threat to democracy or just a bit of fun?" *BBC News*, 2020.

[13] J. Thies, M. Elgharib, A. Tewari, C. Theobalt, and M. Nießner, "Neural Voice Puppetry: Audio-Driven Facial Reenactment," in *ECCV 2020*, 2020, pp. 716–731.

[14] P. Korshunov and S. Marcel, "DeepFakes: a New Threat to Face Recognition? Assessment and Detection," *arXiv:1812.08685 [cs]*, 2018.

[15] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "FaceForensics++: Learning to Detect Manipulated Facial Images," in *ICCV*, 2019, pp. 1–11.

[16] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The DeepFake Detection Challenge (DFDC) Dataset," *arXiv:2006.07397 [cs]*, 2020.

[17] Y. Mirsky and W. Lee, "The Creation and Detection of Deepfakes: A Survey," *ACM Computing Surveys*, vol. 54, no. 1, pp. 7:1–7:41, 2021.

[18] D. A. Coccomini, N. Messina, C. Gennaro, and F. Falchi, "Combining EfficientNet and Vision Transformers for Video Deepfake Detection," in *ICIAP 2022*, 2022, pp. 219–229.

[19] Y.-J. Heo, Y.-J. Choi, Y.-W. Lee, and B.-G. Kim, "Deepfake Detection Scheme Based on Vision Transformer and Distillation," *arXiv:2104.01353 [cs]*, 2021.

[20] D. Wodajo and S. Atnafu, "Deepfake Video Detection Using Convolutional Vision Transformer," *arXiv:2102.11126 [cs]*, 2021.

[21] X. Yang, Y. Li, and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses," in *ICASSP*, 2019, pp. 8261–8265.

[22] D. Nick and J. Andrew, "Contributing Data to Deepfake Detection Research," 2019.

[23] L. Jiang, R. Li, W. Wu, C. Qian, and C. C. Loy, "DeeperForensics-1.0: A Large-Scale Dataset for Real-World Face Forgery Detection," in *CVPR*, 2020, pp. 2889–2898.

[24] Y. Li, X. Yang, P. Sun, and et al., "Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics," in *CVPR*, 2020, pp. 3207–3216.

[25] B. Zi, M. Chang, J. Chen, and et al., "WildDeepfake: A Challenging Real-World Dataset for Deepfake Detection," in *ACM MM*, 2020, pp. 2382–2390.

[26] H. Khalid, S. Tariq, and S. S. Woo, "FakeAVCeleb: A Novel Audio-Video Multimodal Deepfake Dataset," *arXiv:2108.05080 [cs]*, 2021.

[27] Y. He, B. Gan, S. Chen, Y. Zhou, G. Yin, L. Song, L. Sheng, J. Shao, and Z. Liu, "ForgeryNet: A Versatile Benchmark for Comprehensive Forgery Analysis," in *CVPR*, 2021, pp. 4360–4369.

[28] K. Chugh, P. Gupta, A. Dhall, and R. Subramanian, "Not made for each other- Audio-Visual Dissonance-based Deepfake Detection and Localization," in *ACM MM*, 2020, pp. 439–447.

[29] C. Sanderson and B. C. Lovell, "Multi-region probabilistic histograms for robust and scalable identity inference," in *Advances in Biometrics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 199–208.

[30] K. Yang, P. Qiao, D. Li, S. Lv, and Y. Dou, "Exploring Temporal Preservation Networks for Precise Temporal Action Localization," *AAAI*, vol. 32, no. 1, 2018.

[31] T.-N. Le, H. H. Nguyen, J. Yamagishi, and I. Echizen, "OpenForensics: Large-Scale Challenging Dataset for Multi-Face Forgery Detection and Segmentation In-the-Wild," in *ICCV*, 2021, pp. 10 117–10 127.

[32] L. Guarnera, O. Giudice, and S. Battiato, "DeepFake Detection by Analyzing Convolutional Traces," in *CVPRW*, 2020, pp. 666–667.

[33] O. de Lima, S. Franklin, S. Basu, B. Karwoski, and A. George, "Deepfake Detection using Spatiotemporal Convolutional Networks," *arXiv:2006.14749 [cs, eess]*, 2020.

[34] D. M. Montserrat, H. Hao, and et al., "Deepfakes Detection With Automatic Face Weighting," in *CVPRW*, 2020, pp. 668–669.

[35] J. K. Lewis, I. E. Toubal, H. Chen, and et al., "Deepfake Video Detection Based on Spatial, Spectral, and Temporal Inconsistencies Using Multimodal Deep Learning," in *AIPR*, 2020, pp. 1–9.

[36] Z. Gu, Y. Chen, T. Yao, S. Ding, J. Li, F. Huang, and L. Ma, "Spatiotemporal Inconsistency Learning for DeepFake Video Detection," in *ACM MM*, 2021, pp. 3473–3481.

[37] J. Wang, Z. Wu, J. Chen, and Y.-G. Jiang, "M2TR: Multi-modal Multi-scale Transformers for Deepfake Detection," *arXiv:2104.09770 [cs]*, 2021.

[38] H. Idrees, A. R. Zamir, Y.-G. Jiang, and et al., "The THUMOS Challenge on Action Recognition for Videos "in the Wild"," *Computer Vision and Image Understanding*, vol. 155, pp. 1–23, 2017.

[39] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding," in *CVPR*, 2015, pp. 961–970.

[40] R. Zeng, W. Huang, M. Tan, and et al., "Graph Convolutional Networks for Temporal Action Localization," in *ICCV*, 2019, pp. 7094–7103.

[41] M. Xu, C. Zhao, D. S. Rojas, and et al., "G-TAD: Sub-Graph Localization for Temporal Action Detection," in *CVPR*, 2020, pp. 10 156–10 165.

[42] X. Liu, Y. Hu, S. Bai, and et al., "Multi-Shot Temporal Event Localization: A Benchmark," in *CVPR*, 2021, pp. 12 596–12 606.

[43] T. Lin, X. Zhao, and Z. Shou, "Single Shot Temporal Action Detection," in *ACM MM*, 2017, pp. 988–996.

[44] S. Buch, V. Escorcia, B. Ghanem, and et al., "End-to-end, single-stream temporal action detection in untrimmed videos," *BMVC*, 2019.

[45] M. Nawhal and G. Mori, "Activity Graph Transformer for Temporal Action Localization," *arXiv:2101.08540 [cs]*, 2021.

[46] A. Bagchi, J. Mahmood, D. Fernandes, and R. K. Sarvadevabhatla, "Hear Me Out: Fusional Approaches for Audio Augmented Temporal Action Localization," *arXiv:2106.14118 [cs]*, 2021.

[47] Z. Shou, D. Wang, and S.-F. Chang, "Temporal Action Localization in Untrimmed Videos via Multi-Stage CNNs," in *CVPR*, 2016, pp. 1049–1058.

[48] Z. Shou, J. Chan, A. Zareian, K. Miyazawa, and S.-F. Chang, "CDC: Convolutional-De-Convolutional Networks for Precise Temporal Action Localization in Untrimmed Videos," in *CVPR*, 2017, pp. 5734–5743.

[49] J. Gao, Z. Yang, K. Chen, C. Sun, and R. Nevatia, "TURN TAP: Temporal Unit Regression Network for Temporal Action Proposals," in *ICCV*, 2017, pp. 3628–3636.

[50] J. Gao, K. Chen, and R. Nevatia, "CTAP: Complementary Temporal Action Proposal Generation," in *ECCV*, 2018, pp. 68–83.

[51] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "BSN: Boundary Sensitive Network for Temporal Action Proposal Generation," in *ECCV*, 2018, pp. 3–19.

[52] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "BMN: Boundary-Matching Network for Temporal Action Proposal Generation," in *ICCV*, 2019, pp. 3889–3898.

[53] H. Su, W. Gan, W. Wu, Y. Qiao, and J. Yan, "BSN++: Complementary Boundary Regressor with Scale-Balanced Relation Modeling for Temporal Action Proposal Generation," *arXiv:2009.07641 [cs]*, 2021.

[54] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," in *INTERSPEECH*, 2018.

[55] D. E. King, "Dlib-ml: A Machine Learning Toolkit," *The Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[56] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc., 2009.

[57] E. Casanova, C. Shulby, E. Gölge, and et al., "SC-GlowTTS: an Efficient Zero-Shot Multi-Speaker Text-To-Speech Model," *arXiv:2104.05557 [cs, eess]*, 2021.

[58] P. Neekhara, S. Hussain, S. Dubnov, F. Koushanfar, and J. McAuley, "Expressive Neural Voice Cloning," in *ACML*, 2021, pp. 252–267.

[59] J. Shen, R. Pang, R. J. Weiss, and et al., "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," in *ICASSP*, 2018, pp. 4779–4783.

[60] A. v. d. Oord, S. Dieleman, H. Zen, and et al., "WaveNet: A Generative Model for Raw Audio," *arXiv:1609.03499 [cs]*, 2016.

[61] A. Jamaludin, J. S. Chung, and A. Zisserman, "You Said That?: Synthesising Talking Faces from Audio," *International Journal of Computer Vision*, vol. 127, no. 11, pp. 1767–1779, 2019.

[62] P. K R, R. Mukhopadhyay, J. Philip, A. Jha, V. Namboodiri, and C. V. Jawahar, "Towards Automatic Face-to-Face Translation," in *ACM MM*, 2019, pp. 1428–1436.

[63] Y. Guo, K. Chen, S. Liang, Y.-J. Liu, H. Bao, and J. Zhang, "AD-NeRF: Audio Driven Neural Radiance Fields for Talking Head Synthesis," in *ICCV*, 2021, pp. 5784–5794.

[64] L. Chen, R. K. Maddox, Z. Duan, and C. Xu, "Hierarchical Cross-Modal Talking Face Generation With Dynamic Pixel-Wise Loss," in *CVPR*, 2019, pp. 7832–7841.

[65] H. Khalid, M. Kim, S. Tariq, and S. S. Woo, "Evaluation of an Audio-Video Multimodal Deepfake Dataset using Unimodal and Multimodal Detectors," *ADGD*, pp. 7–15, 2021.

[66] J. S. Chung and A. Zisserman, "Out of Time: Automated Lip Sync in the Wild," in *ACCV Workshops*, 2017, pp. 251–263.

[67] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS – Improving Object Detection With One Line of Code," in *ICCV*, 2017, pp. 5561–5569.

[68] A. Paszke, S. Gross, F. Massa, and et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in NeurIPS*, vol. 32, 2019.

[69] J. Carreira and A. Zisserman, "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset," in *CVPR*, 2017, pp. 6299–6308.

[70] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in Frequency: Face Forgery Detection by Mining Frequency-Aware Clues," in *ECCV*, 2020, pp. 86–103.

[71] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: a Compact Facial Video Forgery Detection Network," in *WIFS*, 2018, pp. 1–7.

[72] Y. Li and S. Lyu, "Exposing DeepFake Videos By Detecting Face Warping Artifacts," in *CVPRW*, 2019, p. 7.

[73] T. Mittal, U. Bhattacharya, R. Chandra, A. Bera, and D. Manocha, "Emotions Don't Lie: An Audio-Visual Deepfake Detection Method using Affective Cues," in *ACM MM*, 2020, pp. 2823–2832.