Integrating Frame-Level Boundary Detection and Deepfake Detection for Locating Manipulated Regions in Partially Spoofed Audio Forgery Attacks

Zexin Cai[a], Ming Li[a,b,*]

[a]*Department of Electrical and Computer Engineering, Duke University, Durham, NC, United States*
[b]*Suzhou Municipal Key Laboratory of Multimodal Intelligent Systems, Duke Kunshan University, Kunshan, China*

Abstract

Partially fake audio, a variant of deep fake that involves manipulating audio utterances through the incorporation of fake or externally-sourced bona fide audio clips, constitutes a growing threat as an audio forgery attack impacting both human and artificial intelligence applications. Researchers have recently developed valuable databases to aid in the development of effective countermeasures against such attacks. While existing countermeasures mainly focus on identifying partially fake audio at the level of entire utterances or segments, this paper introduces a paradigm shift by proposing frame-level systems. These systems are designed to detect manipulated utterances and pinpoint the specific regions within partially fake audio where the manipulation occurs. Our approach leverages acoustic features extracted from large-scale self-supervised pre-training models, delivering promising results evaluated on diverse, publicly accessible databases. Additionally, we study the integration of boundary and deepfake detection systems, exploring their potential synergies and shortcomings. Importantly, our techniques have yielded impressive results. We have achieved state-of-the-art performance on the test dataset of the Track 2 of ADD 2022 challenge with an equal error rate of 4.4%. Furthermore, our methods exhibit remarkable performance in locating manipulated regions in Track 2 of the ADD 2023 challenge, resulting in a final ADD score of 0.6713 and securing the top position.

Keywords: Audio splicing forgery, waveform boundary detection, audio deepfake detection challenge

1. Introduction

Digital content, including text, images, and audio, plays a pivotal role in facilitating communication and disseminating knowledge among humans. These forms of content can also serve as evidence in judicial proceedings [1]. However, as deep learning continues to advance rapidly, there are now numerous methods available to generate and manipulate digital data. While these techniques were originally developed to provide a wide range of benefits and opportunities to humanity, they also pose a significant threat to societal security [2, 3, 4]. Notably, there has been

*Corresponding Author: Ming Li
Email addresses: zexin.cai@duke.edu (Zexin Cai), ming.li369@duke.edu (Ming Li)

an increase in the accessibility of deep-generation applications to the general public [5, 6, 7]. These sophisticated tools have the ability to generate synthetic outputs that closely resemble human-generated content, posing a challenge in verifying their authenticity and distinguishing them from genuine materials. Consequently, the misuse of such applications has the potential to tamper with multimedia content and propagate false messages, amplifying the need for scrutiny and verification [8].

Nevertheless, the development of corresponding countermeasures for detecting fraudulent content is relatively slow and significantly lagging behind. Regarding speech synthesis, the two primary approaches, namely text-to-speech and voice conversion, have achieved remarkable levels of naturalness and similarity with various deep architectures such as Tacotron-based [9, 10], Fastspeech-based [11, 12], and VAE-based models [7]. However, this advancement also opens up the possibility of launching audio spoofing attacks using any of these approaches, thus making the detection of such attacks more difficult. In this context, researchers have dedicated their efforts to designing anti-spoofing speech detection systems [13, 14, 15], while comparatively less attention has been given to spoofing algorithm detection [16, 17, 18]. While existing literature has demonstrated impressive performance in anti-spoofing speech detection, it is important to note that most deep learning models are trained and evaluated on specific systems and datasets, limiting their discriminatory ability when faced with unseen scenarios and mismatched domains [19]. This includes detecting attacks generated through synthesis approaches not encountered during training. As a result, the development of a universal anti-spoofing system still requires further research and advancements.

One of the most challenging scenarios in anti-spoofing speech attacks is audio splicing or tampering forgery. Audio splicing forgery refers to techniques used to manipulate audio recordings by cutting, merging, or combining different sections of audio from multiple sources [20, 21]. This involves operations such as insertion, deletion, and substitution of specific audio segments to create a seamless and coherent audio file [22]. Accordingly, fraudsters could exploit these techniques to engage in deceptive practices, such as creating fake conversations, altering speech content to misrepresent information, and generating misleading evidence. Nevertheless, the advancement of deep speech synthesis has made this type of speech spoofing more difficult to detect, as the inserted clips can now be high-fidelity audio segments that maintain the exact same voice as the genuine speech. This presents a more intricate scenario where manipulation can be achieved using either other genuine audio segments or high-fidelity synthesized ones. The research community has recently acknowledged this emerging scenario, referring to it as partially fake audio attacks [23, 24]. Subsequently, several methodologies have been developed to detect such attacks and, in some cases, even to identify and segment potentially falsified regions within the manipulated utterances [25, 26, 27].

However, existing systems in the literature still have limitations. Most research predominantly concentrates on the detection of spoofed audio at the utterance or segment level, without being able to locate the precise manipulated part within spoofed utterances. On the other hand, available techniques that can identify regions in partially spoofed utterances often lack the capability to authenticate each individual region. As such, there remains room for enhancing their performance [28].

Concerning the challenges posed by the emerging partially fake audio attacks, this paper presents our novel frame-level detection systems, building upon our previous research [27]. These systems encompass a boundary detection system and a deepfake/spoofing detection system, both of which are extensively trained and evaluated on multiple publicly available datasets, including those from ADD challenges and the PartialSpoof database. Experimental findings

demonstrate the effectiveness of our proposed systems in detecting partially spoofed audio and accurately localizing the manipulated regions at the frame level. Notably, our approach achieves state-of-the-art performance on the test dataset of ADD2022 Track 2, boasting an impressive EER of 4.4%, and secures the first position in Track 2 of the ADD 2023 challenge. The main contributions of this paper are as follows:

- Proposing effective frame-level systems capable of detecting partially spoofed audio and locating the corresponding manipulated regions.
- Exploring model integration techniques to facilitate the practical use of spoofing detection models in identifying fake regions.
- Investigating the performance of various large-scale self-supervised pre-training models in countering partially fake attacks.

The remaining sections of the paper are structured as follows: Section 2 presents an overview of related works from the existing literature. Section 3 introduces available databases for the study of partially spoofed anti-spoofing. Section 4 details our proposed approach for detecting and locating fake regions. Our experimental setup and corresponding results are presented and analyzed in Section 5. We discuss our work's limitations and potential future directions in Section 6. Finally, in Section 7, we conclude our study.

2. Related Works

Conventional techniques rely on various speech signal processing algorithms to analyze the inconsistencies introduced during the audio tampering process. Researchers have explored different acoustic properties and statistical patterns to differentiate genuine speech utterances from manipulated ones. One approach for audio authenticity verification is based on the electric network frequency (ENF) criterion, which leverages artifacts caused by electronic circuits in recording devices [29, 30]. Yang et al. presented a technique that explicitly investigates the discontinuity of frame offsets in the context of the MP3 encoding scheme [31]. Furthermore, audio reverberation has been estimated as an audio environmental feature in forensic settings [32]. Subsequent studies have extended this exploration to encompass other environmental features, such as background noise level and inconsistency [16, 33, 34], aiming to detect spliced digital audio.

Subsequently, machine learning methods, particularly deep learning, have emerged as dominant approaches in this field due to their remarkable efficacy. Jadhav et al. conducted a pilot study that employed deep learning for audio splicing detection [22]. They utilize convolutional neural networks (CNN) and achieve high detection accuracy, demonstrating robustness against noise attacks and compression. However, their proposed classification model is limited to determining if an audio clip has been tampered with and does not provide localization of the spliced segment.

Consequently, several studies have focused on developing methods to precisely localize splicing segments. Various modern neural architectures, such as the encoder-decoder-based ASLNet [35], ResNet-based models [36], and Conformer-based models [37], have been explored, demonstrating their ability to capture distinguishable patterns for detecting spliced boundaries. These models exhibit promising results in boundary detection, particularly when applied at the chunk level. The chunk sizes vary, with the ASLNet-based model utilizing 1-second chunks, the Conformer-based model using 0.256-second chunks, and the ResNet-based

model employing 0.6-second chunks. Despite the ResNet-based model proposed by Zeng et al. being capable of frame-level detection with a 40ms frame length, its performance does not entirely match the proficiency observed in chunk-level detection [36].

In addition, these deep learning approaches have mainly been trained and assessed on self-constructed datasets, hindering uniform comparison and evaluation across studies. To address this gap, publicly accessible datasets have been developed to encourage research on audio splicing within the context of high-fidelity synthesis techniques. Zhang et al. introduce the PartiallySpoof database [24], which contains manipulated bona fide utterances embedded with synthesized audio segments from the ASVspoof 2019 logical access (LA) database. They also provide a benchmark model using linear frequency cepstral coefficients (LFCC) and Light Convolutional Neural Networks (LCNN) [25].

Another database, named Half-Truth, is specifically designed to encompass additional scenarios allowing for splicing based on both genuine and fake clips [23]. Figure 1 illustrates an example of a partially fake spoofed utterance generated by manipulating bona fide utterances with synthesized and externally-sourced genuine audio segments. The Half-Truth database is publicly available through the Audio Deep Synthesis Detection Challenge (ADD) [38]. In Track 2 of ADD 2022, which represents the first challenge task designed for segment-level partially spoofed audio detection, researchers explore large-scale self-supervised pre-training models and demonstrate their impressive performance in comparison to traditional acoustic features like Mel frequency cepstral coefficients (MFCC) and LFCC [26, 39, 27]. Lv et al. participated in the challenge and achieved the highest performance with an utterance-level detection system that was fine-tuned using large-scale pre-training models [39]. The effectiveness of acoustic features extracted by self-supervised models is also proven on the PartiallySpoof database, achieving an utterance-level equal error rate (EER) of 0.49%, which outperforms models relying on conventional front-end acoustic features [40].

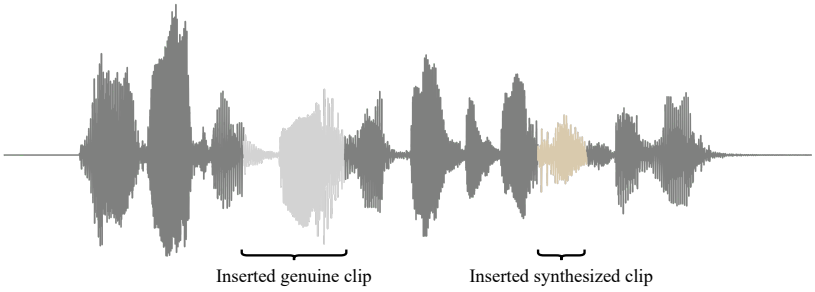

Figure 1: An illustrative example of partially spoofed speech

While approaches for utterance-level partial spoofing detection have demonstrated impressive performance, the study of splicing segment localization is recognized as a more challenging and crucial task. Regarding Track 2 of ADD 2022 challenge, systems based on frame-level detection have already enabled the identification of concatenation boundaries, including our previous boundary detection system and the Fake Span Discovery system developed by Wu et al [41]. In addition to detecting partially fake audio, Track 2 of ADD 2023 emphasizes the localization of manipulated regions within partially spoofed utterances [28].

3. Databases

There are currently limited publicly available datasets for partially spoofed audio. We have access to three databases that can be used for the study: ADD2022-T2 [38], ADD2023-T2 [28], and PartialSpoof [24]. While ADD2022-T2 and ADD2023-T2 contain Mandarin Chinese audio utterances, the PartialSpoof dataset comprises English audio utterances. The detailed statistics of these databases are outlined in Table 1, wherein the category of 'Bona fide' utterances pertains to natural, unaltered utterances, 'Fake' utterances indicate synthetically generated utterances produced by text-to-speech and voice conversion systems, and 'PartialFake' utterances refer to those that are partially manipulated.

Table 1: The statistics of the ADD2022 Track 2 database, the ADD2023 Track 2 database, and the PartialSpoof database, '-' denotes unknown number

Name	**#Utterances**				**Duration (h)**	**Audio length (s)**	
	Bona fide	Fake	PartialFake	All		min - max	mean / std
ADD2022-T2-Train	3,012	24,072	35,808	62,892	47.33	0.96 - 60.01	3.15 / 2.00
ADD2022-T2-Adapt	1,052	0	1,052	2,104	2.22	1.15 - 13.41	3.79 / 1.68
ADD2022-T2-Test	-	-	-	100,625	166.29	1.07 - 158.14	5.95 / 4.87
ADD2023-T2-Train	26,554	1,185	25,354	53,093	53.39	0.87 - 14.77	3.62 / 1.44
ADD2023-T2-Dev	8,914	430	8,480	17,824	17.40	0.82 - 13.34	3.51 / 1.31
ADD2023-T2-Test	20,000	-	-	50,000	56.45	0.77 - 153.81	4.06 / 3.14
PartialSpoof-Train	2,580	0	22,800	25,380	24.25	0.60 - 21.02	3.44 / 1.56
PartialSpoof-Dev	2,548	0	22,296	24,844	24.34	0.62 - 15.34	3.53 / 1.63
PartialSpoof-Eval	7,355	0	63,882	71,237	67.68	0.48 - 18.20	3.42 / 1.70

3.1. The ADD2022-T2 database

The ADD2022-T2 (ADD2022 Track 2) database, consists of three subsets: the training, adaptation, and test sets. The training set, referred to as ADD2022-T2-Train, is utilized for model training. The adaptation set, named as ADD2022-T2-Adapt, is used for performance evaluation and model selection. The test set, denoted as ADD2022-T2-Test, serves as an evaluation set containing out-of-domain data. Within the ADD2022-T2-Train set, there are 3,012 bona fide utterances and 24,072 fake utterances. Additionally, 35,808 utterances are constructed by combining bona fide and fake utterances, as explained in our previous work [27]. Only bona fide utterances and partially fake utterances are used for model training.

The ADD2022-T2 development set, though not displayed in the table, shares similarities with ADD2022-T2-Train in terms of number of utterances. However, it does not contain partially fake utterances. Conversely, the adaptation dataset, provided by the official challenge, contains 1,052 partially fake utterances but lacks bona fide ones. To ensure fair evaluation, we randomly select 1,052 bona fide utterances from the ADD2022-T2 development set. These selected utterances, combined with the original adaptation dataset, constitute the ADD2022-T2-Adapt dataset. This approach guarantees that all evaluation utterances are sourced from the challenge itself. Both ADD2022-T2-Train and ADD2022-T2-Adapt datasets have utterance-level labels. However, the ADD2022-T2-Test set comprises 100,625 utterances but lacks labeling information. The distribution of utterances within this set is unknown. The performance evaluation on ADD2022-T2-Test can be exclusively obtained using the open-source platform CodaLab[1].

[1]https://codalab.lisn.upsaclay.fr/competitions/4111

3.2. The ADD2023-T2 database

Unlike the ADD2022 Track 2 challenge, ADD2023 Track 2 challenge provides partially fake utterances, which are utilized in our study. The ADD2023-T2 database comprises three subsets: ADD2023-T2-Train, ADD2023-T2-Dev, and ADD2023-T2-Test. The ADD2023-T2-Train subset contains 26,554 genuine utterances, 1,185 fake utterances, and 25,354 manipulated utterances. In the ADD2023-T2-Dev dataset, we have 8,914 genuine utterances, 430 fake utterances, and 17,824 partially fake utterances. Since Track 2 of ADD2023 specifically focuses on locating the manipulated regions, both utterance-level and frame-level labeling information is available for ADD2023-T2-Train and ADD2023-T2-Dev. Lastly, the ADD2023-T2 database includes the ADD2023-T2-Test dataset, consisting of 50,000 unlabelled utterances without any provided labeling information. Due to the absence of labeling information for ADD2023-T2-Test, we are unable to analyze results on this dataset. As a result, we have excluded ADD2023-T2-Test as our evaluation dataset in our experiments.

3.3. The PartialSpoof database

Similarly, the PartialSpoof database comprises three subsets, each containing both utterance-level and frame-level labels. This dataset is constructed from the ASVspoof 2019 logical access (LA) database, which includes fake utterances generated by text-to-speech (TTS) systems and voice conversion (VC) systems [42]. The construction of the PartialSpoof database involves several steps to ensure fine-grained data generation, aiming to avoid potential artifacts introduced from concatenating audio segments [40]. This meticulous approach renders this database more challenging in terms of both spoofing detection and the localization of fake regions, as compared to the two databases previously mentioned. As illustrated in Table 1, the audio lengths of utterances in the training, development, and evaluation sets are within a similar range, ranging from approximately 0.5 seconds to 20 seconds. Notably, the PartialSpoof-Eval set consists of spoofed utterances generated by different TTS and VC systems from the PartialSpoof-Dev set.

4. Methods

We employ a frame-level boundary detection system and a frame-level anti-spoofing system to detect partially spoofed utterances and identify the manipulated region at the same time. Both systems utilize the same network architecture as our previous model developed for ADD 2022 Track 2 [27], differing mainly in their output labeling strategy. The architecture and data flow are visually depicted in Figure 2.

The frame-level detection models used in our approach, one for boundary detection and the other for anti-spoofing detection, utilize the deep framework depicted in Figure 2. We employ large-scale self-supervised pre-training models such as Wav2Vec2 [43] and WavLM [44] to extract frame-level acoustic representations from raw audio signals. Following that, a 1-dimensional residual network module (ResNet-1D) is utilized to further extract frame-level features specific to our task. A backend classifier is then applied to predict the classification result for each frame. Specifically, the ResNet-1D module consists of two 1-dimensional convolutional neural network (1D-CNN) layers surrounding a series of residual blocks. Each residual block contains two 1D-CNN layers and incorporates a residual connection from input to output. The backend classifier incorporates transformer encoders and a Bidirectional LSTM (BLSTM) to capture long-range global contexts. It also includes a fully connected layer that maps the high-dimensional output vectors to binary outputs. The details of our proposed model are presented in Table 2.

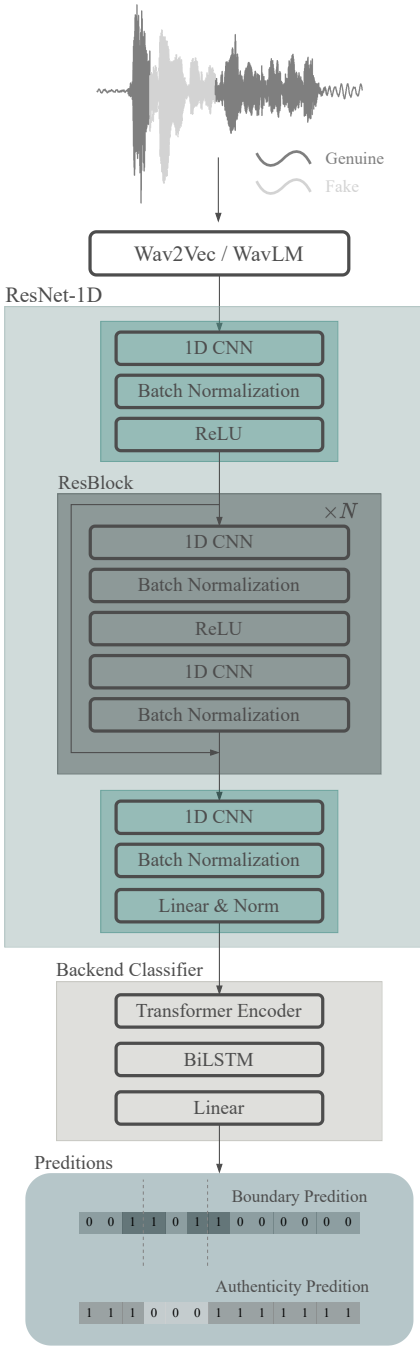

Figure 2: The overall architecture of frame-level detection models

Table 2: The network configuration of frame-level detection models, where **C**(kernal size, padding, stride) denotes the convolutional layer, [·] denotes the residual block, **E**(number of layers, number of heads, FFN size) denotes the Transformer Encoder, BiLSTM(number of layers, hidden units) denotes BiLSTM layer, Linear(input size, output size) denotes the fully-connected layer; L relates to the duration of the input audio signal and T is the number of label frames

Layer	**Output Size**	**Structure**	**#Parameters**	**Note**
Input audio	$L \times 1$	-	-	l seconds
Wav2Vec2/WavLM	$T \times 1024$	-	refer to Table 3	-
1D-CNN	$T \times 512$	$\mathbf{C}(5, 2, 1)$w/o bias	2.62M	-
ResBlock(s)	$T \times 512$	$\begin{bmatrix}\mathbf{C}(1, 0, 1) \text{ w/o bias} \\ \mathbf{C}(1, 0, 1) \text{ w/o bias}\end{bmatrix}$	6.32M	12 blocks
1D-CNN	$T \times 128$	$\mathbf{C}(1, 0, 1)$	0.066M	-
Linear & Norm	$T \times 128$	Linear(128, 128)	0.017M	-
Transformer Encoder	$T \times 128$	$\mathbf{E}(2, 4, 1024)$	0.66M	dropout=0.5
BiLSTM	$T \times 128$	BiLSTM(1, 128)	0.26M	dropout=0
Linear (Boundary) For (Anti-spoofing)	$T \times 1$ $T \times 2$	Linear(256, 1) Linear(256, 2)	- -	Binary Cross Entropy Loss Cross Entropy Loss

Examples of the labeling process are also illustrated in Figure 2. The goal of the boundary detection model is to identify discontinuities in a sequence of frames. To achieve this, we adopt a labeling strategy that assigns a value of '1' to certain frames surrounding waveform concatenation boundaries, while frames within genuine and fake segments are labeled with '0'. On the other hand, the anti-spoofing detection model assigns a label of '0' to fake audio frames and '1' to genuine frames in the target output.

4.1. Large-scale self-supervised pre-training models

Large-scale self-supervised pre-training models have demonstrated significant advantages in extracting robust acoustic features for various speech-related tasks, including speech recognition [45], speaker recognition [46, 47] and spoofing detection [48, 49]. Recent studies have indicated that employing pre-training models with larger parameter sizes generally improves system performance [27, 39, 40]. Therefore, we leverage multiple pre-training models for feature extraction in our approach and comprehensively analyze their performance. The large-scale self-supervised pre-training models used in our study are listed in Table 3. All Wav2Vec2-based and WavLM-based models operate at a frame rate of 20ms.

The Wav2Vec2-Base model, previously utilized in our work, possesses a relatively smaller parameter size and generates output features with a dimension of 768. On the other hand, all other models have approximately 317M parameters and produce output features of size 1024. These models are pre-trained using 16kHz audio files, which aligns with the sample rate of the database utilized in our study. The Wav2Vec2-XLSR-CN model is fine-tuned from Wav2Vec2-XLSR

[2]https://huggingface.co/facebook/wav2vec2-base-960h
[3]https://huggingface.co/facebook/wav2vec2-large-xlsr-53
[4]https://github.com/microsoft/unilm/tree/master/wavlm

Table 3: Large-scale self-supervised pre-training models used in our approach

Model Name	h_{train}	$h_{\text{fine-tune}}$	n_{params}	d_{model}
Wav2Vec2-Base[2] [43]	960	-	94.37M	768
Wav2Vec2-XLSR[3] [50]	50k	-	317.38M	1024
Wav2Vec2-XLSR-CN [51]	50k	215.27	317.38M	1024
Wav2Vec2-XLSR-EN [52]	50k	1,087	317.38M	1024
WavLM-Large[4] [44]	94k	-	316.62M	1024

on around 215.27 hours of Chinese data, using the training and validation splits of Common Voice 6.1 [53], CSS10 [54], and ST-CMDS [55] datasets. Similarly, the Wav2Vec2-XLSR-EN model is fine-tuned from Wav2Vec2-XLSR on 1,087 hours of English data, using the training and validation splits of the Common Voice 6.1 dataset.

4.2. Training

We employ several strategies to train our proposed models, including data loading and sampling processes. During training, Our proposed frame-level models take raw audio clips randomly cut from audio utterances. The input length of the model is fixed and set to l samples of the raw audio signal. Additionally, each database has its unique data distribution and generation process. Therefore, the labeling process and data balancing are crucial to model training.

4.2.1. ADD2022-T2

For the training process on the ADD2022-T2 database, we adhere to the data sampling procedure outlined in our previous work [27]. This approach introduces a sampling strategy that selects data and cuts to fixed length l from utterances with and without boundaries, following a probability distribution of {without boundary: 0.3, with boundaries: 0.7}. This strategy is employed to mitigate the issue of data imbalance during training. The statistics of the ADD2022-T2 database show that the number of bona fide utterances (those without boundaries) is significantly less than the number of partially fake utterances. Therefore, including a limited sampling probability for utterances without boundaries is essential, as the model fails to converge otherwise.

4.2.2. ADD2023-T2

The issue of data imbalance also arises in ADD2023-T2. Specifically, there are 1,185 fake utterances compared to 26,554 bona fide utterances. Furthermore, partially manipulated utterances contain a significant number of authentic audio frames, while the count of fake frames is comparatively lower. To address this imbalance, we employ two strategies during the data loading process to ensure equilibrium between bona fide frames and fake frames. This balance is achieved through the following probability distribution: {Strategy I.: 0.3, Strategy II.: 0.7}.

I. Randomly choose genuine and fake utterances from the training set, with a probability of 0.3 for selecting genuine utterances and a probability of 0.7 for selecting fake utterances. Then cut to fixed length l as input for training.

II. Randomly segment clips from the partially spoofed utterances in the training set.

In the case of boundary detection models, genuine and fake utterances are treated as identical because they don't contain any boundaries within them. Therefore, when use Strategy I., there's no need to differentiate and sample between genuine and fake utterances. Similar to what is used for ADD2022-T2, the distribution used for data sampling in boundary detection models is as follows: {Strategy I. (without boundary): 0.3, Strategy II. (with boundaries): 0.7}. Regarding label generation, we assign labels of '1' to the four closest frames surrounding each boundary.

4.2.3. PartialSpoof

The data construction process of the PartialSpoof database differs from that of ADD2022-T2 and ADD2023-T2. It involves fine-grained editing techniques such as normalization and computation of the best fusion point between genuine and synthesized clips. The splicing of partially spoofed audio occurs in non-speech segments, posing a greater challenge for detection. The insertion is achieved through overlap-add between the non-speech portions of genuine and synthesized audio. When utilizing the PartialSpoof database for training an anti-spoofing system, we assign a label of '0' to all frames associated with fake and edited segments, including the overlap-add regions, while genuine frames are labeled '1'. Regarding training the boundary detection system, we consider the middle frame of the overlap-add sections as the boundaries and set labels '1' to the four closest frames surrounding each boundary. The sampling strategy employed for this database is identical to the one used for ADD2022-T2.

4.3. Inference

The inference process for our frame-level spoofing detection systems is illustrated in Figure 3. Similarly, the inference process for boundary detection models follows a similar procedure, with the distinction that the outputs correspond to boundary probabilities. During the inference stage, we begin by dividing the audio signal into overlapping audio clips. Each clip has the same length as the training samples, which is l, with a step size of $l/2$. After obtaining the authenticity probabilities for each frame within the audio clips, we merge the results of all the clips by averaging the overlapping regions.

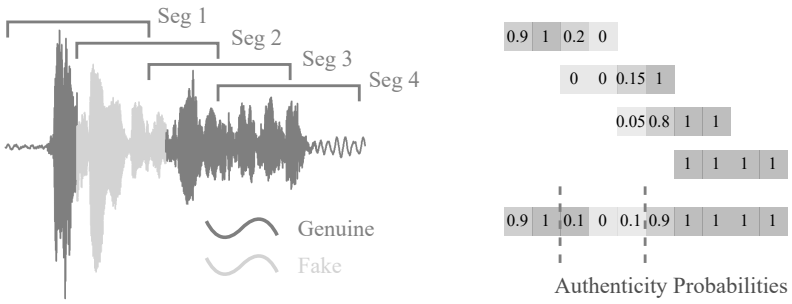

Figure 3: The inference process for our proposed frame-level spoofing detection system

4.4. Model integration

We conducted investigations with several model integration strategies to explore the practical application of our frame-level detection models. The boundary detection system demonstrates

proficiency in detecting certain artifacts in partially spoofed utterances and segmenting them into audio clips. However, it lacks the ability to verify the authenticity of each segment. On the other hand, frame-level anti-spoofing systems do not excel in utterance-level spoofing detection.

Concerning model integration, we can apply various strategies to locate fake regions. One approach is to consider the boundary detection model as a model for utterance-level partially spoofed audio detection. This means that if a boundary is detected, the corresponding utterance is regarded as a partially spoofed utterance. Then, after the given utterance is classified as a partially fake utterance, we can locate fake regions using the following strategies:

I. Directly use the anti-spoofing model for the whole utterance to obtain frame-level classification results regarding authenticity.

II. After segmenting the utterance into audio clips according to the detected boundaries, use utterance-level or segment-level anti-spoofing models to validate the authenticity of each audio segment.

Our frame-level anti-spoofing detection model can also be applied to the second strategy. Specifically, we perform segment-level spoofing detection by employing majority voting. If most frames within an audio segment are considered genuine, then the segment is classified as a bona fide audio clip; otherwise, it is regarded as a fake clip. Mathematically, for an audio clip with n audio frames, and the corresponding binary prediction is $y = [y_1, y_2, \ldots, y_n]$, where $y_i \in \{0, 1\}$, we consider the audio clip as genuine when $sum(y) > n/2$, and regard it as fake otherwise. In our study, for comparison purposes, we also incorporated the state-of-the-art segment-level anti-spoofing system known as Audio Anti-Spoofing using Integrated Spectro-Temporal (AASIST) system[5] [56] for the second strategy in our evaluation. The original AASIST model utilizes a RawNet2-based encoder [57] for extracting high-level acoustic representations. Given that our proposed model employs large-scale pre-training models for acoustic feature extraction, we also incorporated pre-training models into AASIST, following the approach outlined by Tak et al. [58], to ensure a fair comparison.

Since the label for ADD2023-T2-Test remains inaccessible, we can only access the overall performance through Codalab. In the context of the ADD 2023 Track 2 challenge, we engaged with algorithm 1 presented below, which demonstrated the most exceptional performance in the competition. The algorithm is implemented for each utterance in the ADD2023-T2-Test. Initially, we extract segmented audio clips using the boundaries identified by the boundary detection model. Some of the detection scenarios are shown in Figure 4. If the number of segments is 1, indicating that no boundary is detected and the utterance is either bona fide or fake without any inserted audio clips, we determine the authenticity of the utterance by assessing the proportion of frames classified as fake, utilizing a spoofing detection model. The threshold ratio is set to 0.4. Consequently, if fewer than 40% of frames are classified as fake, the utterance is categorized as bona fide; otherwise, it is labeled as fake.

In cases where the number of segments is 2, we apply the following criteria to classify a segment as fake: 1. The proportion of predicted fake frames within the segment exceeds that of the other segment; 2. This proportion is greater than the threshold ratio of 0.4. Alternatively, if these conditions are not met, we designate the segment with the shorter length as fake based on insights gained from the training set of ADD 2023 Track 2. For utterances with 3 segments, we

[5]https://github.com/clovaai/aasist

Algorithm 1 Scoring algorithm for ADD 2023 Track 2

Require: segmented audio clips from the boundary detection model

FakeProportionRatio = 0.4 ▷ The Proportion of fake frames predicted by spoofing detection model

if #Segments = 1 **then**

  if $\frac{\text{\#Fake frames}}{\text{\#Frames}} < \text{FakeProportionRatio}$ **then**

    Classify as a bona fide segment

  else

    Classify as a fake segment

  end if

else if #Segments = 2 **then**

  if $\frac{\text{\#Fake frames}_{seg1}}{\text{\#Frames}_{seg1}} > \text{FakeProportionRatio}$ and $\frac{\text{\#Fake frames}_{seg1}}{\text{\#Frames}_{seg1}} > \frac{\text{\#Fake frames}_{seg2}}{\text{\#Frames}_{seg2}}$ **then**

    Classify segment_2 as a bona fide segment and segment_1 as a fake segment

  else if $\frac{\text{\#Fake frames}_{seg2}}{\text{\#Frames}_{seg2}} > \text{FakeProportionRatio}$ and $\frac{\text{\#Fake frames}_{seg2}}{\text{\#Frames}_{seg2}} > \frac{\text{\#Fake frames}_{seg1}}{\text{\#Frames}_{seg1}}$ **then**

    Classify segment_1 as a bona fide segment and segment_2 as a fake segment

  else

    Assign the segment with shorter length as fake and the other as bona fide

      ▷ Prior knowledge from the training set, most fake clips are with shorter length

  end if

else if #Segments = 3 **then**

  Classify the middle segment as fake and the other two as bona fide segments

else

  for each audio segment **do**

    if $\frac{\text{\#Fake frames}}{\text{\#Frames}} < \text{FakeProportionRatio}$ **then**

      Classify as a bona fide segment

    else

      Classify as a fake segment

    end if

  end for

end if

classify the middle segment as fake and the other two as bona fide segments, again relying on knowledge derived from the training set. Lastly, for utterances with more than 3 segments, we determine their predicted authenticity based on the proportion of predicted fake frames.

The frame-level anti-spoofing model we proposed exhibits limited robustness when dealing with cross-domain data. We believe that ADD2023-T2-Test contains a substantial amount of out-of-domain data from previously unseen scenarios, especially when compared to ADD2023-T2-Dev. In light of this, we have integrated a robust Variational Autoencoder (VAE) model into the scoring process to enhance the system's performance in the competition. The VAE model is a probabilistic graphical model that effectively reduces dimensionality in a statistically grounded manner. Unlike other techniques like autoencoders and principal component analysis (PCA), the VAE offers reconstructed probability as a measure of deviation rather than relying on reconstruction error as an anomaly score [59]. The reconstruction probability is commonly employed as the final score for deviation-based outlier detection [60]. Notably, the VAE model, functioning as an outlier detection model, only requires bona fide samples for its training. Wang et al. demonstrat-

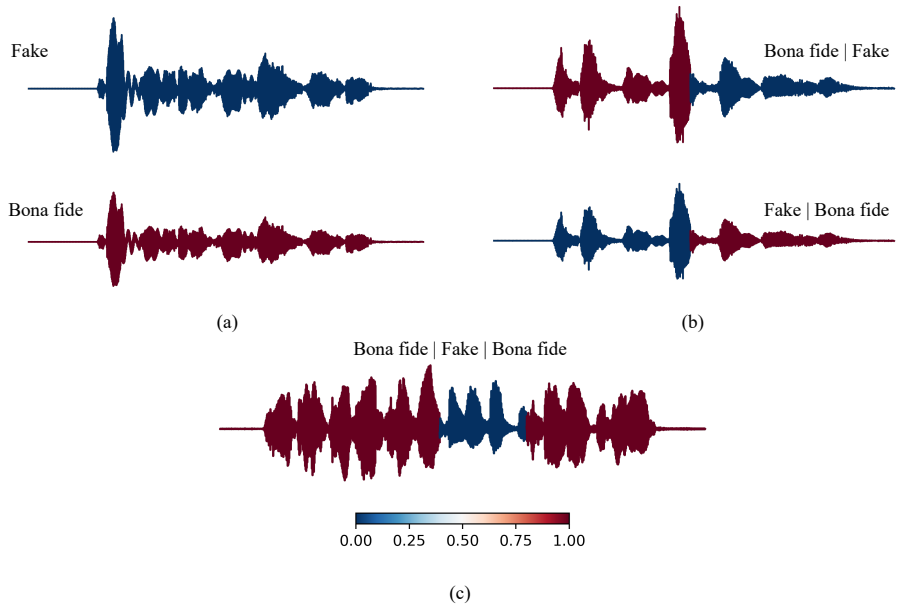

Figure 4: Three scenarios encountered during ADD2023-T2-Test scoring: (a) No boundary detected, (b) One boundary detected, (c) Two boundaries detected

ed the applicability of this model in spoofing detection using real speech differential features as input [15]. In line with their work, we utilized the open-source module "pyod"[6] to train a VAE model with bona fide utterances from ADD2023-T2-Train. It's worth noting that the VAE model is trained with WavLM-based acoustic features.

The VAE model provides scoring at the utterance level, contributing solely to the enhancement of sentence accuracy in the ADD2023-T2-Test evaluation. In this model, when there is a larger deviation during inference, the input utterances are more likely to be classified as fake. Consequently, we utilize the VAE model to finalize the authenticity of utterances falling into two categories: those with either 0 boundaries (either bona fide or fake) and those with more than ten boundaries (considered outliers) as detected by the boundary detection system. Within these categories, we identify and label utterances as fake based on the rescoring of samples within the top 45% (Figure 5), which exhibit the greatest deviation from the genuine training samples.

5. Experimental Results

5.1. Experimental setup

For all frame-level detection models, l is set to 1.28 seconds, according to the findings from our previous work [27]. In this case, the input size L in Table 2 corresponds to 20,480 samples. The number of frames T is set to 64, considering that Wav2Vec2-based and WavLM-based models operate at a frame rate of 20 ms. Furthermore, we incorporate online data augmentation using

[6]https://github.com/yzhao062/pyod

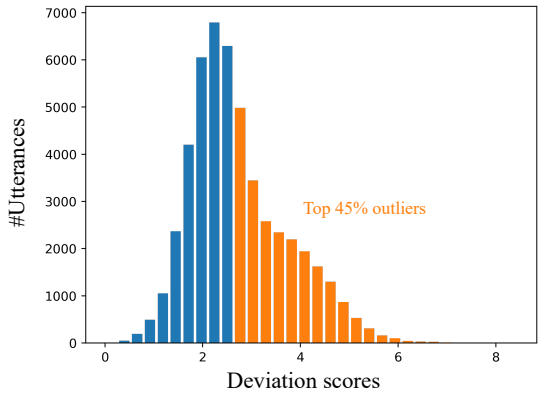

Figure 5: Illustrative deviation score distribution

the MUSAN [61] and RIRs [62] corpora. All models are trained using binary cross-entropy loss and the Adam optimizer for a total of 200 epochs. The training process is carried out utilizing eight 2080-Ti GPUs with a mini-batch size 16. The initial learning rate is 10^{-4}, and we utilize the Noam scheduler [63] with 1600 warm-up steps.

During the training process, the model's performance measured by the relevant evaluation metric is evaluated and recorded for each epoch. For training on the ADD2022-T2-Train dataset, the performance of the models is evaluated using the ADD2022-T2-Adapt dataset. The utilization of a smaller dataset can expedite the evaluation procedure and performance tracking during the training phase. This rationale guides the approach taken when training on ADD2023-T2. Here, a subset of the ADD2023-T2-Dev dataset, comprising 1500 bona fide utterances, 430 fake utterances, and 1,570 partially spoofed utterances, is randomly selected to monitor performance. Likewise, in the case of the PartialSpoof database, the performance evaluation set utilized during training consists of all bona fide utterances and 2,548 partially spoofed utterances from the PartialSpoof-Dev dataset.

The boundary detection models are evaluated based on the Equal Error Rate (EER), while the anti-spoofing systems' evaluation employs the F_1 score. Following 200 epochs of training, for the boundary detection model, the average of the top five models with the lowest EER is selected for inference and evaluation. The corresponding EERs are determined using the confidence score at the utterance level, derived from the mean of the four largest probabilities. This approach is taken as we assign the surrounding four frames for each boundary during the labeling process in training. Regarding spoofing detection models, the top five models achieving the highest F_1 scores are averaged and employed for inference on evaluation datasets.

We conduct training of various models using different databases, systems, and features extracted by large-scale pre-training models. The obtained results are presented in this section. Throughout the following content, the boundary detection system is referred to as **BDR**, and the anti-spoofing system as **SPF**. Moreover, besides analyzing the individual performance of single models, we also investigate the effectiveness of model integration for different evaluation datasets, aiming to find the best approach for detecting and locating manipulated audio clips.

5.2. Evaluation metrics

In our study, we employ two primary evaluation metrics, namely the Equal Error Rate (EER) and the segment F_1 score. The choice of these metrics is based on their common usage in the literature, with the EER serving as the evaluation metric for Track 2 of ADD2022, and the F_1 score being utilized in ADD2023. We evaluate the boundary detection models based on the utterance-level EER, whereas the performance of the anti-spoofing detection systems is assessed using the segment F_1 score. The EER is a threshold-free metric calculated when the false alarm rate $P_{fa}(\theta)$ is equal to the miss rate $P_{miss}(\theta)$ at a specific threshold θ. $P_{fa}(\theta)$ and $P_{miss}(\theta)$ are defined as shown in Equation 1 and Equation 2, respectively.

$$P_{fa}(\theta) = \frac{\#\ fake\ utterances\ with\ score > \theta}{\#\ total\ spoofing\ utterances} \tag{1}$$

$$P_{miss}(\theta) = \frac{\#\ bona\ fide\ utterances\ with\ score \leq \theta}{\#\ total\ bona\ fide\ utterances} \tag{2}$$

Segment F_1 score is employed to assess the performance in locating manipulated regions, as formatted in Equation 3. Specifically, TP represents true positive (the count of bona fide audio frames correctly predicted as bona fide), FP represents false positive (the count of fake frames incorrectly classified as bona fide), and FN represents false negative (the count of bona fide frames incorrectly predicted as fake).

$$F_1 = \frac{2 * TP}{2 * TP + FN + FP} \tag{3}$$

The performance evaluation for ADD2023-T2-Test is only accessible through CodaLab[7]. Regarding our approach's performance on the ADD2023-T2-Test dataset in detecting partially spoofed audio and locating manipulated regions, we report the ADD score obtained from the challenge's official leaderboard. The ADD score combines multiple metrics to calculate the overall performance. These metrics encompass sentence accuracy (A) and segment F_1 score, with the sentence accuracy formulated in Equation 4. The sentence accuracy metric measures the model's effectiveness in correctly classifying genuine and manipulated audio, while the segment F_1 score evaluates the model's ability to identify fake regions. The official ADD score (S_{ADD}) is computed by a weighted sum of the sentence-level accuracy and the segment F_1 score, as illustrated in Equation 5.

$$A = \frac{\#\ utterances\ with\ correct\ classification}{\#\ total\ utterances} \tag{4}$$

$$S_{ADD} = 0.3 \times A + 0.7 \times F_1 \tag{5}$$

5.3. Single models

The performance of boundary detection models trained with the ADD2022-T2 database is shown in Table 4. Our previous model, which utilized Wav2Vec2-Base as the front-end feature

[7] https://codalab.lisn.upsaclay.fr/competitions/11361

extractor, performs well on ADD2022-T2-Adapt and ADD2022-T2-Test. However, its performance on the cross-domain dataset PartialSpoof-Eval is unsatisfactory, with an EER of 34.11%. Models based on larger Wav2Vec2 pre-training models show improvement across all three evaluation datasets. Notably, the boundary detection model based on Wav2Vec2-XLSR-CN, which is fine-tuned on Chinese data, demonstrates improvements across all evaluation datasets in comparison to the model based on Wav2Vec2-XLSR. This suggests that using front-end features extracted by a pre-training model trained on Chinese data can benefit the system's performance. On the other hand, although the model based on the WavLM feature has a similar parameter size to the Wav2Vec2-based models, it exhibits more robust performance. The WavLM-based model achieves an EER of 4.40% on ADD2022-T2-Test, significantly outperforming the best model based on Wav2Vec2 features. This outstanding performance establishes our system as the new state-of-the-art on the ADD2022-T2-Test set, surpassing the previous leading performance of 4.80% EER achieved by an utterance-level partially fake audio detection model that relies on a larger pre-training model with 1 billion parameters [39]. Furthermore, the WavLM-based model achieves an EER of 13.25% on the cross-domain dataset PartialSpoof-Eval, which is significantly better than those based on Wav2Vec2 features.

Table 4: The experimental performance of boundary detection (BDR) systems trained with the ADD2022-T2 database, '-' denotes unavailable result

Database	**System**	**Feat.**	**Performance on evaluation sets** EER ↓ (%)		
			ADD2022-T2-Adapt	ADD2022-T2-Test	PartialSpoof-Eval
ADD2022	BDR	Wav2Vec2-Base [27]	3.71	6.64	34.11
		Wav2Vec2-XLSR	3.14	6.45	30.86
		Wav2Vec2-XLSR-CN	3.04	5.47	27.59
		WavLM	**2.85**	**4.40**	**13.25**
	[39]	Wav2Vec2-XLSR-1B	3.33†	4.80	-
	[41]	LFCC, MFCC, MSTFT	-	7.90	-
	[26]	Wav2Vec2-XLSR-128 [45]	-	16.59	-

† The number of bona fide utterances used for evaluation might be different

Table 5 presents the performance of systems trained on the ADD2023-T2 database, evaluated on two datasets: ADD2023-T2-Dev, and the cross-domain dataset PartialSpoof-Eval. Among the boundary detection systems, all models demonstrate impressive performance on ADD2023-T2-Dev, with the WavLM-based model achieving the best result, obtaining an EER of 0.064. Concerning the performance on the cross-domain dataset PartialSpoof-Eval, the Wav2Vec2-based boundary detection systems trained on ADD2023-T2 outperform those trained on the ADD2022-T2 database. However, the WavLM-based model trained on ADD2023-T2 achieves a higher EER than the one trained on ADD2022-T2.

Regarding frame-level anti-spoofing systems, all models achieve remarkably high F_1 scores on the ADD2023-T2-Dev set, with scores higher than 99.9%. The performance on the cross-domain dataset PartialSpoof-Eval is also provided in the table. Here, the WavLM-based model demonstrates the best performance with an EER of 87.01%.

The performance of systems trained on the PartialSpoof database exhibits similar trends regarding different pre-training models, and the results are presented in Table 6. The WavLM-based models consistently outperform the Wav2Vec2-based models on most evaluation dataset-

Table 5: Performance of frame-level systems trained on ADD2023-T2 database regarding acoustic features from different pre-training models, where SenAcc denotes the sentence accuracy

Database	System	Feat.	Performance on evaluation sets	
			ADD2023-T2-Dev EER ↓ (%)	PartialSpoof-Eval EER ↓ (%)
	BDR	Wav2Vec2-XLSR	0.289	25.20
		Wav2Vec2-XLSR-CN	0.235	23.98
ADD2023		WavLM	**0.064**	**15.27**
			F_1 ↑ (%)	F_1 ↑ (%)
	SPF	Wav2Vec2-XLSR	99.914	86.39
		Wav2Vec2-XLSR-CN	99.918	86.72
		WavLM	**99.925**	**87.01**

s. Specifically, the WavLM-based boundary detection model achieves an EER of 1.16% on PartialSpoof-Dev and an EER of 1.74% on PartialSpoof-Eval, presenting a significant advantage over the models trained with Wav2Vec2 features.

Table 6: Performance of frame-level systems trained on PartialSpoof database regarding acoustic features from different pre-training models

Database	System	Feat.	Performance on evaluation sets		
			PartialSpoof-Dev EER ↓ (%)	PartialSpoof-Eval EER ↓ (%)	ADD2023-T2-Dev EER ↓ (%)
	BDR	Wav2Vec2-XLSR	3.49	3.94	49.86
		Wav2Vec2-XLSR-EN	2.91	3.64	49.75
PartialSpoof		WavLM	**1.16**	**1.74**	**45.34**
			F_1 ↑ (%)	F_1 ↑ (%)	F_1 ↑ (%)
	SPF	Wav2Vec2-XLSR	**96.20**	91.48	91.55
		Wav2Vec2-XLSR-EN	96.02	91.44	92.45
		WavLM	95.95	**92.96**	**94.06**

However, when evaluating the performance on the cross-domain dataset ADD2023-T2-Dev, all boundary detection systems fail to make correct predictions, as the EERs are above 40%. In contrast, the boundary system trained with the ADD2023-T2 dataset can achieve an EER of 15.27% in the PartialSpoof-Eval set. This suggests that models trained with utterances generated by coarse concatenation are robust for utterances with fine-grained concatenation, while the opposite may not hold true.

In terms of anti-spoofing systems, all models demonstrate similar performance. The F_1 scores on PartialSpoof-Dev are approximately 96%, while on PartialSpoof-Eval, the F_1 scores are around 92%. On the cross-domain dataset ADD2023-T2-Test, the Wav2Vec2-XLSR-based SPF model achieves an F_1 score of 91.55%, and the Wav2Vec2-XLSR-EN-based SPF model achieves 92.45%. However, the WavLM-based model outperforms both systems with an F_1 score of 94.06%, further showcasing the robustness of WavLM features on cross-domain datasets. It is worth noting that SPF systems trained with Wav2Vec2-XLSR-EN feature perform similarly to those trained with Wav2Vec-XLSR. This could be partly attributed to English utterances constituting a substantial portion of the multilingual corpus used to pre-train Wav2Vec-XLSR,

accounting for over 40% of the dataset [53]. Consequently, fine-tuning the Wav2Vec2-XLSR-EN model on the English parts does not yield significant improvements.

Additionally, we conducted testing using edited samples generated by a recently proposed synthesis model called Voicebox [64]. 12 audio samples, comprising 6 original utterances and their corresponding 6 edited versions, are available online[8]. Note that the unseen editing is performed at the spectrogram level, and subsequently converted into seamless waveforms without any discernible artifacts. It's worth mentioning that our model, trained with ADD2023-T2, which typically detects editing boundaries at the waveform level, exhibited a noteworthy robustness in identifying editing within these utterances, even in this spectrogram-level editing scenario. The boundary detection results obtained by the three BDR systems are presented in Figure 6. This figure illustrates the probability scores, indicating the likelihood that a given utterance has been manipulated, where a higher score suggests a greater likelihood of manipulation. In particular, the BDR system trained with ADD2023-T2 demonstrated a strong capability in detecting editing boundaries within the manipulated utterances generated by Voicebox. This detection is evident in the audio samples labeled with ID 0, 1, 3, and 4. One of the detection results is shown in Figure 7, where the boundaries of the edited content are correctly recognized.

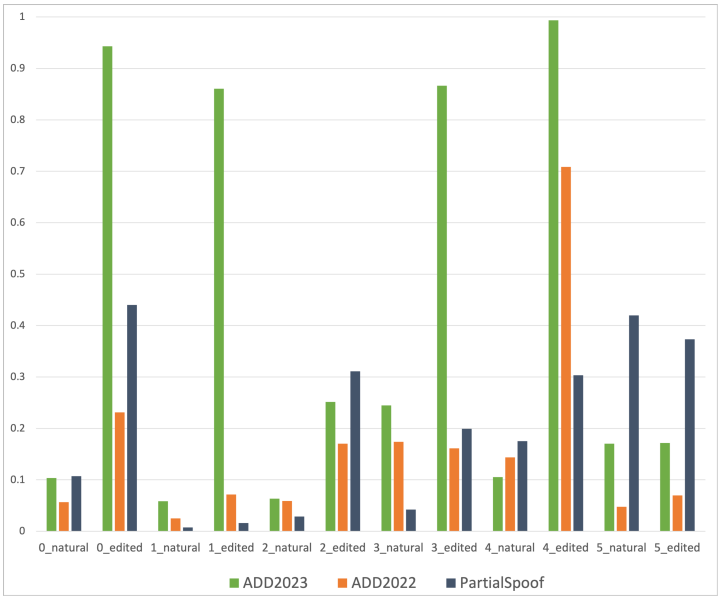

Figure 6: Boundary detection scores on utterances manipulated by Voicebox

5.4. Model integration

Figure 8 depicts the ground truth label and predictions generated by WavLM-based frame-level detection models for an utterance from the PartialSpoof-Eval dataset. The ground truth labels indicate that the utterance is constructed by merging four audio segments, comprising two fake and two bona fide segments. The anti-spoofing prediction obtained from the spoofing

[8]https://voicebox.metademolab.com/edit.html

Figure 7: Boundary detection result of an edited utterance from VoiceBox

detection system demonstrates accurate predictions for most frames except for the last segment. As for the predictions from the boundary detection system, the boundaries are well predicted and match those from the ground truth.

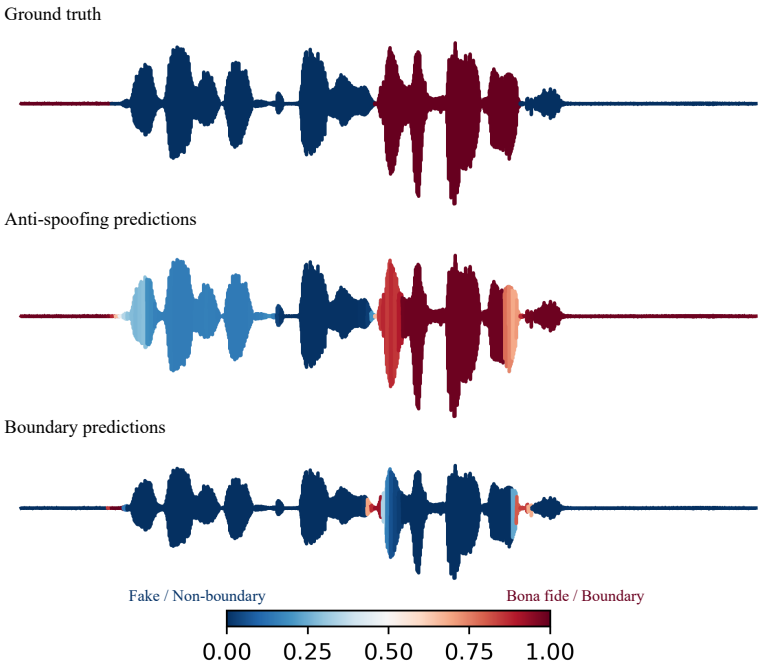

Figure 8: A randomly selected sample from the PartialSpoof-Eval dataset. The predictions are obtained from frame-level WavLM-based models

As illustrated in Figure 8, we can segment the utterance into four audio clips based on the boundary detection prediction results. However, the BDR output does not provide information about the authenticity of each segment. Therefore, we can subsequently adopt the anti-spoofing results to classify each segmented region with the strategies mentioned in Section 4.4.

Table 7 demonstrates the limitations of the frame-level spoofing system in utterance-level partially spoofed audio detection. Specifically, we calculate the utterance-level scores by taking the minimum value of the predicted probabilities of all frames in an utterance and obtain

the EERs for the anti-spoofing system. As shown in Table 7, the EERs from the SPF systems are notably higher than those obtained from the BDR systems, indicating the weaknesses of the SPF systems in this context of utterance level prediction. Consequently, to optimize the practical application of these two systems, we can use the BDR system to identify partially spoofed utterances and subsequently employ the SPF system to locate fake regions.

Table 7: Utterance-level partially spoofed detection performance of systems trained with PartialSpoof database

Training Set	**Evaluation Set**	**Performance regarding Feats. and models** EER of BDR / EER of SPF ↓ (%)		
		Wav2Vec2-XLSR	Wav2Vec2-XLSR-EN	WavLM
PartialSpoof-Train	PartialSpoof-Dev	3.49 / 7.99	2.91 / 9.54	1.16 / 9.22
	PartialSpoof-Eval	3.94 / 28.77	3.64 / 28.30	1.74 / 24.56

This section examines the model integration performance based on the strategies proposed in Section 4.4. We conduct the evaluation using WavLM-based models, as they demonstrate the best performance in the single model evaluation.

5.4.1. PartialSpoof database

In addition to the WavLM-based SPF system, we incorporate three models with the state-of-the-art framework AASIST as baselines for spoofing detection. As the PartialSpoof database is derived from ASVspoof2019, we utilize a pre-trained AASIST model trained on the LA dataset of ASVspoof2019 for spoofing detection (the model is available on the AASIST project website). Furthermore, we train another AASIST model using audio clips segmented based on the ground truth labels from PartialSpoof-Train. The third AASIST model, referred to as WavLM-AASIST, utilizes the same WavLM model for acoustic feature extraction and is trained on segmented audio clips from PartialSpoof-Train. The results of the integration performance on the PartialSpoof database are presented in Table 8.

Table 8: Frame-level performance of integration results from systems trained with PartialSpoof database

Waveform Segmentation	**Spoofing System**	**Performance on evaluation sets** $F_1 \uparrow$ (%)	
		PartialSpoof-Dev	PartialSpoof-Eval
WavLM predicted boundaries	AASIST (ASVspoof2019)	64.90	61.26
	AASIST (PartialSpoof)	92.29	88.00
	WavLM-AASIST (PartialSpoof)	94.89	90.29
	WavLM SPF	95.71	92.90
Ground truth boundaries	AASIST (ASVspoof2019)	66.35	62.76
	AASIST (PartialSpoof)	95.68	89.90
	WavLM-AASIST (PartialSpoof)	96.74	91.18
	WavLM SPF	96.69	93.14
w/o segmentation	WavLM SPF	95.95	92.96

As shown in the table, when incorporating the boundary information from the BDR model,

the WavLM-based SPF model's spoofing performance surpasses the AASIST's. It's important to note that the WavLM-based SPF model conducts segment-level spoofing detection based on the majority class of frames within each segment. Specifically, on the PartialSpoof-Eval dataset, the WavLM-based SPF model achieves an impressive F_1 score of 92.9%. However, the fake region detection performance of the AASIST model trained on ASVspoof2019 is unsatisfactory. This model achieves an F_1 score of 61.26%, primarily due to domain mismatch since the utterances from ASVspoof2019 undergo normalization in the generation process of PartialSpoof. Conversely, the AASIST system trained on PartialSpoof segments achieves a score of 88%. Moreover, the spoofing perfomrance has improved with AASIST system trained on WavLM-based features, resulting with an F_1 score of 90.29%. Similar results can be observed from the PartialSpoof-Dev dataset.

Additionally, we can assess the segment-level spoofing performance by evaluating with ground truth boundaries. On the PartialSpoof-Dev set, both the WavLM-based AASIST and WavLM-based SPF models perform well, achieving an F_1 score close to 97%. However, for the out-of-domain set, PartialSpoof-Eval, our proposed SPF model demonstrates greater robustness, surpassing the performance of the WavLM-based AASIST model by approximately 2%.

Without ground truth boundaries, the best performance is achieved when using the frame-level WavLM-based SPF model directly. These performance gaps show the effectiveness of frame-level spoofing detection model compared to the AASIST baselines in the task of locating fake regions.

5.4.2. ADD2023-T2 database

We conduct a similar experiment using the ADD2023-T2 database, where the spoofing detection model AASIST is trained with the ADD2023-T2-Train dataset. The results of this experiment are presented in Table 9. Notably, all model integration pairs perform exceptionally well on the ADD2023-T2-Dev dataset, with an F_1 score close to 100%. We observe that the WavLM-based SPF model also outperforms the AASIST baselines on the ADD2023-T2-Dev dataset. In addition, in scenarios where ground truth boundaries are available, conducting segment-level spoofing detection with the WavLM-based SPF system yields an impressive F_1 score of 100%.

Table 9: Performance of integration results from systems trained with ADD2023-T2 database

Waveform Segmentation	**Spoofing System**	**Performance on evaluation sets** ADD2023-T2-Dev $F_1 \uparrow$ (%)
WavLM predicted boundaries	AASIST (ADD2023-T2)	99.89
	WavLM-AASIST (ADD2023-T2)	99.89
	WavLM SPF	99.925
Ground truth boundaries	AASIST (ADD2023-T2)	99.923
	WavLM-AASIST (ADD2023-T2)	99.92
	WavLM SPF	100
w/o segmentation	WavLM SPF	99.925

The performance of our proposed models in Track 2 of the ADD 2023 challenge is outlined in Table 10. We employ Algorithm 1 to derive spoofing decisions based on regions segmented by the SPF system, resulting in an ADD score (S_{ADD}) of 0.6538, which is about 7.4% relatively

higher than team C02. Then we apply the VAE model to rescore utterance-level spoofing decisions. After applying the VAE scoring strategy presented in Section 4.4, our final fusion system achieves an ADD score of 0.6713. This performance positions our system at the top among 16 teams in the ADD 2023 challenge, outperforming the baseline system S04 [28] by a substantial margin.

Table 10: ADD 2023 Track 2 Rankings on the test set evaluated by ADD score S_{ADD} [28]

#	System / ID	S_{ADD}	#	ID	S_{ADD}	#	ID	S_{ADD}
1	Our system	**0.6713**	7	C07	0.5399	13	**S04**	0.4225
2	C02	0.6249	8	C08	0.5086	14	C13	0.4211
3	C03 [65]	0.6202	9	C09	0.4855	15	C14	0.3874
4	C04 [66]	0.5962	10	C10	0.4539	16	C15	0.2757
5	C05 [67]	0.5912	11	C11	0.4456	17	C16	0.1880
6	C06	0.5663	12	C12	0.4350		Avg.	0.4882

6. Discussion

The frame-level system offers the advantage of potential real-time detection with a certain degree of delay. While real-time processing has yet to be studied and evaluated in our work, it is valuable for practical deployment in various applications requiring real-time audio stream processing.

However, our findings reveal certain limitations in partially spoofed audio detection. Despite the impressive improvement and robustness demonstrated by the WavLM-based features in detecting cross-domain data, in some cases, models trained with specific datasets still struggle to generalize to new, previously unseen, partially spoofed attacks. This highlights the significant challenge in adapting spoofing detection to handle novel spoofing techniques. Our experiments demonstrate that the model trained with PartialSpoof does not perform well on ADD2023-T2, while the one trained with ADD2023-T2 performs relatively better on PartialSpoof evaluation datasets.

Furthermore, partially spoofed audio attacks can employ sophisticated techniques, such as various synthesis approaches and methods to minimize artifacts between concatenated audio clips. These complex attacks pose challenges for researchers, requiring the development of advanced algorithms and, at times, extensive computational resources. To facilitate relevant studies, the need for larger and more diverse datasets becomes evident. However, acquiring such datasets can be a time-consuming and labor-intensive process, and it may also raise privacy and ethical concerns. In this case, the release of the label for ADD2023-T2-Test could be helpful for future studies on partially spoofed audio.

In future research, considering that partially spoofed audios contain audio clips from different utterances with preserved environmental acoustic backgrounds, the introduction of artifacts becomes an important consideration. Exploring the integration of traditional signal processing methods with machine learning approaches for our frame-level systems could be a worthwhile endeavor to enhance the overall effectiveness of detecting forgery attacks. A hybrid approach may prove beneficial in addressing the challenges posed by complex spoofing techniques. Additionally, considering the capability of neural networks to learn representations of artifacts, there

is an opportunity to develop novel network structures specifically designed to obtain discriminative environmental embeddings. Such advancements in network design may prove beneficial not only in detecting partially spoofed attacks but also in accurately locating fake regions within the audio clips.

In our study, while frame-level systems show promise in detecting and locating fake regions, there are still several avenues for further research to improve the robustness and generalization of these systems in practical applications. Addressing real-time processing, obtaining diverse and larger datasets, and exploring hybrid approaches with traditional signal processing methods are potential directions for future advancements in the field of partially spoofed audio detection.

7. Conclusion

This paper introduces our novel approach for detecting partially spoofed audio and locating fake regions within the audio clips at the frame level. Our method utilizes frame-level detection systems based on large-scale self-supervised pre-training models, and we extensively evaluate its performance on various publicly available datasets. The results affirm the effectiveness of our proposed method in accurately detecting partially spoofed utterances and precisely identifying manipulated regions at the same time. Notably, the acoustic features extracted by the WavLM model outperform those extracted by the Wav2Vec2 models in our frame-level detection systems. This robustness is particularly evident in cross-domain evaluations, showcasing the potential of our approach in handling diverse audio datasets. The impressive outcomes are highlighted by our state-of-the-art performance in Track 2 of the ADD 2022 challenge and securing the top position in Track 2 of the ADD 2023 challenge. However, there is still room for improvement in the practical deployment of partially spoofed audio detection models. This inspires our future efforts to design countermeasures with enhanced robustness and generalization capabilities against partially spoofed audio forgery attacks. By further refining and extending our approach, we aim to contribute to the ongoing efforts to strengthen the security and trustworthiness of audio-based applications in real-world scenarios.

[1] H. Zhao, Y. Chen, R. Wang, H. Malik, Audio Splicing Detection and Localization using Environmental Signature, Multimedia Tools and Applications 76 (2017) 13897–13927.
[2] T. Kinnunen, Z.-Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, H. Li, Vulnerability of Speaker Verification Systems against Voice Conversion Spoofing Attacks: The Case of Telephone Speech, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2012, pp. 4401–4404.
[3] S. Lyu, Deepfake Detection: Current Challenges and Next Steps, in: 2020 IEEE international conference on multimedia & expo workshops, pp. 1–6.
[4] E. N. Crothers, N. Japkowicz, H. L. Viktor, Machine-Generated Text: A Comprehensive Survey of Threat Models and Detection Methods, IEEE Access 11 (2023) 70977–71002.
[5] OpenAI, GPT-4 Technical Report (2023). arXiv:2303.08774.
[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution Image Synthesis with Latent Diffusion Models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10684–10695.
[7] J. Kim, J. Kong, J. Son, Conditional Cariational Autoencoder with Adversarial Learning for End-to-end Text-to-speech, in: International Conference on Machine Learning, 2021, pp. 5530–5540.
[8] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, H. Li, Spoofing and Countermeasures for Speaker Verification: A Survey, Speech Communication 66 (2015) 130–153.
[9] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, et al., Natural TTS Synthesis by Conditioning Wavenet on Mel Spectrogram Predictions, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2018, pp. 4779–4783.
[10] Z. Cai, C. Zhang, M. Li, From Speaker Verification to Multispeaker Speech Synthesis, Deep Transfer with Feedback Constraint, in: Proc. Interspeech, 2020, pp. 3974–3978.
[11] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, T.-Y. Liu, FastSpeech 2: Fast and High-Quality End-to-End Text to Speech (2022). arXiv:2006.04558.
[12] Z. Cai, Y. Yang, M. Li, Cross-lingual Multi-speaker Speech Synthesis with Limited Bilingual Training Data, Computer Speech & Language 77 (2023) 101427.
[13] Z. Wu, J. Yamagishi, T. Kinnunen, C. Hanilçi, M. Sahidullah, A. Sizov, N. Evans, M. Todisco, H. Delgado, ASVspoof: The Automatic Speaker Verification Spoofing and Countermeasures Challenge, IEEE Journal of Selected Topics in Signal Processing 11 (4) (2017) 588–604.
[14] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, H. Delgado, ASVspoof 2021: Accelerating Progress in Spoofed and Deepfake Speech Detection, in: Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge, pp. 47–54.
[15] X. Wang, X. Qin, T. Zhu, C. Wang, S. Zhang, M. Li, The DKU-CMRI System for the ASVspoof 2021 Challenge: Vocoder based Replay Channel Response Estimation, in: Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge, pp. 16–21.
[16] X. Meng, C. Li, L. Tian, Detecting Audio Splicing Forgery Algorithm Based on Local Noise Level Estimation, in: 2018 5th International Conference on Systems and Informatics, pp. 861–865.
[17] T. Zhu, X. Wang, X. Qin, M. Li, Source Tracing: Detecting Voice Spoofing, in: 2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, IEEE, 2022, pp. 216–220.
[18] X. Yan, J. Yi, J. Tao, C. Wang, H. Ma, T. Wang, S. Wang, R. Fu, An Initial Investigation for Detecting Vocoder Fingerprints of Fake Audio, in: Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia, 2022, p. 61–68.
[19] M. R. Kamble, H. B. Sailor, H. A. Patil, H. Li, Advances in Anti-spoofing: From the Perspective of ASVspoof Challenges, APSIPA Transactions on Signal and Information Processing 9.
[20] Z. Ali, M. Imran, M. Alsulaiman, An Automatic Digital Audio Authentication/Forensics System, IEEE Access 5 (2017) 2994–3007.
[21] M. Imran, Z. Ali, S. T. Bakhsh, S. Akram, Blind Detection of Copy-move Forgery in Digital Audio Forensics, IEEE Access 5 (2017) 12843–12855.
[22] S. Jadhav, R. Patole, P. Rege, Audio Splicing Detection using Convolutional Neural Network, in: 10th International Conference on Computing, Communication and Networking Technologies, IEEE, 2019, pp. 1–5.
[23] J. Yi, Y. Bai, J. Tao, H. Ma, Z. Tian, C. Wang, T. Wang, R. Fu, Half-Truth: A Partially Fake Audio Detection Dataset, in: Proc. Interspeech 2021, pp. 1654–1658.
[24] L. Zhang, X. Wang, E. Cooper, J. Yamagishi, J. Patino, N. Evans, An Initial Investigation for Detecting Partially Spoofed Audio, in: Proc. Interspeech 2021, pp. 4264–4268.
[25] L. Zhang, X. Wang, E. Cooper, J. Yamagishi, Multi-task Learning in Utterance-level and Segmental-level Spoof Detection, in: Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge, pp. 9–15.
[26] J. M. Martín-Doñas, A. Álvarez, The Vicomtech Audio Deepfake Detection System Based on Wav2vec2 for the 2022 ADD Challenge, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp.

9241–9245.
[27] Z. Cai, W. Wang, M. Li, Waveform Boundary Detection for Partially Spoofed Audio, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2023, pp. 1–5.
[28] J. Yi, J. Tao, R. Fu, X. Yan, C. Wang, T. Wang, C. Y. Zhang, X. Zhang, Y. Zhao, Y. Ren, L. Xu, J. Zhou, H. Gu, Z. Wen, S. Liang, Z. Lian, S. Nie, H. Li, ADD 2023: The Second Audio Deepfake Detection Challenge, in: Proceedings of IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis.
[29] M. Kajstura, A. Trawinska, J. Hebenstreit, Application of The Electrical Network Frequency (ENF) Criterion: A Case of A Digital Recording, Forensic science international 155 (2-3) (2005) 165–171.
[30] C. Grigoras, Applications of ENF Criterion in Forensic Audio, Video, Computer and Telecommunication Analysis, Forensic science international 167 (2-3) (2007) 136–145.
[31] R. Yang, Z. Qu, J. Huang, Detecting Digital Audio Forgeries by Checking Frame Offsets, in: Proceedings of the 10th ACM Workshop on Multimedia and Security, 2008, pp. 21–26.
[32] H. Malik, H. Farid, Audio Forensics from Acoustic Reverberation, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2010, pp. 1710–1713.
[33] X. Pan, X. Zhang, S. Lyu, Detecting Splicing in Digital Audios using Local Noise Level Estimation, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2012, pp. 1841–1844.
[34] D. Yan, M. Dong, J. Gao, Exposing Speech Transsplicing Forgery with Noise Level Inconsistency, Security and Communication Networks (2021) 1–6.
[35] Z. Zhang, X. Zhao, X. Yi, ASLNet: An Encoder-decoder Architecture for Audio Splicing Detection and Localization, Security and Communication Networks.
[36] Z. Zeng, Z. Wu, Audio Splicing Localization: Can We Accurately Locate the Splicing Tampering?, in: 13th International Symposium on Chinese Spoken Language Processing, IEEE, 2022, pp. 120–124.
[37] L. Wang, B. Yeoh, J. W. Ng, Synthetic Voice Detection and Audio Splicing Detection using SE-Res2Net-Conformer Architecture, in: 13th International Symposium on Chinese Spoken Language Processing, IEEE, 2022, pp. 115–119.
[38] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, C. Wang, T. Wang, Z. Tian, Y. Bai, C. Fan, S. Liang, S. Wang, S. Zhang, X. Yan, L. Xu, Z. Wen, H. Li, ADD 2022: The First Audio Deep Synthesis Detection Challenge, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp. 9216–9220.
[39] Z. Lv, S. Zhang, K. Tang, P. Hu, Fake Audio Detection Based On Unsupervised Pretraining Models, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp. 9231–9235.
[40] L. Zhang, X. Wang, E. Cooper, N. Evans, J. Yamagishi, The PartialSpoof Database and Countermeasures for the Detection of Short Fake Speech Segments Embedded in an Utterance, IEEE/ACM Transactions on Audio, Speech, and Language Processing.
[41] H. Wu, H.-C. Kuo, N. Zheng, K.-H. Hung, H.-Y. Lee, Y. Tsao, H.-M. Wang, H. Meng, Partially Fake Audio Detection by Self-Attention-Based Fake Span Discovery, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp. 9236–9240.
[42] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee, L. Juvela, P. Alku, Y.-H. Peng, H.-T. Hwang, Y. Tsao, H.-M. Wang, S. L. Maguer, M. Becker, F. Henderson, R. Clark, Y. Zhang, Q. Wang, Y. Jia, K. Onuma, K. Mushika, T. Kaneda, Y. Jiang, L.-J. Liu, Y.-C. Wu, W.-C. Huang, T. Toda, K. Tanaka, H. Kameoka, I. Steiner, D. Matrouf, J.-F. Bonastre, A. Govender, S. Ronanki, J.-X. Zhang, Z.-H. Ling, ASVspoof 2019: A Large-scale Public Database of Synthesized, Converted and Replayed Speech, Computer Speech & Language 64 (2020) 101114.
[43] A. Baevski, Y. Zhou, A. Mohamed, M. Auli, Wav2vec 2.0: A Framework for Self-supervised Learning of Speech Representations, Advances in Neural Information Processing Systems 33 (2020) 12449–12460.
[44] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, et al., Wavlm: Large-scale Self-supervised Pre-training for Full Stack Speech Processing, IEEE Journal of Selected Topics in Signal Processing 16 (6) (2022) 1505–1518.
[45] A. Babu, C. Wang, A. Tjandra, K. Lakhotia, Q. Xu, N. Goyal, K. Singh, P. von Platen, Y. Saraf, J. Pino, A. Baevski, A. Conneau, M. Auli, XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale, in: Proc. Interspeech 2022, pp. 2278–2282.
[46] N. Vaessen, D. A. Van Leeuwen, Fine-Tuning Wav2Vec2 for Speaker Recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp. 7967–7971.
[47] Z. Fan, M. Li, S. Zhou, B. Xu, Exploring wav2vec 2.0 on Speaker Verification and Language Identification, in: Proc. Interspeech 2021, pp. 1509–1513.
[48] Y. Xie, Z. Zhang, Y. Yang, Siamese Network with Wav2vec Feature for Spoofing Speech Detection, in: Proc. Interspeech 2021, pp. 4269–4273.
[49] X. Wang, J. Yamagishi, Investigating Self-Supervised Front Ends for Speech Spoofing Countermeasures, in: Proc. The Speaker and Language Recognition Workshop (Odyssey 2022), pp. 100–106.
[50] A. Conneau, A. Baevski, R. Collobert, A. Mohamed, M. Auli, Unsupervised Cross-Lingual Representation Learn-

ing for Speech Recognition, in: Proc. Interspeech 2021, pp. 2426–2430.
[51] J. Grosman, Fine-tuned XLSR-53 Large Model for Speech Recognition in Chinese, https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-chinese-zh-cn (2021).
[52] J. Grosman, Fine-tuned XLSR-53 Large Model for Speech Recognition in English, https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-english (2021).
[53] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, G. Weber, Common Voice: A Massively-Multilingual Speech Corpus, in: Proceedings of the 12th Language Resources and Evaluation Conference, 2020, pp. 4218–4222.
[54] K. Park, T. Mulc, CSS10: A Collection of Single Speaker Speech Datasets for 10 Languages, Proc. Interspeech 2019 1566–1570.
[55] OpenSLR, ST-CMDS-20170001_1, Free ST Chinese Mandarin Corpus, https://www.openslr.org/38/ (2017).
[56] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, N. Evans, AASIST: Audio Anti-spoofing using Integrated Spectro-temporal Graph Attention Networks, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2022, pp. 6367–6371.
[57] J. weon Jung, S. bin Kim, H. jin Shim, J. ho Kim, H.-J. Yu, Improved RawNet with Feature Map Scaling for Text-Independent Speaker Verification Using Raw Waveforms, in: Proc. Interspeech 2020, 2020, pp. 1496–1500.
[58] H. Tak, M. Todisco, X. Wang, J. weon Jung, J. Yamagishi, N. Evans, Automatic Speaker Verification Spoofing and Deepfake Detection Using Wav2vec 2.0 and Data Augmentation, in: Proc. The Speaker and Language Recognition Workshop, 2022, pp. 112–119.
[59] J. An, S. Cho, Variational Autoencoder Based Anomaly Detection Using Reconstruction Probability, Special lecture on IE 2 (1) (2015) 1–18.
[60] D. P. Kingma, M. Welling, Auto-encoding Variational Bayes (2013). arXiv:1312.6114.
[61] D. Snyder, G. Chen, D. Povey, MUSAN: A Music, Speech, and Noise Corpus (2015). arXiv:1510.08484.
[62] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, S. Khudanpur, A Study on Data Augmentation of Reverberant Speech for Robust Speech Recognition, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2017, pp. 5220–5224.
[63] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention Is All You Need, in: Advances in Neural Information Processing Systems, 2017, pp. 5998–6008.
[64] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar, W.-N. Hsu, Voicebox: Text-Guided Multilingual Universal Speech Generation at Scale (2023). arXiv:2306.15687.
[65] K. Li, X.-M. Zeng, J.-T. Zhang, Y. Song, Convolutional Recurrent Neural Network and Multitask Learning for Manipulation Region Location, in: Proceedings of IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis, 2023.
[66] J. M. Martín-Doñas, A. Álvarez, The Vicomtech Partial Deepfake Detection and Location System for the 2023 ADD Challenge, in: Proceedings of IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis, 2023.
[67] J. Li, L. Li, M. Luo, X. Wang, S. Qiao, Y. Zhou, Multi-grained Backend Fusion for Manipulation Region Location of Partially Fake Audio, in: Proceedings of IJCAI 2023 Workshop on Deepfake Audio Detection and Analysis, 2023.