ASLNet: An Encoder-Decoder Architecture for Audio Splicing Detection and Localization

Zhenyu Zhang,1,2 Xianfeng Zhao,1,2 and Xiaowei Yi1,2

1State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
2School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100093, China

Correspondence should be addressed to Xianfeng Zhao; zhaoxianfeng@iie.ac.cn

Received 16 November 2021; Accepted 3 February 2022; Published 29 March 2022

Copyright © 2022 Zhenyu Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Audio splicing means inserting an audio segment into another audio, which presents a great challenge to audio forensics. In this paper, a novel audio splicing detection and localization method based on an encoder-decoder architecture (ASLNet) is proposed. Firstly, an audio clip is divided into several small audio segments according to the size of the smallest localization region \( L_{slr} \), and the acoustic feature matrix and corresponding binary ground truth mask are created from each audio segment. Then, we concatenate acoustic feature matrices from all segments of an audio clip into an acoustic feature matrix and send it to a fully convolutional network (FCN) based encoder-decoder architecture which consists of a series of convolutional, pooling and transposed convolutional layers to get a binary output mask. Next, the binary output mask is divided into small segments according to the \( L_{slr} \), and the ratio \( \rho \) of the number of elements equal to one to the number of all elements in a small segment is calculated. Finally, we compare \( \rho \) with the predetermined threshold \( T \) to determine whether the corresponding audio segment is spliced. We evaluate the effectiveness of the proposed ASLNet on four datasets produced from publicly available speech corpus. Extensive experiments show that the best detection accuracy of ASLNet for the intradatabase and cross-database evaluation can achieve 0.9965 and 0.9740 receptively, which outperforms the state-of-the-art method.

1. Introduction

With the widespread availability of audio manipulating tools available online, it has become easier to create forgeries without a perceptual trace. Audio splicing, inserting an audio segment at the start, middle or at the end of another audio recording, is one of the most common types of audio forgery. The spliced audios reduce the reliability of judicial evidence and defeat intellectual property protection. In addition, these spliced audios can be used for fake news spreading which will make a negative impact on society. Therefore, the ability to detect whether an audio recording has been spliced is a task of great interest in the audio forensics community [1].

In the past decades, many kinds of research have been investigated for audio splicing detection and localization. As the audio splice operations cause the inconsistency of the noise level, researchers have developed audio splicing detection methods based on local noise levels of audio signals [2–6]. However, when the signal-to-noise ratio between the spliced segments is close or even the same, the performance of the noise levels based audio splicing detection methods will decrease sharply. In addition, based on the fact that inserting an audio segment into another audio recording leads to anomalous variations of the electric network frequency (ENF) signal, several kinds of research [7–9] have shown that it is an efficient way to detect spliced audio via the analysis of ENF signal. Whereas due to legal restrictions, it is difficult to obtain concurrent reference datasets of power systems, which makes the ENF based audio splicing detection methods difficult to implement [10]. Nowadays, the convolutional neural network has been introduced to audio splicing detection [11]. However, the neural network based methods only deduce whether the given audio has been spliced and can not localize the spliced
segment. Based on the above overview, although some audio splicing detection and localization methods have achieved effective performance, the new techniques are required to improve detection and localization performance. To the best of our knowledge and literature review, the encoder-decoder architecture has not been used for the researches of audio splicing detection and localization.

In this paper, we describe a novel audio splicing detection and localization method based on an encoder-decoder architecture (ASLNet). Firstly, we divide the audio into several small audio segments according to the size of the smallest localization region \(L_{slr}\), and extract the acoustic feature matrix and corresponding ground truth binary mask where the original parts are black point (labeled 0) and the spliced parts are the white point (labeled 1) from each audio segment. Then, we concatenate acoustic feature matrices from all segments of an audio clip into an acoustic feature matrix and send it to an encoder-decoder network which consists of a series of convolutional, pooling and transposed convolutional layers to get a binary output mask. Next, the binary output mask is divided into small segments according to the \(L_{slr}\) and the ratio \(\rho\) of the number of elements equal to one to the number of all elements in a small segment is calculated. Finally, we compare \(\rho\) with the predetermined threshold \(T\) to determine whether the corresponding audio segment is spliced. The cross-database and intradatabase evaluations on four datasets produced from publicly available speech corpus demonstrate the efficiency of ASLNet for audio splicing detection and localization. To verify and reproduce the experiments presented in the paper, all of our source codes and datasets are now available via GitHub: https://github.com/Amforever/ASLNet.

The remainder of this paper is organized as follows. Section 2 introduces the related works about audio splicing detection and fully convolutional network briefly. We formally describe the proposed ASLNet in Section 3. In Section 4, the datasets and evaluation metrics are introduced. The comparative experimental results for audio splicing detection and localization are presented in Section 5. Finally, the conclusion of this paper is drawn in Section 6.

2. Related Work

2.1. Audio Splicing Detention. During the last decade, many audio splicing detection and localization methods based on ENF signal and local noise levels have been proposed. Lin and Kang [8] proposed a wavelet-filtered ENF signal to highlight the abnormal ENF variations and employed autoregressive coefficients to train the classifier under a supervised-learning framework. Mao et al. [9] utilized the multiple ENF features as input eigenvectors to the convolutional neural networks for detecting spliced audio. Meng et al. [4] used the spectral entropy method to determine the length of each syllable and calculated the variance of the background noise of each syllable, then judged whether there is an operation of the heterogeneous splicing tampering in the audio by comparing the similarities between the variance of the background noise of each syllable. Yan et al. extracted the noise signal of the suspected speech by a parameter-optimized noise estimation algorithm and calculated the statistic of Mel frequency features of the estimated noise signal to detect the splicing trace [6]. With the successful achievement of convolutional neural networks in many fields, Jadhav et al. [11] first introduced the convolutional neural network to audio splicing detection, which directly fed the spectrogram of an audio clip to the convolutional neural network and make the classification.

2.2. Fully Convolutional Network. The fully convolutional network (FCN) is an encoder-decoder architecture with convolutional blocks and transposed convolutional blocks (as shown in Figure 1), which can efficiently learn to make dense predictions for per-pixel tasks. Long et al. [12] first developed FCN for the task of semantic segmentation by converting all fully connected layers of common classification networks to convolutional ones. They compared three classification architectures: AlexNet [13], GoogLeNet [14], and VGG16 [15], which found that FCN adopting VGG16 as the backbone (FCN-VGG16) performed better than that with adopting AlexNet and GoogLeNet. Salloff et al. [16] first proposed a single-task fully convolutional network (SFCN) and multi-task fully convolutional network (MFCN) based on the FCN-VGG16 architecture to the image splicing localization problem. The SFCN and MFCN were trained on ground truth masks, which was a binary mask that classified each pixel in an image as spliced or authentic. Segal et al. [17] proposed SpeechYOLO to localize boundaries of utterances within the input signal, which first applied FCN from the vision domain on the speech recognition domain. It has been shown that the FCN-VGG16 achieved good results in the image splicing localization problem. In this paper, inspired by [16, 17], we apply the FCN based encoder-decoder architecture from the vision domain to the audio splicing detection and localization domain, by treating the acoustic feature of the spliced audio segment as objects.

3. Proposed Work

3.1. The Framework of ASLNet. In this paper, we describe a novel audio splicing detection and localization method based on an encoder-decoder architecture. The whole procedure of ASLNet is shown in Figure 2, which includes the training phase and the test phase. For the training phase, firstly, we divide the audio signal into several small audio segments according to the \(L_{slr}\) (i.e. the predefined size of the smallest localization region that includes a number of sample points), and create the acoustic feature matrix and corresponding ground truth binary mask from each audio segment. Then we concatenate acoustic feature matrices from all audio segments into an acoustic feature matrix and send it to a fully convolutional network (FCN) based encoder-decoder architecture, which consists of a series of convolutional, pooling and transposed convolutional layers, to get a binary output mask. Finally, the binary output mask is compared with the ground truth mask for computing the output error of the current set of neural network’s weights, which are used for backward propagation of the neural network.
For the test phase, firstly, we create an acoustic feature matrix from a whole audio clip, which is similar to that in the training phase. Then we sent the acoustic feature matrix into the trained encoder-decoder model to get a binary output mask and divide the binary mask into small segments according to $L_{slr}$. Finally, the $\rho$ of the number of elements equal to one to the number of all elements in a small segment is calculated, which is compared with the predetermined threshold $T$ to determine whether the corresponding audio segment is spliced. Specifically, if $\rho$ is larger than the predetermined threshold $T$ ($\rho > T$), we judge the audio segment as the spliced; otherwise, the audio segment is judged as the original.
3.2. Acoustic Feature and Binary Ground Truth Mask.

With the development of fake audio detection, many kinds of acoustic features have been proposed to improve detection performances. Linear frequency cepstral coefficient (LFCC) and Mel-frequency cepstral coefficient (MFCC) are two of the most used acoustic features for the detection of fake speech [18–20]. In this paper, we extract LFCC and MFCC features from the audio signals as the input data of the ASLNet. The block diagrams of LFCC and MFCC extractions are shown in Figure 3.

As can be seen from Figure 3, the extractions of the LFCC and MFCC features are identical except for the kinds of filter banks. The LFCC feature adopts a Linear-frequency filter bank that equally covers all audio frequency ranges and considers them of equal importance. However, the MFCC, considering the nonlinear property of the human hearing system with respect to different frequencies, adopts the Mel-frequency filter bank. The detailed extraction procedure of MFCC is described as follows. Firstly, we make pre-emphasis to increase the energy of signal at the higher frequency and compute the short-time Fourier transform (STFT) of the signal using a periodic Hamming window of length 2048 samples with an overlap of 512 samples. Then, we map the powers of the spectrum on the Mel scale by Mel-filter bank. Next, discrete cosine transform is applied to calculate the transformed coefficients that contain important amounts of energy. In this paper, to extract the LFCC and MFCC matrix from an audio clip, we defined the smallest length of the audio segments that are located as 16000 samples (i.e., $L_{sr} = 16000$). We pick the first 24 coefficients as static LFCC and MFCC features, while the delta and acceleration coefficients are calculated and concatenated to the static coefficients to form the 72 feature vector. Therefore, the LFCC and MFCC feature matrices have a shape of $72 \times 32$, where 72 is the number of coefficients and 32 is the number of frames. In addition, to train the encoder-decoder network in per-pixel, we design a binary ground truth mask for each LFCC or MFCC feature matrix. The binary ground truth mask consists of 0 or 1 elements, with the size of $72 \times 32$. For the original audio segment, every element in the corresponding ground truth mask is 0, while every element in the corresponding ground truth mask is 1 for the spliced audio segment. After extracting all LFCC or MFCC matrices from all segments of an audio clip, we further concatenate them as the final LFCC or MFCC matrix of a whole audio clip.

3.3. The Encoder-Decoder Architecture of ASLNet.

The encoder-decoder architecture is a common architecture of current semantic segmentation algorithms, which is composed of an encoder and decoder. The encoder performs convolution and down-sampling to capture contextual information, while the decoder is responsible for deconvolution and up-sampling to predict pixel-wise class labels. As significant success with encoder-decoder architectures so far, the FCN [12], U-Net [21] and SegNet [22] have been proposed for image pixel-wise segmentation. In this paper, the base network architecture of ASLNet is the modified FCN-VGG16, which is composed of a VGG16 encoder and the decoder with a skip connection. The VGG16 encoder is aimed at capturing contextual representations of the acoustic feature, while the decoder is aimed to transform the intermediate feature maps as the binary output mask.

As described in Figure 2, we stack five VGG blocks to construct a VGG16 encoder, where each VGG block consists of two or three convolutional blocks and is followed by a max-pooling layer, resulting in a total of 13 convolutional layers and five max-pooling layers. The convolutional block consists of a convolutional layer, a batch normalization layer and a rectified linear unit (ReLU) activation function. For all convolutional layers, the same kernel size $3 \times 3$ is applied and the convolution stride is fixed to 1. In addition, a padding size of 1 to keep the size of the output after each convolutional layer. The max-pooling layers are with the size of $2 \times 2$ and the stride of 2, which are applied to halve the resolution after each VGG block. The decoder aims to reconstruct the binary mask from the essential information extracted by the VGG16 encoder, which consists of two transposed convolutional layers and a softmax layer. The first transposed convolution is with the kernel size of $4 \times 4$ and stride of 2, while the second transposed convolution is with the kernel size of $32 \times 32$ and stride of 16. In addition, a skip connection from the fourth VGG block to the first transposed convolution is used to aggregate the feature learned at low layers with the higher layers. The softmax layer is used for calculating the probability that the element is coming from the spliced audio segment.

4. Datasets and Evaluation Metrics

4.1. Datasets. In this paper, we create four spliced audio datasets: English language dataset consists of audio clips with 2 seconds duration (ENSet2s), English language dataset consists of audio clips with 3 seconds duration (ENSet3s), Chinese language dataset consists of audio clips with 2 seconds duration (CNSet2s), and Chinese language dataset consists of audio clips with 3 seconds duration (CNSet3s) based on two publicly available speech corpus to evaluate the performance of the proposed ASLNet. The TIMIT database [23], which is composed of 6300 English language audio recordings from 438 male and 192 female speakers and each audio recording is about from two seconds to six seconds, is used to generate original and spliced audio of ENSet2s and ENSet3s. In addition, the FMFCC-A database [24], which contains 10,000 genuine Chinese recordings, is used to generate original and spliced audio clips of CNSet2s and CNSet3s. The aim of the creation of audio clips with 2 seconds and 3 seconds is to validate the influence of splicing positions (splicing audio segment at the middle or the end of another audio recording as in Figure 4) to detection performance of ASLNet. All audio clips in the experimental datasets are delivered in the WAV file with 16kHz, 16-bit and mono format by using FFmpeg software.

The procedure of constructions of the ENSet2s and ENSet3s is as in Figure 5. Firstly, the audio clips of the TIMIT database are cut into audio clips with one second duration ($1 - s$), audio clips with two seconds duration ($2 - s$) and audio clips with three seconds duration ($3 - s$).
The 2-s audio clips and 3-s audio clips are used as the original samples of ENSet2s and ENSet3s respectively. Then, we randomly select two non-homologous 1-s audio clips (coming from two different audio recordings) and concatenate them as a spliced audio clip that the splicing positions are at the end of another audio clip (as Figure 4(d)). As the number of combinations is too high, we randomly make 15,173 2-s spliced audio clips to construct ENSet2s. Furthermore, we randomly selected a 1-s audio clip and a 2-s audio clip, and insert the 1-s audio clip into the middle of the 2-s audio clip (as Figure 4(e)). After randomly crating 19,783 3-s spliced audio clips, we produced ENSet3s completely. The procedures of creating CNSet2s and CNSet3s are similar to those of the ENSet2s and ENSet3s. More details about the audio clips in ENSet2s, ENSet3s, CNSet2s and CNSet3s are shown in Table 1.

4.2. Evaluation Metrics. In this paper, we evaluate the performances of audio splicing detection and localization methods using three metrics. The true positive rate is the proportion of the spliced audio clips that are correctly identified, which shows how good the model is at detecting the spliced audio. The true negatives rate represents the probability that original audio clips are correctly judged as nonspliced audio clips, which suggests how good the model is at identifying original audio. The detection accuracy is the ratio of the total number of correctly classified audio clips to the total number of audio clips in the test dataset. In this paper, the false acceptance error rate, false reject rate and detection accuracy are denoted as $P_{tp}$, $P_{tn}$ and $P_{acc}$ respectively. Mathematically, the $P_{tp}$, $P_{tn}$ and $P_{acc}$ can be expressed as follows:

$$
P_{tp} = \frac{TP}{TP + FN},
$$

$$
P_{tn} = \frac{TN}{TN + FP},
$$

$$
P_{acc} = \frac{TP + TN}{TP + TN + FP + FN},
$$

where TP, TN, FP, and FN represent the number of true positives, true negatives, false positives and false negatives respectively. It can be seen that the higher $P_{tp}$, $P_{tn}$ and $P_{acc}$, the better the detection capability of the models. In this paper, to obtain convincing results, $P_{tp}$, $P_{tn}$, and $P_{acc}$ are averaged over 10 testing splits.

5. Experiments

In this section, we evaluate the performance of the proposed ASLNet and compare it with a neural network based audio splicing detection model (named Jadhav [11]) on four datasets. The Jadhav et al. [11] has a good feature learning ability and provides competitive performance in audio splicing detection.
5.1. Implementation Details. For training and testing neural networks, we randomly divide the datasets into three sets: a training set (60% of the dataset), a development set (20% of the dataset) and an evaluation set (20% of the dataset). During the training phase of neural networks, the training set is used to train the model and the development set is used to validate the fitness of the model. The weights of neural networks are initialized with random numbers following a Gaussian distribution and the biases are initialized to be zero. A mini-batch of 64 audio clips is selected from the training dataset to estimate the true gradient and the training dataset is shuffled after each epoch. For optimization, we use stochastic gradient descent optimization algorithms AdaDelta [25], which can dynamically adapt the learning rate based on the gradient, with its default setting values. The model validation performance on the development dataset is calculated after each epoch. The training process of the neural networks has run for 200 epochs, and the model that achieves the best validation performance is used for testing.

When the well-trained neural network model has been chosen, we make the test with a mini-batch of 64 audio clips which are randomly sampled from the evaluation dataset and determine whether the audio clip is spliced according to the predetermined threshold. After all audio clips on the evaluation dataset are recycled without replacement, we calculate three metrics for the detection model. To obtain convincing results, we randomly split the training, development and evaluation sets for 10 times and average all results to get the final \( P_{tp}, P_{tn} \) and \( P_{acc} \).

5.2. The Selection of Threshold and Acoustic Feature. As described in Section 3.1, we determine whether the audio segment is spliced by comparing \( P \) from the binary output mask and predetermined threshold \( T \). The selection of an appropriate threshold \( T \) is of great concern. If the threshold is too large, some spliced audio segments may be omitted; if too low, false detection may exist. In addition, the different acoustic features explore the properties of the audio signal from various aspects. It has been shown choosing an appropriate acoustic feature for fake speech detection is important [26], so the acoustic features need to be selected for ASLNet. This part will do the experiment to prove the superiority of our selection of threshold and acoustic features.

To show the localization ability of the ASLNet more clearly, we present output examples from ASLNet for a 2 - s spliced audio clip and a 3 - s spliced audio clip on Figure 6. Each row in Figure 6 shows the audio waveform, the LFCC feature matrix, the ground truth mask, the output mask of ASLNet after 2 epochs and the output mask of ASLNet after 200 epochs. We can see from Figure 6 that certain elements of the binary output mask are predicted with a wrong label after 2 epochs. However, the binary output mask of ASLNet after 200 epochs has closely resembled the ground truth mask of the spliced audio clip. It is shown that as the number of epochs increases, the detection performance of ASLNet has been greatly improved, which achieves finer spliced segments localization.

To select the appropriate threshold \( T \) and acoustic feature, in this paper, the thresholds \( T = [0.5, 0.6, 0.7, 0.8] \) are applied to find an appropriate \( T \), while MFCC and LFCC features are used to select the appropriate acoustic feature as the input data of ASLNet. The comparison results are provided in Table 2. As we can see from Table 2, with an increment of threshold \( T \), the \( P_{tp} \) of ASLNet is increase, which achieved the \( P_{tp} = 0.9955, P_{tp} = 0.9132, P_{tp} = 0.9743 \) and \( P_{tp} = 0.8869 \), on CNSet3s, ENSet3s, CNSet2s, ENSet2s respectively, but the \( P_{tn} \) and \( P_{acc} \) have a little decrement. For different thresholds the detection accuracy of ASLNet with MFCC features is higher than 0.9, which means the proposed
ASLNet has an effectively performance for audio splicing detection and Localization. Considering that $P_{\text{acc}}$ shows the overall performance of the model at detecting a spliced audio, $T$ is set at 0.5. In addition, the $P_{\text{tp}}$, $P_{\text{tn}}$ and $P_{\text{acc}}$ of ASLNet with MFCC features on four datasets are all higher than those with LFCC features, which means that the MFCC feature is the more appropriate acoustic feature for audio splicing detection and localization. Hence, the MFCC feature is selected as the suitable acoustic feature. Hence, we choose the $T = 0.5$ and MFCC feature for ASLNet in the following experiments.

5.3. **Intradatabase Evaluation.** In this part, we would like to compare the performance of ASLNet with a neural network based audio splicing detection model Jadhav et al. [11] under the intradatabase evaluation scenario. The intradatabase evaluation has been followed in most of the existing works because it can show detection ability on some specific databases. The $P_{\text{tp}}$, $P_{\text{tn}}$ and $P_{\text{acc}}$ of Jadhav et al. [11] and proposed ASLNet on four datasets are summarized in Table 3.

As we can see from Table 3, the best detection results are achieved by ASLNet on CNSet3s, which are $P_{\text{tp}}$, $P_{\text{tn}}$ and $P_{\text{acc}}$ of 0.9938, 0.9979 and 0.9965 respectively. We can easily find that the better detection performance is achieved by using the proposed ASLNet, because the modified FCN-VGG16 can effectively capture contextual information of acoustic features. The $P_{\text{tp}}$ of detection models is slightly lower than $P_{\text{tn}}$ on all datasets, which shows the spliced audio clips are more difficult to detection model than the original audio clips. In addition, the performance of both ASLNet and Jadhav et al. [11] on CNSet2s and CNSet3s are better than those on ENSet2s and ENSet3s, which might be because there are more audio samples on CNSet2s and CNSet3s for training neural networks. Comparing the performance of detection methods on the datasets of the splicing at the end (ENSet2s and CNSet2s) and splicing at the end (ENSet3s and CNSet3s), we can find it is more difficult to detect the spliced audio that spliced at the end than that at the middle. It might be due to that splicing at the middle will provide more contextual information to the neural networks. In a word, the $P_{\text{acc}}$ of ASLNet and Jadhav et al. [11] are higher than 0.9, which means that both methods have great performances for audio splicing detection under intradatabase evaluation scenario.

5.4. **Cross-Database Evaluation.** To evaluate the generalization of the detection model for different databases, we conduct a cross-database evaluation on four datasets. As the
trained networks take fixed-size acoustic features, we only make the cross-database evaluation on the ENSet2s and CNSet2s, and the ENSet3s and CNSet3s respectively. To compare the performance of ASLNet with the state-of-the-art audio splicing detection models, we also make cross-database evaluations for Jadhav et al. [11]. Table 4 describes the detecting results of ASLNet and Jadhav et al. [11] for cross-database evaluation.

From Table 4, we can see that the average $P_{\text{acc}}$ of detection models is approximately decreased by 0.0632 compared with that under intradatabase evaluation scenario, which means cross-database scenario is more difficult for audio splicing detection. In addition, it is observed that the ASLNet trained on the ENSet2s and ENSet3s deliver slightly better performance than learned on the CNSet2s and CNSet3s. It also found that the ASLNet also has better generalization than Jadhav et al. [11] under cross-database evaluation scenarios. Specificity, the ASLNet has achieved $P_{\text{acc}}$ of 0.9740 and 0.9420 respectively, when tested on CNSet3s and CNSet2s, which demonstrates the promising generalization capability of ASLNet. From Tables 3 and 4, we can see that the ASLNet is better than the Jadhav et al. [11] under different scenarios.

### 6. Conclusion

In this paper, we propose a novel audio splicing detection and localization method based on an encoder-decoder architecture. The base network architecture of ASLNet is the modified FCN-VGG16, which is composed of a VGG16 encoder and the decoder with a skip connection. The VGG16 encoder is used to extract contextual representations of the acoustic features, while the decoder transforms the intermediate feature map as the binary output mask for judging whether the audio segment is spliced. Four spliced audio datasets ENSet2s, ENSet3s, CNSet2s and CNSet3s that are produced from publicly available speech corpus are used to evaluate the performance of the proposed ASLNet. Experimental results prove the effectiveness of the proposed ASLNet for audio splicing detection and localization task, which significantly improves the detection performance compared with a current state-of-the-art approach. In addition, according to the results of the cross-database evaluation, it demonstrates the promising generalization capability of the ASLNet. In practical scenarios, the size of the smallest localization region $L_{\text{slr}}$ and the threshold $T$ can be adjusted to improve the universality of the ASLNet. In the future, we will explore more effective deep neural network based methods for audio splicing detection and localization.

### Data Availability

The experimental datasets are created using the TIMIT and FMFCC-A databases, which are publicly available.

### Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

### Acknowledgments

This work was supported by National Key Technology Research and Development Program under 2019QY2202 and 2020AAA0140000.

### References

[1] M. Zakariah, M. K. Khan, and H. Malik, “Digital multimedia audio forensics: past, present and future,” Multimedia Tools and Applications, vol. 77, no. 1, pp. 1009–1040, 2018.
[2] X. Pan, X. Zhang, and S. Lyu, “Detecting splicing in digital audio using local noise level estimation,” in Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1841–1844, Kyoto, Japan, March 2012.

[3] H. Zhao, Y. Chen, R. Wang, and H. Malik, “Audio splicing detection and localization using environmental signature,” Multimedia Tools and Applications, vol. 76, no. 12, pp. 13897–13927, 2017.

[4] X. Meng, C. Li, and L. Tian, “Detecting audio splicing forgery algorithm based on local noise level estimation,” in Proceedings of the 2018 International Conference on Systems and Informatics (ICSAI), pp. 861–865, Nanjing, China, November 2018.

[5] D. Caporelli, C. Borrelli, P. Bestagini, F. Antonacci, A. Sarti, and S. Tubaro, “Speech audio splicing detection and localization exploiting reverberation cues,” in Proceedings of the 2020 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1–6, New York, USA, December 2020.

[6] D. Yan, M. Dong, and J. Gao, “Exposing speech transsplicing forgery with noise level inconsistency,” Security and Communication Networks, vol. 2021, Article ID 6659371, 6 pages, 2021.

[7] P. A. A. Esquef, J. A. Apolinário, and L. W. P. Biscainho, “Edit detection in speech recordings via instantaneous electric network frequency variations,” IEEE Transactions on Information Forensics and Security, vol. 9, no. 12, pp. 2314–2326, 2014.

[8] X. Lin and X. Kang, “Supervised audio tampering detection using an auto regressive model,” in Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2142–2146, New Orleans, USA, March 2017.

[9] M. Mao, Z. Xiao, X. Kang, X. Li, and L. Xiao, “Electric network frequency based audio forensics using convolutional neural networks,” in Proceedings of the IFIP International Conference on Digital Forensics, pp. 253–270, New Delhi, India, January 2020.

[10] A. Hajj-Ahmad, R. Garg, and M. Wu, “ENF-based region-of-recording identification for media signals,” IEEE Transactions on Information Forensics and Security, vol. 10, no. 6, pp. 1125–1136, 2015.

[11] S. Jadhav, R. Patole, and P. Rege, “Audio splicing detection using convolutional neural network,” in Proceedings of the 2019 International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1–5, Kanpur, India, July 2019.

[12] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440, Boston, MA, USA, June 2015.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105, Lake Tahoe, NV, USA, 2012.

[14] C. Szegedy, W. Liu, Y. Jia et al., “Going deeper with convolutions,” in Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9, June 2015.

[15] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” https://arxiv.org/abs/1409.1556.

[16] R. Salloum, Y. Ren, and C.-C. Jay Kuo, “Image splicing localization using a multi-task fully convolutional network (MFCN),” Journal of Visual Communication and Image Representation, vol. 51, pp. 201–209, 2018.

[17] Y. Segal, T. S. Fuchs, J. Keshet, and Speechyolo, “Detection and localization of speech objects,” in Proceedings of the 2019 Annual Conference of the International Speech Communication Association (INTERSPEECH), pp. 1–6, Graz, Australia, September 2019.

[18] S. Novoselov, A. Kozlov, G. Lavrentyeva, K. Simonchik, and V. Shchemelinin, “STC anti-spoofing systems for the ASVspoof 2015 challenge,” in Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5475–5479, Lujiazui, March 2016.

[19] D. Yan and T. Wu, “Detection of various speech forgery operations based on recurrent neural network,” in Proceedings of the International Conference on Security and Privacy in Digital Economy, pp. 415–426, Quzhou, China, November 2020.

[20] Z. Zhang, X. Yi, and X. Zhao, “Fake speech detection using residual network with transformer encoder,” in Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security, pp. 13–22, Virtual event Belgium, June 2021.

[21] O. Ronneberger, P. Fischer, and T. Brox, “U-net: convolutional networks for biomedical image segmentation,” in Proceedings of the International Conference on Medical Image Computing and Computer-assisted Intervention, pp. 234–241, Munich, Germany, October 2015.

[22] V. Budhrinarayanan, A. Kendall, and R. Cipolla, “SegNet: a deep convolutional encoder-decoder architecture for image segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 12, pp. 2481–2495, 2017.

[23] V. Zue, S. Seneff, and J. Glass, “Speech database development at MIT: TIMIT and beyond,” Speech Communication, vol. 9, no. 4, pp. 351–356, 1990.

[24] Z. Zhang, Y. Gu, X. Yi, and X. Zhao, “FMFCC-a: a challenging Mandarin dataset for synthetic speech detection,” in Proceedings of the Digital Forensics and Watermarking: 20th International Workshop, IWDW 2021, p. 117, Beijing, China, November 2021.

[25] M. D. Zeiler, “Adadelta: an adaptive learning rate method,” 2012, https://arxiv.org/abs/1212.5701.

[26] M. Sahidullah, T. Kinnunen, and C. Hanilçi, “A comparison of features for synthetic speech detection,” in Proceedings of the 2015 Annual Conference of the International Speech Communication Association (INTERSPEECH), pp. 2087–2091, Dresden, Germany, September 2015.