# MFCC vs. LFCC for Audio Deepfake Detection: The Role of Delta Features and Input Length

Karla Schäfer
*Fraunhofer SIT, ATHENE*
Darmstadt, Germany
karla.schaefer@sit.fraunhofer.de

Martin Steinebach
*Fraunhofer SIT, ATHENE*
Darmstadt, Germany
martin.steinebach@sit.fraunhofer.de

***Abstract*—Basic feature representations like MFCCs and LFCCs allow resource efficient and explainable feature extraction for e.g. audio deepfake detection. Despite the frequent utilisation of these feature representations, a comprehensive examination of the number of coefficients employed and the impact of delta and double delta values remains to be undertaken. We analysed MFCCs and LFCCs combined with four classifiers, using in-domain and out-of-domain test sets. MFCCs performed superior on out-of-domain data, LFCC on the in-domain test set. The combination of lower amounts of coefficients with longer audio inputs, in conjunction with the utilisation of delta and double delta features, yielded enhanced generalisable results. For instance, for ResNet34 with 128 coefficients we calculated an EER of 65.15% on the out-of-domain test set, with 20 coefficients we calculated an EER of 29.71%. Furthermore, we identified specific patterns in the MFCCs when employed with various classifiers. For all classifiers, lower MFCCs (0, 1) were identified as contributing to a classification as bona-fide, whereby higher MFCCs contributed to a classification as spoof for all detectors.**

***Index Terms*—audio deepfakes, detection, LFCC, MFCC, explainability.**

## I. Introduction

Mel-frequency cepstral coefficients (MFCCs) are known for being a good performing feature representation of audio features for speech processing tasks. A similar feature representation are linear-frequency cepstral coefficients (LFCCs), the only difference between them being the filter bank, which divides the signal into several components. LFCC filter bank coefficients covering all speech frequency ranges equally and with equal importance (linear frequency filter bank) and MFCC filter banks using the mel scale. For audio deepfake detection, recently, often self-supervised learning (SSL) based feature extraction models are used, achieving superior performance when viewing the generalizability of the detectors. But, using SSL-based models for feature extraction, no explanation for the resulting classification can be given. In addition, SSL models necessitate substantial resources. Therefore, in this study, we view MFCCs and LFCCs as feature representation, using different amounts of coefficients and combining them with four classifiers (LCNN, ResNet18, ResNet34 and MesoNet). Besides evaluating the performance of the detectors on the ASVspoof 2019 LA (ASV19) dataset (in-domain evaluation) we tested the detectors on the in-the-wild (ITW) dataset [1] for its generalizability, i.e. out-of-domain performance.

This research work was supported by ATHENE in the project DREAM.

Furthermore, as each speech signals are time-variant signals and in a constant flux, the acoustical signal is more accurately described as a sequence of transitions between phonemes. A common method for extracting information about such transitions is to determine the first difference of signal features, known as the delta ($\Delta$) of a feature, and the second difference, known as the double delta ($\Delta\Delta$). Consequently, we evaluate the impact of additionally using the $\Delta$ and $\Delta\Delta$ of the coefficients as feature input.

We viewed various papers using LFCCs and MFCCs as feature representation. This is the first study, evaluating the performance of MFCCs and LFCCs with various classifiers using different amounts of coefficients and differently combining them with its $\Delta$ and $\Delta\Delta$ features. Furthermore, recent works [1] found that longer audio recordings used as input led to improved detection results. Therefore, we tested the effect of the audio length as input, using 3 seconds and 30 seconds recordings. Using coefficients as input to the classifier also allowed us to evaluate on which basis the detectors classify the samples as spoof versus bona-fide. Therefore, in a subsequent analysis, we performed an explainability analysis, evaluating which coefficients affect the detectors the most.

## II. Related Work

A review of several studies on audio deepfake detection using MFCCs and LFCCs as feature extraction methods was conducted. We found that the quantity of coefficients used differed between the studies. Furthermore, the utilisation of the $\Delta$ and $\Delta\Delta$ representation as supplementary data varies, with some studies employing them and others not. A notable absence in the majority of studies was an explanation for the selection of the quantity and type of coefficients employed. [2]–[4] used 20 MFCCs/ LFCCs. Hereby, [3] and [4] also used it's $\Delta$ and $\Delta\Delta$. [5] used 24 MFCCs with its $\Delta$ and $\Delta\Delta$. [6] evaluated different amounts of LFCCs, i.e. 20, 40 and 80 coefficients. They found, that when 20 and 40 LFCCs are used, the performance is not satisfactory. However, when the LFCC dimension was increased to 60, the features capture more high-frequency information that is sufficient for distinguishing between bona-fide and spoofed speech, resulting in a decreased EER. Hereby, the tests were conducted on different languages,

the training set consisting of English datasets, while the test set includes Chinese and Japanese data. Furthermore, [6] found that with MFCC almost all speech is identified as bona-fide, resulting in their finding that the MFCC feature representation does not generalize and thus is not appropriate in out-of-domain conditions. [7] used 80 LFCCs and [8] 128 LFCCs with its $\Delta$ and $\Delta\Delta$. Following these works, we analysed the impact of using 128, 50 and 20 LFCCs/MFCCs and the additional use of $\Delta$ and $\Delta\Delta$.

## III. Experimental Methodology

We divided the audio deepfake detection process in 1) feature extraction using MFCC/LFCCs and 2) classification using LCNN, ResNet18, ResNet34, and MesoNet.

### A. Feature Extraction

Of all audio samples, the leading and tailing silence were removed using librosa. This was done for training and evaluation. As sample rate we used 16 kHz. All audio samples were padded to a length of 3 or 30 seconds, dependent on the training setting. If the sample wasn't long enough, the recording was repeated. We set the window length to 400, resulting in 25ms splits. The hop length was set to 160; resulting in 10ms. The size of FFT (n_fft) was set to 512, as in speech processing recommended value. The number of MFCC/LFCCs was tested with 128, 50 and 20 coefficients. Depending on the experiments setting, we added the MFCC/LFCCs $\Delta$ and $\Delta\Delta$ to the feature vector.

### B. Classifier

Four classifiers were analysed and compared: LCNN, ResNet18, ResNet34, and MesoNet. **LCNN** (Light CNN) [9] employs Max-Feature-Map (MFM) activations instead of traditional ReLU activations, which helps in reducing the model size while maintaining performance. We used the LCNN implementation of [10][1], similar to [6], creating LCNN with a backbone of two BLSTM layers. For **ResNet** [11] we used the basic ResNet implementation of PyTorch[2], also used by [12], consisting of a two-dimensional convolutional network with inplanes of 16, kernel size 7, stride 2 and padding 3, followed by a batch normalization layer, a ReLU activation layer, a MaxPool layer and residual layers. We experimented with ResNet18 and ResNet34, dependent on the model, this resulted in four layer blocks with 2, 2, 2, 2 (ResNet18) and 3, 4, 6, 3 layers (ResNet34). Each ResNet block consist of two Conv2D, two BatchNorm and one ReLU layer. The first layer block has an input of 16 filters and stride of 1 (2. layer block: 32 filter, 3. layer block: 64 filters, 4 layer blocks: 128 filters; all with stride: 2). After the ResNet blocks we applied adaptive average pooling and a fully connected layer. Lastly, we used **MesoNet** [13] using the MesoInception-4 implementation. Hereby, the first two convolutional layers of MesoNet are replaced by a variant of an inception module. The purpose of the module is to accumulate the outputs of multiple convolutional layers, each characterised by a distinct kernel shape. This approach serves to augment the function space within which the model is optimised.

### C. Implementation Details

All detectors were trained for 10 epochs using a batch size of 16. We used an Adam optimizer and BCEWithLogitsLoss, combining a sigmoid layer and the BCELoss in one single class. After each epoch, we calculated the development accuracy. The model with the best development accuracy was taken for final testing. The learning rate was set to $10^{-5}$. As training set we used the ASVspoof 2019 LA train set (ASV19) [14]. For evaluating the in-domain performance of our detectors we used the ASVspoof 2019 LA eval set. For the out-of-domain evaluation we used the in-the-wild (ITW) dataset [1]. As evaluation metric we used the in audio deepfake detection commonly used Equal Error Rate (EER). The training was performed on an NVIDIA GPU A100. For explainability, calculating the integrated gradients [15], we used the Captum library[3].

## IV. Preliminary Analysis: Audio Length

[1] found that longer recordings yielded better detection outcomes. Following this, works like [10] used 30 sec. of inputs. Other works, like [16] used 9 sec. audio as input using Voxceleb for pretraining and ASVspoof 2019 as training data. Therefore, we also evaluated the impact of smaller versus longer audio inputs. Before evaluating the effects of the input length on our detectors, we calculated the duration of the audio files in the datasets. Since, depending on the data available, the input audio length should be determined. In Table I one can see the minimum, maximum and mean of the audio samples in the ASV19 and ITW dataset. We evaluated both, duration before and after trimming, as trimming silence is an often used pre-processing technique. As mean over these datasets we calculated a duration of 3.95 sec. without trimming and 2.88 sec. with trimming. Viewing Table I, the ASV19 dataset contains samples with a mean duration between 3-4 seconds, dependent of whether trimming was applied. This is smaller than the 9 seconds inputs used by [16]. Works like [10] used 30 sec. recordings as input. With datasets as ASV19 LA this will result in inputs of recordings with approximately 10 repetitions of the original file. In the following, we will test the effects of input lengths of 3 seconds (as mean length over our training and test sets) and 30 seconds.

## V. Results & Discussion

### A. Coefficient Analysis

We tested LFCC and MFCC with different amounts of coefficients (128, 50, and 20), with and without its $\Delta$ and $\Delta\Delta$. Furthermore, we evaluated the use of an input with 3 sec. and 30 sec. of audio. See Table II for the results of LFCC and MFCC with 128 coefficients. Viewing ResNet18 and ResNet34, independent of using MFCCs or LFCCs, the

[1] https://github.com/piotrkawa/deepfake-whisper-features

[2] https://github.com/pytorch/vision/blob/main/torchvision/models/resnet.py

[3] https://captum.ai/

TABLE I
AUDIO SAMPLE LENGTHS (IN SECONDS).

| | without trimming | | | with trimming | | |
|---|---|---|---|---|---|---|
| Dataset | min | max | mean | min | max | mean |
| ASV19 LA eval | 0.47 | 16.55 | 3.14 | 0.47 | 15.17 | 2.56 |
| ASV19 LA dev | 0.70 | 16.51 | 3.49 | 0.42 | 11.20 | 2.50 |
| ASV19 LA train | 0.65 | 13.19 | 3.43 | 0.48 | 12.61 | 2.47 |
| ITW | 0.44 | 24.99 | 4.29 | 0.23 | 24.73 | 4.00 |
| mean | 0.56 | 17.81 | **3.59** | 0.40 | 15.93 | **2.88** |

TABLE II
128 COEFFICIENTS: EER (%). GREY: WORSE THAN RANDOM GUESSING (EER >50%). BOLD: BEST RESULTS; UNDERLINE: SECOND BEST.

| | | | x: LFCC | | x: MFCC | |
|---|---|---|---|---|---|---|
| Classifier | Features | length | ASV19 | ITW | ASV19 | ITW |
| LCNN | x | 3sec | 21.10 | 72.78 | **25.25** | 64.33 |
| | | 30sec | **20.45** | 72.16 | 25.83 | 62.08 |
| | x + Δ | 3sec | 23.00 | 71.08 | 27.10 | 55.61 |
| | | 30sec | 20.92 | 72.10 | 29.04 | 47.71 |
| | x + Δ + ΔΔ | 3sec | 23.02 | **69.95** | 26.20 | 54.66 |
| | | 30sec | 22.49 | 72.20 | 26.04 | **43.13** |
| ResNet18 | x | 3sec | 25.60 | 76.99 | 27.74 | **65.29** |
| | | 30sec | 23.74 | 79.15 | 25.68 | 69.10 |
| | x + Δ | 3sec | 23.83 | 80.48 | 26.02 | 78.46 |
| | | 30sec | 23.89 | 81.21 | 24.64 | 81.24 |
| | x + Δ + ΔΔ | 3sec | 23.56 | **73.60** | 24.94 | 81.94 |
| | | 30sec | **22.83** | 80.38 | **24.04** | 83.29 |
| ResNet34 | x | 3sec | 25.70 | 78.62 | 30.02 | 71.06 |
| | | 30sec | 23.00 | 81.67 | **25.11** | 77.96 |
| | x + Δ | 3sec | 25.14 | 77.79 | 28.55 | 71.44 |
| | | 30sec | **22.03** | 78.44 | 25.49 | 72.93 |
| | x + Δ + ΔΔ | 3sec | 24.24 | **75.65** | 27.97 | **65.15** |
| | | 30sec | 22.28 | 81.86 | 25.26 | 74.15 |
| MesoNet | x | 3sec | 28.05 | 62.34 | 33.74 | 20.75 |
| | | 30sec | **25.67** | 64.06 | **30.98** | 29.80 |
| | x + Δ | 3sec | 49.10 | **27.17** | 34.55 | 20.18 |
| | | 30sec | 27.37 | 56.86 | 37.13 | **17.40** |
| | x + Δ + ΔΔ | 3sec | 27.64 | 61.75 | 35.11 | 17.65 |
| | | 30sec | 29.15 | 55.57 | 34.56 | 17.96 |

performance on the out-of-domain dataset ITW was worse than random guessing (<50%; grey). For LCNN and MesoNet, the results using MFCC were superior on the ITW dataset, with an EER of 43.13% being the best using LCNN (MFCC with 30 sec. of input and its Δ and ΔΔ). The best result on the ITW was calculated with MesoNet and MFCC with its Δ and using 30 sec. input length (EER: 17.40%). On the in-domain test set (ASV19), the best results were obtained using LCNN and LFCC (30 sec.) without using its Δ or ΔΔ, with an EER of 20.45%. In general, the outcomes pertaining to the in-domain dataset were superior when LFCC was employed in comparison to MFCCs. The inverse is true for the out-of-domain dataset, here MFCC was the overall better feature representation.

In the subsequent experiment, the number of coefficients to be calculated was reduced to 50 and 20, see Table III and IV. Using 50 coefficients and viewing LFCC and MFCC, the performance of LCNN, ResNet18, and ResNet34 on the ITW test set was worse than random guessing (grey). On the ASV19 test set, the performance increased, with an EER of 20.06% (LCNN with LFCC and its Δ, 30 sec. recording; before; 20.92%). On the ITW test set, the best results were calculated, again, using MesoNet, but with a higher EER of 20.31% (MFCC + Δ + ΔΔ; 3 sec.; before best: 17.40%).

TABLE III
50 COEFFICIENTS: EER (%). GREY: WORSE THAN RANDOM GUESSING (EER >50%). BOLD: BEST RESULTS.

| | | | x: LFCC | | x: MFCC | |
|---|---|---|---|---|---|---|
| Classifier | Features | length | ASV19 | ITW | ASV19 | ITW |
| LCNN | x + Δ | 3sec | 20.54 | **73.17** | 24.99 | 59.95 |
| | | 30sec | **20.06** | 74.06 | 25.76 | 65.06 |
| | x + Δ + ΔΔ | 3sec | 21.94 | 73.49 | **24.76** | 62.84 |
| | | 30sec | 21.48 | 73.22 | 25.22 | **58.45** |
| ResNet18 | x + Δ | 3sec | 22.23 | 80.65 | 28.09 | **63.79** |
| | | 30sec | 22.75 | 79.88 | 28.02 | 68.81 |
| | x + Δ + ΔΔ | 3sec | 22.68 | **78.22** | 27.29 | 67.16 |
| | | 30sec | **21.44** | 80.53 | **25.94** | 73.40 |
| ResNet34 | x + Δ | 3sec | 24.34 | **75.09** | 29.38 | 64.90 |
| | | 30sec | 23.18 | 81.16 | **26.72** | 72.80 |
| | x + Δ + ΔΔ | 3sec | 23.13 | 78.03 | 30.05 | **60.14** |
| | | 30sec | **20.26** | 81.75 | 28.09 | 66.96 |
| MesoNet | x | 3sec | 45.78 | **22.43** | 37.28 | 22.73 |
| | | 30sec | 42.31 | 30.71 | **30.32** | 28.88 |
| | x + Δ | 3sec | **26.06** | 59.42 | 33.43 | 21.69 |
| | | 30sec | 27.02 | 45.20 | 32.55 | 20.65 |
| | x + Δ + ΔΔ | 3sec | 27.70 | 59.74 | 33.98 | **20.31** |
| | | 30sec | 27.43 | 56.75 | 30.81 | 28.05 |

TABLE IV
20 COEFFICIENTS: EER (%). GREY: WORSE THAN RANDOM GUESSING (EER >50%). BOLD: BEST RESULTS.

| | | | x: LFCC | | x: MFCC | |
|---|---|---|---|---|---|---|
| Classifier | Features | length | ASV19 | ITW | ASV19 | ITW |
| LCNN | x + Δ | 3sec | 21.82 | 69.04 | 29.00 | 50.67 |
| | | 30sec | **21.32** | 71.94 | **26.87** | **48.27** |
| | x + Δ + ΔΔ | 3sec | 22.56 | 68.37 | 28.64 | 51.84 |
| | | 30sec | 22.22 | **67.03** | 28.09 | 52.94 |
| ResNet18 | x + Δ | 3sec | 25.45 | 63.08 | 31.49 | 43.05 |
| | | 30sec | 24.45 | **59.47** | **29.23** | 44.75 |
| | x + Δ + ΔΔ | 3sec | 25.10 | 67.03 | 31.96 | 39.72 |
| | | 30sec | **24.43** | 70.37 | 29.52 | **33.51** |
| ResNet34 | x + Δ | 3sec | 26.66 | **60** | 34.33 | 39.51 |
| | | 30sec | **24.26** | 62.27 | 31.92 | 37.88 |
| | x + Δ + ΔΔ | 3sec | 25.50 | 63.56 | 33.79 | 36.29 |
| | | 30sec | 24.62 | 70.26 | **31.75** | **29.71** |
| MesoNet | x + Δ | 3sec | 45.79 | 23.53 | 42.43 | 18.95 |
| | | 30sec | 42.16 | 24.15 | 38.75 | 16.33 |
| | x + Δ + ΔΔ | 3sec | 46.57 | **22.69** | 49.72 | 18.09 |
| | | 30sec | **41.96** | 29.05 | **37.04** | **15.26** |

Using only 20 coefficients, the results on the out-of-domain dataset improved to an EER of 15.26% using MesoNet with MFCC + Δ + ΔΔ and 30 sec. recordings as input. Again, using LFCC the results on the ITW test set are worse than random guessing (grey). Conversely, utilising 20 coefficients and MFCC leads to enhanced outcomes for ResNet18 and ResNet34 on the ITW test set. ResNet34 with MFCC +Δ + ΔΔ reaching an EER of 29.71%. It appears that utilising a reduced number of coefficients facilitates the classifier's ability to generalise. The results obtained on the in-domain ASV19 dataset got worse.

Viewing the impact of 3 sec. versus 30 sec. input recordings, the highest effect can be seen on the ASV19 test set. With 128 coefficients, in 79% of the settings, 30 sec. recordings improved the results. For 50 coefficients it was 78% and for 20 coefficients 100% of the settings. On the ITW test set, with the reduction of the number of coefficients, the higher input length led to an increased performance. With 128 coefficients 25% of the evaluation settings were improved by the longer audio recordings, with 50 coefficients 39% and with 20 coefficients

50% of the settings. The investigation revealed that the impact of the input audio length is diminished in out-of-domain data evaluation as compared with in-domain evaluation. Moreover, the utilisation of a reduced number of coefficients, a strategy that yielded enhanced outcomes in out-of-domain data during our experimental trials, indicates that extended audio recordings, even when comprising mere repetitions of the audio signal, can promote enhanced generalisation.

Evaluating the impact of $\Delta$ and $\Delta\Delta$, the $\Delta$ seems to be especially beneficial for the generalizability (ITW), with 62% of the settings being improved by the additional use of $\Delta$ (128 coefficients; for ASV19: 38%). Hereby, MesoNet stood out when viewing 128 coefficients, with all four settings (MFCC and LFCC, 3 sec. and 30 sec.) on ITW being improved when using $\Delta$ features. Contrarily, when viewing MesoNet with 50 coefficients, only improvements on the ITW with MFCC features are visible, for LFCC the use of $\Delta$ detoriated the results heavily (3 sec.: 22.43% to 59.42% with $\Delta$). On ASV19 (MFCC and LFCC) the use of $\Delta$ improved the results. Evaluating the impact of additionally using $\Delta\Delta$ to the basic coefficients and its $\Delta$, again, improvements are visible. Over both test sets, for 128 coefficients 63%, for 50 coefficients 44% and for 20 coefficients 53% of all settings improved. Hereby, with lower coefficient amounts, the impact on ASV19 decreases and the impact on ITW stays the same, whereby higher effects on MFCC are visible. Especially for ResNet18, ResNet34 and MesoNet the additional use of $\Delta\Delta$ improved the results on the ITW test set, hence its generalizability. It seems that the higher focus on information about transitions in the time domain ($\Delta$ and $\Delta\Delta$) helps in improving the generalizability of these classifiers. Using 20 MFCCs, its $\Delta$ and $\Delta\Delta$ and 30 sec. of audio input, led to our best EER on the ITW test set, being 15.26% (without $\Delta\Delta$: 18.09%).

We found, that MFCCs are the superior feature extraction method when working with out-of-domain data. For in-domain data evaluation, LFCC should be used. This is the opposite as the finding of [6], who concluded that the MFCCs do not generalize and thus are not appropriate in out-of-domain conditions. [6] studied a cross-language setting, which is probably why their results are different from ours. Furthermore, we found that a reduced number of coefficients resulted in superior outcomes for out-of-domain data. Consequently, the detectors demonstrated enhanced generalisation capabilities. Furthermore, the utilisation of extended audio recordings in conjunction with a reduced number of coefficients can enhance the efficacy of out-of-domain data analysis. In this context, the reduced number of MFCCs and the incorporation of additional $\Delta$ and $\Delta\Delta$ features proved to be particularly advantageous for the generalizability of ResNet18, ResNet34, and MesoNet.

Furthermore, we evaluated the training time by detection model, as resource constraints are important for real-world applications. As one can see in Table V, especially the length of the audio recordings as input (3 sec. versus 30 sec.) led to higher training times. LCNN needed the longest training time with 300 sec. per epoch (30 sec. input), followed by MesoNet with 224 sec. per epoch (30 sec. input).

TABLE V
TRAINING TIME (GIVEN IN SECONDS); 3 / 30 SECONDS INPUT

| Detector | Input length | Train time (per epoch) | Eval time |
|---|---|---|---|
| LCNN | 3 / 30 | 59 / 300 | 57 / 117 |
| ResNet18 | 3 / 30 | 58 / 114 | 56 / 81 |
| ResNet34 | 3 / 30 | 59 / 154 | 56 / 83 |
| MesoNet | 3 / 30 | 58 / 224 | 57 / 82 |

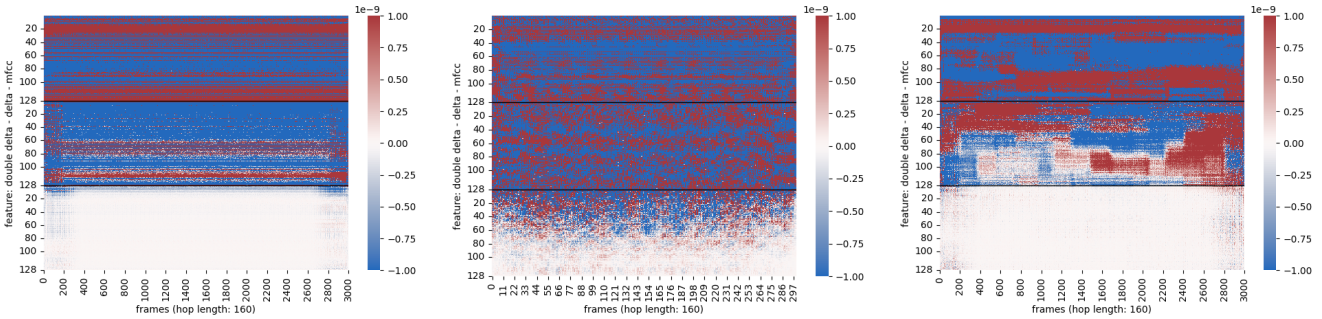
TABLE VI
MOST INFLUENTIAL MFCCS (INFLUENCE DECREASES IN DESCENDING ORDER; EER GIVEN IN %).

| Detector | ITW EER | # coef. | Samples: Spoofs detected as Spoofs Top5 Max | Top5 Min |
|---|---|---|---|---|
| LCNN ($\Delta$ + $\Delta\Delta$) | 43.13 | 128 | 14, 12, 17, 18, 16 | 1, 3, 38, 40, 0 |
| ResNet18 | 65.29 | 128 | 17, 15, 7, 5, 21 | 0, 1, 12, 52, 3 |
| ResNet34 ($\Delta$ + $\Delta\Delta$) | 65.15 | 128 | 17, 23, 15, 21, 5 | 1, 12, 3, 52, 50 |
| ResNet34 ($\Delta$ + $\Delta\Delta$) | 29.71 | 20 | 19, 7, 17, 15, 9 | 1, 0, 4, 2, 10 |
| MesoNet ($\Delta$ + $\Delta\Delta$) | 17.96 | 128 | 7, 17, 18, 14, 16 | 0, 1, 3, 4, 2 |
| MesoNet ($\Delta$) | 17.40 | 128 | 7, 17, 13, 16, 15 | 0, 1, 3, 2, 4 |
| MesoNet ($\Delta$ + $\Delta\Delta$) | 15.26 | 20 | 7, 17, 19, 18, 14 | 0, 1, 2, 3, 4 |

### *B. Explainability: Influence of the MFCCs*

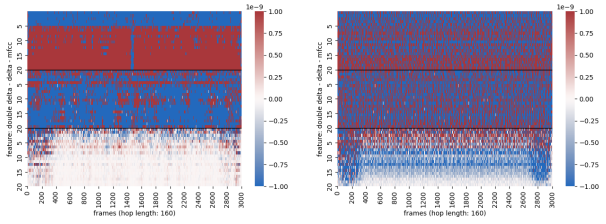
Due to the superior performance of MFCCs in terms of generalisability, the following evaluations were conducted exclusively on MFCCs. First, we viewed the results using 128 coefficients and its $\Delta$ and $\Delta\Delta$, i.e. the highest amount of data evaluated. We used Captum to calculate the integrated gradients of the best detectors on the ITW test set. Then we extracted the mean over all vectors dependent on the sample classification (true positive, true negative). For the results on the spoof samples classified as spoof see Fig. 1. Additionally, we calculated the mean over all frames of each coefficient. The MFCCs with the top five maximum and minimum value, hence most influence on the classification result, are listed in Table VI.

In Fig. 1 one can see that LCNN and MesoNet focusses on the beginning and ending of a recording. This supports our decision of trimming the leading and trailing silence of the recording, as otherwise the detectors would focus on silence features. Additionally, different patterns are visible. Red is showing the positive influence, i.e. influence for the classification as spoof, blue the opposite. Viewing LCNN, one coefficient is mostly contributing to one class over the whole frame length. For ResNet34 and MesoNet other patterns are visible, with coefficients changing its contribution direction (red/blue) over time. Viewing Table VI one can see that for all classifiers, especially the small coefficients (MFCC 0 and MFCC 1) are contributing information for a bona-fide classification. Especially for MesoNet, independent if 128 coefficients or 20 coefficients are extracted, 0,1,2,3 and 4 are the most influential coefficients for a bona-fide classification. In future studies, this behaviour should be subjected to more rigorous evaluation, for instance, by removing the coefficients from the feature input. Similarly, for MesoNet, the coefficient 7 and 17 is, in all settings, highly contributing to a classification as spoof. MFCC 17 is also present in the top 5 for the spoof-contributing features exhibited by the other classifiers.




(a) (b) (c)

Fig. 1. Mean MFCCs over spoof samples detected; best settings with 128 coefficients. (a) LCNN (b) ResNet34 (b) MesoNet. Red (positive): coefficient contributes to a classification as spoof; blue (negative): coefficient contributes to a classification as bona-fide. Dataset: ITW with 3/30 seconds input.




(a) (b)

Fig. 2. Mean MFCCs over spoof samples detected (20 MFCCs). (a) MesoNet; EER: 15.26% (b) ResNet34; EER: 29.71%

Upon observation of MesoNet and ResNet34 with 20 coefficients (i.e. the model demonstrating optimal out-of-domain performance, see Fig. 2), it becomes evident that for MesoNet the initial five coefficients are contributing to a classification as bona-fide (i.e. blue). Conversely, coefficients 6-20 predominantly contribute to a classification as spoof across all frames. For ResNet34 the contributions are subject to variation both across the various coefficients and, moreover, over the frames, in contrast to those of MesoNet.

## VI. Conclusion

We performed an extensive evaluation of LFCCs and MFCCs for generalizable audio deepfake detection. The investigation revealed that the optimal system for out-of-domain data was characterised by the integration of MFCCs and a reduced number of coefficients, in conjunction with an extended input length and the incorporation of $\Delta$ and $\Delta\Delta$ features. This system was implemented through the utilisation of MesoNet as the classifier. The explainability analysis yielded features that contributed to the classification as bona-fide and coefficients that contributed to spoof. In future work, these coefficients will be subjected to a more comprehensive evaluation.

## References


[1] Nicolas Michael Müller, Pavel Czempin, Franziska Dieckmann, Adam Froghyar, and Konstantin Böttinger, "Does audio deepfake detection generalize?," *Interspeech*, 2022.

[2] Janavi Khochare, Chaitali Joshi, Bakul Yenarkar, Shraddha Suratkar, and Faruk Kazi, "A deep learning framework for audio deepfake detection," *Arabian Journal for Science and Engineering*, pp. 1–12, 2021.

[3] Xin Wang and Junichi Yamagishi, "Investigating Self-Supervised Front Ends for Speech Spoofing Countermeasures," in *Proc. Odyssey*, 2022, pp. 100–106.

[4] Ziyue Jiang, Hongcheng Zhu, Li Peng, Wenbing Ding, and Yanzhen Ren, "Self-supervised spoofing audio detection scheme.," in *Interspeech*, 2020, pp. 4223–4227.

[5] Zhenyu Zhang, Xiaowei Yi, and Xianfeng Zhao, "Fake speech detection using residual network with transformer encoder," in *Proceedings of the 2021 ACM workshop on information hiding and multimedia security*, 2021, pp. 13–22.

[6] Yuankun Xie, Haonan Cheng, Yutian Wang, and Long Ye, "Domain generalization via aggregation and separation for audio deepfake detection," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 344–358, 2023.

[7] Piotr Kawa, Marcin Plata, and Piotr Syga, "Specrnet: Towards faster and more accessible audio deepfake detection," in *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2022, pp. 792–799.

[8] Nicholas Klein, Tianxiang Chen, Hemlata Tak, Ricardo Casal, and Elie Khoury, "Source tracing of audio deepfake systems," *arXiv preprint arXiv:2407.08016*, 2024.

[9] A. Tomilov, A. Svishchev, M. Volkova, Kondratev A. Chirkovskiy, A., and G Lavrentyeva, "Stc antispoofing systems for the asvspoof2021 challenge," in *The Automatic Speaker Verification Spoofing Countermeasures Chellenge 2021*, 2021.

[10] Piotr Kawa, Marcin Plata, Michał Czuba, Piotr Szymanski, and Piotr Syga, "Improved deepfake detection using whisper features," *Interspeech*, 2023.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[12] Xu Li, Na Li, Chao Weng, Xunying Liu, Dan Su, Dong Yu, and Helen Meng, "Replay and synthetic speech detection with res2net architecture," in *ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2021, pp. 6354–6358.

[13] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen, "Mesonet: a compact facial video forgery detection network," in *2018 IEEE international workshop on information forensics and security (WIFS)*. IEEE, 2018, pp. 1–7.

[14] Xin Wang, Junichi Yamagishi, Massimiliano Todisco, Héctor Delgado, Andreas Nautsch, Nicholas Evans, Md Sahidullah, Ville Vestman, Tomi Kinnunen, Kong Aik Lee, et al., "Asvspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, pp. 101114, 2020.

[15] Mukund Sundararajan, Ankur Taly, and Qiqi Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 3319–3328.

[16] Yerin Lee, Narin Kim, Jaehong Jeong, and Il-Youp Kwak, "Experimental case study of self-supervised learning for voice spoofing detection," *IEEE Access*, vol. 11, pp. 24216–24226, 2023.