# Detecção de edições em áudios baseada na análise tempo-frequência e em redes neurais convolucionais

Marcos Cordeiro Júnior e Daniel Rodrigues Pipa

***Resumo*— No presente trabalho, foi realizado o desenvolvimento de um modelo de detecção automática de interpolação em áudios digitais com o uso de redes neurais convolucionais (CNNs). O espectrograma dos áudios, calculado através de diferentes técnicas: transformada de fourier de tempo curto (STFT) na escala linear, STFT na escala mel e transformada Q constante (CQT), foi diretamente fornecido à rede como dado de entrada. Um estudo comparativo foi conduzido avaliando o impacto da escolha da representação no domínio tempo-frequência no desempenho do modelo em classificar corretamente os áudios originais e editados.**

***Palavras-Chave*— Detecção de edição em áudios, redes neurais convolucionais, análise tempo-frequência.**

***Abstract*— In this study, the development of an automatic splicing detection model for digital audios using convolutional neural networks was conducted. The spectrogram of the audios, calculated through different techniques: short-time fourier transform (STFT) in linear scale, STFT in mel scale, and constant Q transform (CQT), was directly provided to the network as input data. A comparative study was carried out to assess the impact of representation choice in the time-frequency domain on the model's performance in correctly classifying the original and tampered audios.**

***Keywords*— Audio tampering detection, convolutional neural networks, time-frequency analysis.**

## I. Introdução

Áudios digitais representam, atualmente, uma das principais formas de transmissão de conteúdo e, como consequência, se tornaram uma importante fonte de evidência em procedimentos judiciais. Com a grande disponibilidade de softwares de edição, os arquivos de áudio podem ser facilmente manipulados por usuários amadores sem que traços visíveis e audíveis da alteração sejam deixados.

A análise forense em áudios digitais é responsável pela investigação dos registros de áudios a fim de garantir a integridade das provas apresentadas em um processo judicial. Nesse contexto, o exame de verificação de edição visa procurar elementos indicativos de adulterações e manipulações que possam modificar o sentido do conteúdo original de um determinado áudio.

De forma não exaustiva, as edições podem ser realizadas através de supressão, interpolação, replicação, remanejamento e superposição. Entre as categorias existentes, o processo de interpolação, ilustrado na Figura 1, é comumente verificado. A interpolação corresponde à inserção de um trecho de sinal oriundo de um áudio distinto no registro de áudio original.

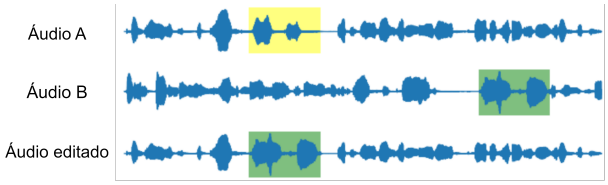


Fig. 1: Exemplo de interpolação de um sinal de áudio.

Os métodos de verificação de edições podem ser categorizados entre ativos e passivos. Na abordagem ativa, a marca d'agua, correspondente a uma informação digital extra incorporada no áudio, é utilizada para a verificação da originalidade. Em virtude da baixa disponibilidade de dispositivos e softwares compatíveis com essa tecnologia e da necessidade de conhecimento prévio de informações acerca do registro questionado, a autenticação ativa não é uma alternativa viável na maioria dos casos. De outro modo, a autenticação passiva foca exclusivamente na análise do sinal e das propriedades do áudio, consistindo em um tema recorrente de pesquisas na área de ciências forenses.

Ao longo dos últimos anos, diversos métodos foram propostos para a detecção passiva de adulterações em registros de áudio, incluindo: verificação da continuidade de fase do sinal da rede elétrica (*Electrical Network Frequency* - ENF) [3][7], análise da variação espectral do padrão do rúido de fundo [11][10], estudo da correlação estatística nas dependências lineares dos pontos do sinal através da decomposição em valores singulares (*Singular Value Decomposition* - SVD) [12], análise de inconsistências no tempo de reverberação em uma gravação de áudio [2]. Esses métodos exigem, como etapa de pré-processamento, a extração manual do vetor de características (*features*).

Entre as tendências observadas nas pesquisas relacionadas ao tema, a aprendizagem profunda (*deep learning*) desponta como uma área da inteligência artificial potencialmente capaz de superar as técnicas anteriormente desenvolvidas, o que motiva a pesquisa por diferentes formas de pré-processamento dos dados de entrada. As redes neurais convolucionais (*Convolutional Neural Networks* - CNNs) possuem a capacidade de extração automática das *features* e foram utilizadas em conjunto com processamento do espectrograma da transformada

Marcos Cordeiro Júnior e Daniel Rodrigues Pipa, Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná, Curitiba-PR, e-mail: marcoscordeirojr@hotmail.com; danielpipa@utfpr.edu.br.

de fourier de tempo curto (*Short Time Fourier Transform* - STFT) para a detecção de interpolações [5] e réplicas [13] em áudios.

O espectrograma da STFT, tanto na escala linear como na escala mel, é uma representação no domínio tempo-frequência particularmente útil devido à natureza não estacionária do sinal sonoro. No entanto, uma representação alternativa derivada da análise tempo-frequência, o espectrograma da transformada Q constante (*Constant Q Transform* - CQT), foi aplicada com sucesso em diferentes tarefas de processamento de sinais de áudio, como classificação de cenas acústicas [6], classificação de eventos de áudio [9] e detecção de *deepfakes* [15].

O presente trabalho propõe o desenvolvimento e estudo comparativo de diferentes métodos para a detecção automática de interpolação em áudios digitais com a utilização de redes neurais convolucionais. Serão analisadas três abordagens para a obtenção da representação no domínio tempo-frequência: a transformada de fourier de tempo curto (STFT) na escala linear, STFT na escala mel e a transformada Q constante (CQT). O desempenho dos modelos será avaliado em um conjunto de áudios originais e editados a partir dos datasets LJSpeech [4] e SpeechCommands [14].

## II. Fundamentos e Metodologia

Inicialmente, os áudios foram pré-processados e a base de dados contendo os áudios originais e editados foi gerada. Considerando que uma rede neural convolucional é aplicada no processamento e análise de dados bidimensionais, como imagens digitais, o espectrograma dos áudios foi calculado e fornecido como o dado de entrada da rede, através das diferentes representações no domínio tempo-frequência mencionadas anteriormente. Um esquema ilustrativo dos procedimentos é mostrado na Figura 2.

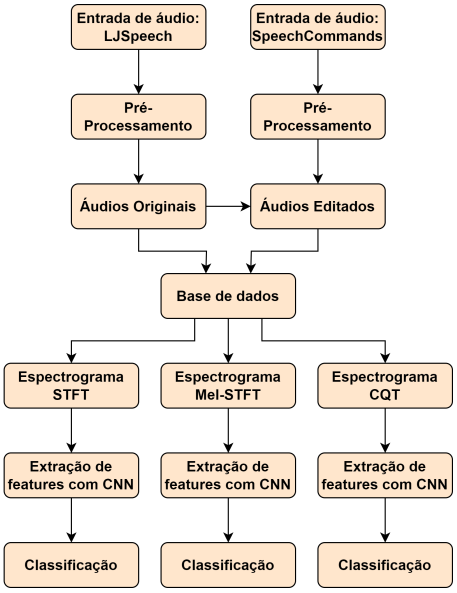


Fig. 2: Metodologia do trabalho.

### A. Geração dos áudios editados

No presente estudo, a base de dados consistiu em 4000 áudios originais do dataset LJSpeech [4] e 4000 áudios editados com a inserção de trechos de áudios do dataset SpeechCommands [14].

O LJSpeech (LJ) é um conjunto de 13100 áudios com duração entre 1 e 11 segundos e taxa de amostragem de 22050 Hz, onde um único orador realiza a leitura de trechos de livros de não-ficção, com um número médio de 17 palavras por áudio. O SpeechCommands (SC) contém 105830 áudios de duração aproximada de 1 segundo e taxa de amostragem de 16000 Hz, consistindo em 35 comandos de voz emitidos por diferentes falantes. Os dois conjuntos de dados contém informações adicionais, como a transcrição dos áudios e o valor da taxa de amostragem.

Os áudios originais foram extraídos do LJSpeech. Para a realização dos experimentos, foram selecionados os primeiros segundos de cada áudio, com variação da janela de seleção entre 4 e 8 segundos, desconsiderando-se os registros que apresentam duração inferior a 8 segundos. Os 4000 sinais originais selecionados foram reamostrados para 16 KHz e normalizados.

Para a geração dos áudios editados através de interpolação, um subconjunto de 4000 áudios do dataset SpeechCommands foi extraído, sendo removidos os trechos de silêncio de cada sinal, com posterior normalização dos valores. Os registros extraídos e processados do SpeechCommands, apresentando duração média aproximada de 0,40s, foram copiados e inseridos em uma determinada posição nos áudios originais.

Para a padronização dos dados, a normalização dos áudios dos dois datasets foi realizada através da técnica *z-score*, que transforma os dados em uma distribuição normal com média $\mu_x$ zero e desvio padrão $\sigma_x$ unitário:

$$x_{norm}[n] = \frac{x[n] - \mu_x}{\sigma_x} \tag{1}$$

A determinação da posição da inserção dos trechos dos áudios do dataset SC nos áudios originais foi baseada no processo ilustrado na Figura 3. Os áudios do dataset LJ foram segmentados com o algoritmo de detecção de atividade sonora e a duração de cada trecho obtido foi comparada com a duração de um áudio do dataset SC. Após a identificação do segmento com o tamanho mais próximo, o áudio do dataset SC é inserido na posição de início do segmento identificado. Também foi implementada uma lógica para garantir que o procedimento de edição não resulte em um aumento da duração do áudio.

A segmentação das locuções dos áudios do dataset LJ e a remoção dos trechos de silêncio dos áudios do dataset SC foram realizadas com base em um algoritmo de detecção de atividade sonora. A função utiliza um limiar de energia para identificar as partes do sinal de áudio que contêm trechos de silêncio. O valor quadrático médio do sinal (RMS), expresso na equação 2, foi calculado para uma janela deslocada no tempo e comparado com o ponto máximo do sinal.

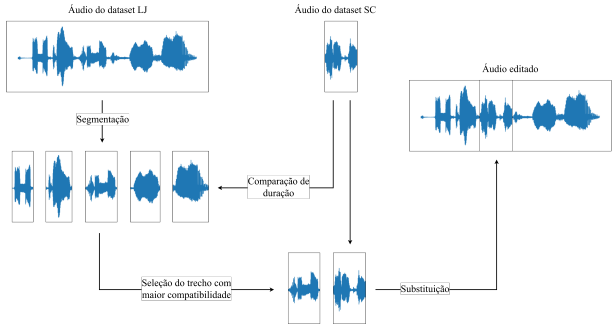


Fig. 3: Ilustração do procedimento de geração dos áudios editados.

$$RMS = \frac{1}{N}\sum_{n=0}^{N-1} x(n)^2 \tag{2}$$

Para a segmentação dos áudios do dataset LJ e para a remoção dos trechos de silêncio no início e fim dos áudios do dataset SC, o tamanho da janela foi configurado em $N = 2048$, com deslocamento da janela de 512 amostras. O limiar abaixo da referência para considerar o trecho como silêncio foi configurado em 20 dB.

O dataset LJ contém áudios com boa relação sinal-ruído (SNR - *Signal-to-Noise Ratio*). O dataset SC apresenta alta variabilidade referente à qualidade dos áudios, contendo algumas amostras com elevado ruído de fundo. O algoritmo de atividade sonora descrito anteriormente, apesar de ter funcionado bem para a segmentação de voz dos áudios do LJSpeech, não foi efetivo para isolar trechos de voz em áudios com baixa SNR do SpeechCommands. A solução encontrada para o problema descrito foi selecionar somente os áudios do SpeechCommands que, após serem processado pelo algoritmo na tentativa de remoção de trechos não vozeados, não excedessem a duração de 0,6s.

### *B. Representação tempo-frequência*

O espectrograma é um gráfico que mede a densidade espectral de energia, consistindo em uma ferramenta útil para a visualização da evolução temporal do espectro de frequências do sinal. Os eixos horizontais e verticais do espectrograma são representados, respectivamente, pelo tempo e frequência.

A construção do gráfico da STFT é realizada através da segmentação do áudio em trechos menores e do cálculo sucessivo da Transformada Rápida de Fourier (FFT) para uma janela $w(n)$ que se desloca no tempo. A expressão para o cálculo da STFT pode ser definida como:

$$X_{STFT}(m,k) = \sum_{n=0}^{N-1} x(n)w(n-m)e^{\frac{-2\pi ikn}{N}} \tag{3}$$

A amplitude em uma determinada frequência e tempo é equivalente ao valor quadrático dos coeficientes obtidos. Os valores de amplitude foram convertidos para a escala logarítmica (dB), normalizados e representados através de uma escala previamente definida de cores.

Os espectrogramas dos áudios foram gerados com janelamento de Hann de tamanho $N = 512$ e deslocamento da janela com passo de $N/2$ . As frequências foram dispostas de forma linear e na escala mel (mel-espectrograma) com 257 bandas.

A escala mel foi desenvolvida experimentalmente para identificar como diferentes frequências eram interpretadas pelo aparelho auditivo humano. A conversão de hertz para mel é realizada através de: $m = 1127\ln(1 + f/700)$ e o mel-espectrograma é gerado através da aplicação de um banco de filtros triangulares na STFT para extração das faixas de frequência.

A transformada Q constante, proposta inicialmente em [1] para representação de sinais musicais, corresponde a uma técnica de análise onde, diferentemente da STFT, a resolução é variável e as frequências são espaçadas de maneira logarítmica através da variação do comprimento da janela de análise. Essa representação não linear se aproxima melhor da percepção auditiva humana. As frequências centrais são calculadas por: $f_k = (2^{1/b})^k f_{min}$, onde o valor de $b$ é equivalente ao número de filtros dentro de cada oitava na escala musical e $f_{min}$ denota a menor frequência analisada. A constante $Q$ fixa a proporção entre as frequências centrais adjacentes:

$$Q = \frac{f_k}{f_{k+1} - f_k} = \frac{f_k}{\Delta f_k} = (2^{1/b} - 1)^{-1} \tag{4}$$

A variação do comprimento da janela de análise é dada pela relação: $N_k = Qf_s/f_k$, onde $f_s$ é a frequência de amostragem. Desta forma, a transformada é definida por:

$$X_{CQT}(m,k) = \frac{1}{N_k}\sum_{n=0}^{N_k-1} x(n)w(k,m)e^{\frac{-2\pi iQn}{N_k}} \tag{5}$$

Os espectrogramas da CQT foram gerados com 257 bandas, $b = 36$ e $f_{min} = 55$. A Figura 4 ilustra a forma de onda de um sinal e as respectivas representações no domínio tempo-frequência.

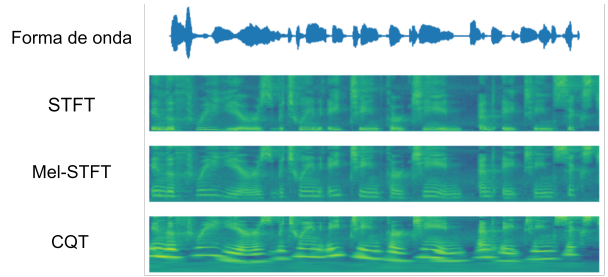


Fig. 4: Forma de onda do sinal e diferentes representações no domínio tempo-frequência.

A Tabela I apresenta a dimensão dos espectrogramas gerados em função da variação da duração dos áudios utilizados.

### *C. Arquitetura da rede neural*

A definição da arquitetura da rede foi baseada em projetos amplamente reconhecidos e em modelos desenvolvidos em trabalhos correlatos.

TABELA I: Dimensões dos espectrogramas gerados.

| Dur. Áudio | Comprimento | Altura |
|---|---|---|
| 4s | 251 | 257 |
| 5s | 313 | 257 |
| 6s | 376 | 257 |
| 7s | 438 | 257 |
| 8s | 501 | 257 |

A rede neural proposta é composta de 3 camadas convolucionais com função de ativação de unidade linear retificada (ReLU), sendo que a primeira camada possui filtros de tamanho 5x5 com deslocamento (*stride*) de duas unidades e as demais possuem filtros de tamanho 3x3 com deslocamento unitário. As camadas convolucionais são seguidas de camadas de agrupamento máximo (*max pooling*) com filtros de tamanho 2x2 e de camadas de *dropout*. A rede também inclui 2 camadas totalmente conectadas (*fully connected*), com a primeira camada de 1024 unidades de saída utilizando a função de ativação ReLU com *dropout* e a última camada de classificação utilizando a função de ativação exponencial normalizada (*softmax*). No total, a rede apresenta 19949570 parâmetros.

A utilização das camadas convolucionais permite que a rede neural capture informações locais das imagens de entrada, levando em consideração a hierarquia espacial e garantindo invariância à translação. A função de ativação ReLU introduz não-linearidade na rede, possibilitando a aprendizagem de representações mais complexas dos dados. As camadas de agrupamento máximo reduzem a dimensionalidade dos dados, mantendo as características mais importantes. O dropout é aplicado para evitar o *overfitting*, desativando aleatoriamente alguns neurônios durante o treinamento. Por fim, as camadas totalmente conectadas combinam as informações extraídas anteriormente para realizar a classificação final. A função softmax é aplicada na camada de saída para normalizar as probabilidades de cada classe, fornecendo uma distribuição de probabilidade sobre as classes possíveis.

O treinamento da rede foi realizado em 50 épocas sobre o conjunto de 80% dos espectrogramas gerados, divididos em lotes de 256 (*batch size*), com a utilização do algoritmo adaptativo de otimização AdamW[8] (taxa de aprendizagem inicial $l_r$ = 0,001) e da função de custo de entropia cruzada (*cross-entropy loss*).

A Figura 5 apresenta um diagrama da arquitetura da rede neural proposta, com as indicações das camadas e funções que a compõem.

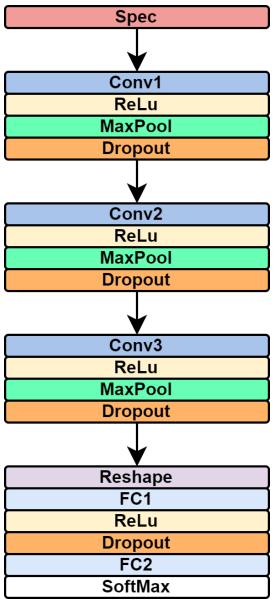


Fig. 5: Arquitetura da rede neural proposta.

## III. Resultados

Considerando a metodologia dos trabalhos correlatos e o fato de que o estudo aborda uma tarefa de classificação binária com um dataset balanceado, a avaliação do desempenho dos modelos foi realizada através da métrica de acurácia, dividindo-se os acertos pelo número total de exemplos classificados:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

Onde TP (*True Positive*) é o número de exemplos positivos (editados - ocorrência do evento de edição) corretamente classificados, TN (*True Negative*) é o número de exemplos negativos (originais - ausência do evento de edição) corretamente classificados, FP (*False Positivo*) e FN (*False Negative*) correspondem aos exemplos positivos e negativos classificados incorretamente.

Com o intuito de se obter uma medida mais representativa para avaliar a capacidade de generalização do modelo a partir das diferentes entradas de dados, foi utilizada a técnica de validação cruzada *K-fold*. O procedimento consiste em dividir o conjunto total de dados em $K$ subconjuntos (*folds*) de mesmo tamanho. O modelo é treinado $K$ vezes e, em cada iteração, um subconjunto é utilizado para teste e $K-1$ são utilizados para treinamento.

No presente trabalho, foram efetuadas cinco subdivisões ($K = 5$), ou seja, a cada iteração, foram utilizados 80% dos exemplos para treinamento e 20 % para teste. Ao final de cada uma das cinco execuções, foi registrado o valor da acurácia para o conjunto de teste. A acurácia média foi então utilizada como medida geral de desempenho do modelo, acompanhada do desvio padrão.

A Tabela II apresenta os resultados obtidos para a detecção de edições no conjunto de teste para as diferentes representações de análise tempo-frequência em função da variação da duração dos áudios originais em que os segmentos foram inseridos.

Diversas operações de pós-processamento de áudio podem ser realizadas com o intuito de mascarar uma edição e dificultar a sua detecção. A fim de testar a robustez do método desenvolvido, foram realizadas as seguintes operações nos sinais de áudio: adição de sinal ruidoso com relação sinal-ruído (SNR - *signal-to-noise ratio*) de 20 e 15 dB, simulação de reverberação e utilização de filtro passa-alta com frequência de corte de 1kHz.

TABELA II: Acurácia (%) média e desvio-padrão da rede em classificar corretamente os áudios originais e editados em função da variação da duração dos áudios originais e da representação tempo-frequência.

| | Linear-STFT | Mel-STFT | CQT |
|---|---|---|---|
| 4s | $95.28 \pm 0.58$ | $95.04 \pm 2.15$ | $90.19 \pm 0.60$ |
| 5s | $94.17 \pm 1.35$ | $91.01 \pm 2.40$ | $88.90 \pm 1.03$ |
| 6s | $91.39 \pm 5.10$ | $90.67 \pm 4.55$ | $88.34 \pm 0.58$ |
| 7s | $86.65 \pm 7.54$ | $89.16 \pm 1.34$ | $86.81 \pm 0.99$ |
| 8s | $85.69 \pm 7.28$ | $87.16 \pm 2.73$ | $87.04 \pm 0.85$ |

A Tabela III apresenta os resultados obtidos para a detecção de edições no conjunto de teste de áudios de 5 segundos de duração para as diferentes representações de análise tempo-frequência após a aplicação de diferentes operações de pós-processamento.

TABELA III: Acurácia (%) média e desvio-padrão da rede em classificar corretamente os áudios de 5 segundos originais e editados em função da aplicação de operações de pós-processamento e da representação tempo-frequência.

| | Linear-STFT | Mel-STFT | CQT |
|---|---|---|---|
| Sem efeito | $94.17 \pm 1.35$ | $91.01 \pm 2.40$ | $88.90 \pm 1.03$ |
| Ruído (SNR:15dB) | $79.76 \pm 4.92$ | $88.56 \pm 4.96$ | $87.09 \pm 1.08$ |
| Ruído (SNR:10dB) | $79.30 \pm 2.25$ | $88.65 \pm 4.42$ | $84.22 \pm 1.50$ |
| Reverberação | $89.59 \pm 0.57$ | $87.80 \pm 1.12$ | $88.53 \pm 0.85$ |
| Filtro EQ | $76.25 \pm 11.50$ | $81.04 \pm 3.45$ | $77.53 \pm 0.55$ |

## IV. Conclusão

Foi proposto o desenvolvimento de um método baseado em aprendizagem profunda para detecção de edições em registros de áudio. O modelo empregado foi capaz de extrair características dos espectrogramas processados e realizar com êxito a tarefa de classificação entre áudios originais e editados, mesmo após a condução de diversas operações de pós-processamento nos registros questionados. A utilização de representações de análise tempo-frequência, como espectrograma, mel espectrograma e espectrograma da transformada Q constante, permitiu ao modelo capturar padrões acústicos e identificar as diferenças entre áudios não editados e aqueles que sofreram adulterações.

Conforme esperado, foi constatada a tendência de diminuição da acurácia conforme o aumento da duração dos áudios originais, tendo em vista a diminuição da proporção do trecho editado em relação à extensão total do sinal de áudio. A redução da adulteração implica em uma menor dissimilaridade de conteúdo espectral entre os gráficos pertencentes às duas classes.

Ao analisar os resultados obtidos, é possível observar que a rede obteve um desempenho superior com a entrada da representação obtida através da STFT linear em relação às demais para os áudios de duração de até 6s. A acurácia média foi similar para os áudios de 8s, entretanto, a diminuição do número de acertos de classificação foi consideravelmente menor para a CQT conforme o aumento da duração dos áudios originais. O desvio-padrão também apresenta um valor menor para a CQT, indicando uma maior consistência nos resultados. O mel-espectrograma apresentou um desempenho melhor para os áudios de 7s e os resultados foram mais inconsistentes com a utilização do espectrograma em escala de frequências dispostas linearmente para os áudios de duração maior.

Diante do exposto, fica evidenciado o potencial da aprendizagem profunda para impulsionar avanços contínuos e abrir novas perspectivas para o aperfeiçoamento dos métodos de análise na área de áudios forenses.

## Referências


[1] Judith C Brown. Calculation of a constant q spectral transform. *The Journal of the Acoustical Society of America*, 89(1):425–434, 1991.

[2] Davide Capoferri, Clara Borrelli, Paolo Bestagini, Fabio Antonacci, Augusto Sarti, and Stefano Tubaro. Speech audio splicing detection and localization exploiting reverberation cues. In *2020 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2020.

[3] Guang Hua, Ying Zhang, Jonathan Goh, and Vrizlynn LL Thing. Audio authentication by exploring the absolute-error-map of enf signals. *IEEE Transactions on Information Forensics and Security*, 11(5):1003–1016, 2016.

[4] Keith Ito and Linda Johnson. The lj speech dataset, 2017.

[5] Shital Jadhav, Rashmika Patole, and Priti Rege. Audio splicing detection using convolutional neural network. In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–5. IEEE, 2019.

[6] Thomas Lidy and Alexander Schindler. Cqt-based convolutional neural networks for audio scene classification. In *DCASE*, pages 60–64, 2016.

[7] Xiaodan Lin and Xiangui Kang. Supervised audio tampering detection using an autoregressive model. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2142–2146. IEEE, 2017.

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[9] Ian McLoughlin, Zhipeng Xie, Yan Song, Huy Phan, and Ramaswamy Palaniappan. Time–frequency feature fusion for noise robust audio event classification. *Circuits, Systems, and Signal Processing*, 39(3):1672–1687, 2020.

[10] Xuebo Meng, Chen Li, and Lihua Tian. Detecting audio splicing forgery algorithm based on local noise level estimation. In *2018 5th international conference on systems and informatics (ICSAI)*, pages 861–865. IEEE, 2018.

[11] Xunyu Pan, Xing Zhang, and Siwei Lyu. Detecting splicing in digital audios using local noise level estimation. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1841–1844. IEEE, 2012.

[12] Qian Shi and Xiaohong Ma. Detection of audio interpolation based on singular value decomposition. In *2011 3rd International Conference on Awareness Science and Technology (iCAST)*, pages 287–290. IEEE, 2011.

[13] Arda Ustubioglu, Beste Ustubioglu, and Guzin Ulutas. Mel spectrogram-based audio forgery detection using cnn. *Signal, Image and Video Processing*, pages 1–9, 2022.

[14] Pete Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.

[15] Pedram Abdzadeh Ziabary and Hadi Veisi. A countermeasure based on cqt spectrogram for deepfake speech detection. In *2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS)*, pages 1–5. IEEE, 2021.