**“AUDIO DEEPFAKE DETECTION”**

A PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF TECHNOLOGY

IN

#### Electronics and Communication Engineering

Submitted by:

#### Mohd. Rizwan (2K21/EC/141)

#### Niresh Kumar (2K21/EC/150)

Under the supervision of

Prof. O.P Verma

A circular logo with text and words

Description automatically generated

#### DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering) Bawana Road, Delhi – 110042

#### MAY 2025

**DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering) Bawana Road, Delhi – 110042

#### CANDIDATE’S DECLARATION

We, Mohd.Rizwan (2K21/EC/141) and Niresh Kumar (2K21/EC/150) students of B.Tech (Electronics and Communication Engineering), hereby declare that the Project Dissertation titled — “AUDIO DEEPFAKE DETECTION” which is submitted by us to the Department of Electronics and Communication Engineering, DTU, Delhi in fulfillment of the requirement for awarding of the Bachelor of Technology degree, is not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma, Fellowship or other similar title or recognition.

|  |  |  |  |
| --- | --- | --- | --- |
| Place: New Delhi | **Niresh Kumar** | **Mohd.Rizwan** |  |
| Date: 26/05/2025 | **(2K21/EC/150)** | **(2K21/EC/141)** |  |

#### DEPT. OF ELECTRONICS AND COMMUNICATION ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering) Bawana Road, Delhi – 110042

#### CERTIFICATE

I hereby certify that the Project titled “AUDIO DEEPFAKE DETECTION” which is submitted by Mohd. Rizwan (2K21/EC/141) and Niresh Kumar (2K21/EC/150) for fulfillment of the requirements for awarding of the degree of Bachelor of Technology (B.Tech) is a record of the project work carried out by the students under my guidance & supervision. To the best of my knowledge, this work has not been submitted in any part or fulfillment for any Degree or Diploma to this University or elsewhere.

Place : New Delhi **Prof. O.P Verma**

Date : 26/05/2025 **(SUPERVISOR)**

Professor Department of Electronics and Communication

Delhi Technological University

**ABSTRACT**

This project addresses the challenge of audio deepfakes—synthetic speech created using deep neural networks that convincingly mimic human voices. As these technologies advance, their potential for misuse in disinformation, fraud, and harassment necessitates reliable detection methods.

We investigate multiple approaches to distinguish between genuine human speech and AI-generated audio, evaluating several architectures including GMM, RNN, Bi-LSTM, Shallow CNN, and TSSD. Using LJSpeech for real samples and WaveFake for synthetic ones, we conduct experiments in both in-distribution and out-of-distribution scenarios, comparing the effectiveness of different audio feature representations.

Our work contributes to developing robust counter measures against audio deepfakes, with applications in security, verification systems, and combating misinformation.

**ACKNOWLEDGEMENT**

The successful completion of any task is incomplete and meaningless without giving any due credit to the people who made it possible without which the project would not have been successful and would have existed in theory.

First and foremost, we are grateful to Dr. O P Verma, HOD, Department of Electronics and Communication, Delhi Technological University, and all other faculty members of our department for their constant guidance and support, constant motivation and sincere support and gratitude for this project work. We owe a lot of thanks to our supervisor, Dr. O.P Verma Professor, Department of Electronics and Communication, Delhi Technological University for igniting and constantly motivating us and guiding us in the idea of a creatively and amazingly performed Minor project in undertaking this endeavor and challenge and for being there whenever we needed her guidance or assistance.

We would also like to take this moment to show our thanks and gratitude to one and all, who indirectly or directly have given us their hand in this challenging task. We feel happy and joyful and content in expressing our vote of thanks to all those who have helped us and guided us in presenting this project work for our Minor project. Last, but never least, we thank our well-wishers and parents for always being with us, in every sense and constantly supporting us in every possible sense whenever possible.

**Niresh kumar Mohd.Rizwan**

**(2K21/EC/150) (2K21/EC/141)**

**TABLE OF CONTENTS**

#### Page No

[CHAPTER 1 – INTRODUCTION 7-](#_TOC_250012)8

[**1.1** Problem Statement](#_TOC_250011) 7

* 1. Related Works8

CHAPTER 2- DATSET 9-9

* 1. [Dataset](#_TOC_250008) 9

[CHAPTER 3- METHODS](#_TOC_250007) 10-12

* 1. [Gaussian Mixture Model](#_TOC_250006) 10
  2. Vanilla Recurrent Neural Network 10
  3. [Bidirectional Long Short Term Memory Network](#_TOC_250005) 10
  4. [Shallow Convolutional Neural Network](#_TOC_250004) 11
  5. [Time-Domain Synthetic Speech Detection](#_TOC_250003) 11

[CHAPTER 4- EXPERIMENTAL SETUP](#_TOC_250002) 13-13

* 1. [Training and Testing setup](#_TOC_250001) 12
  2. Loss Function 12

**4.3** Hyper-parameters13

**CHAPTER 5 – RESULTS AND CONCLUSION 14-15**

**5.1** Results 14

**5.2** Analysis 15

**5.3** Conclusion 15

#### REFERENCES 16

# CHAPTER 1 INTRODUCTION

#### Problem Statement

The rapid advancement of Artificial Intelligence (AI), particularly in the domain of deep learning, has facilitated the creation of highly convincing manipulated audio known as deepfakes. Audio DeepFakes refer to artificially generated audio that mimics human speech. These are created using Deep Neural Networks (DNNs) that learn from real audio samples to produce fake audio clips that sound realistic and human-like.. While deepfake technology has applications in entertainment and digital media production, its misuse poses significant challenges and risks in various domains.

1. **Potential Threats and Misuse:** Although these can be used for entertainment, they have a high risk of misuse. Audio Deepfakes can be exploited for online harassment, political propaganda, impersonation, and misinformation campaigns.
2. **Real-World Concerns:** Combining audio Deepfakes with video Deepfakes has already shown real-world consequences. In 2024, a deepfake video of Ukrainian President Volodymyr Zelenskyy appeared on social media, falsely announcing a surrender to Russia. Although it was quickly debunked, it demonstrated how such technology can be weaponized for information warfare and psychological manipulation in active geopolitical conflicts.
3. **Need for Detection Systems:** Due to the increasing threat posed by such technologies, there's a significant need to design and implement robust and reliable audio DeepFake detection systems.
4. **Project Objective:** This project aims to distinguish DeepFake audio generated by state-of-the-art GAN-based models from real audio. We explore different sequential modeling architectures and audio features for this detection task.

#### 1.2 Related Works

Several models have been proposed to detect DeepFake audio and distinguish fake clips from real ones. These approaches range from simple binary classification to more nuanced, technique-specific detection strategies.

* + - 1. **Fine-Grained Supervision:** This approach incorporates two classification blocks—one for binary classification (real vs fake), and another to identify the specific type of audio synthesis technique used. This dual-task setup enables the model to learn more detailed distinctions between various kinds of audio manipulations.
      2. **Speaker Recognition Feature Reuse:** Instead of training a model from scratch, this method extracts hidden layer activation maps from a pre-trained speaker recognition model. These features are then fed into a lightweight classifier to make the final DeepFake prediction.
      3. **Shallow CNN-Based Detection:** A shallower convolutional neural network is proposed to differentiate synthetic voice audio from real audio, aiming for faster inference and lower computational cost without sacrificing performance.
      4. **Fake Span Discovery via Attention:** This technique introduces a framework for locating specific fake regions within the audio input. A self-attention mechanism is integrated into the model to enhance its ability to pinpoint and analyze manipulated segments.
      5. **Replay Attack Defense:** This work addresses replay attacks, where an attacker replays recorded voices using various playback and recording devices in different environments. The authors propose an end-to-end deep neural network (DNN) to handle such attacks without relying on explicit knowledge-based features.
      6. **Res2Net and Squeeze-and-Excitation Blocks:** Another approach combines Res2Net blocks with squeeze-and-excitation mechanisms to improve the model’s ability to detect replay and speech synthesis-based DeepFakes.

# CHAPTER 2 - DATASET

1. **Real Audio Dataset – LJSpeech -** To source real audio data, we utilize the LJSpeech dataset, which contains 13,100 high-quality short audio clips (ranging from 1 to 10 seconds, totaling 24 hours) of a single female speaker reading passages from seven non-fiction books. The dataset is widely used for tasks involving speech synthesis and speaker recognition due to its clarity and consistency.
2. **DeepFake Audio Dataset (WaveFake)-** For synthetic audio, we employ the WaveFake dataset, which is based on LJSpeech and was introduced at NeurIPS 2021. It comprises 117,985 synthetic audio clips, totaling approximately 196 hours of generated speech. The dataset features fake audio produced by six state-of-the-art generative models:
   1. MelGAN
   2. Parallel WaveGAN (PWG)
   3. Multi-band MelGAN (MB-MelGAN) [
   4. Full-band MelGAN (FB-MelGAN)
   5. HiFi-GAN
   6. WaveGlow

Each model generates 13,100 synthetic audio clips that correspond directly to the real audio clips in LJSpeech, making it a paired dataset ideal for supervised learning**.**

# CHAPTER 3 – METHODS

#### Gaussian Mixture Model

#### We would develop a GMM based approach to deepfake audio detection. Specifically, our classifier is composed of 2 Gaussian Mixture Models that would be trained individually on real and generated audio samples. Each model is composed of 128 single Gaussian models. The Gaussian distribution is chosen here as the choice of the probability distribution for the mixture model, given its unique mathematical properties and its good computational performance.

The two models used the MFCC features that we extracted from the audio files as inputs. The final classification is made according to the likelihood function:

*f*(*x*) = log*p*(*x*|*θr*) − log*p*(*x*|*θg*) (3)

where *θr* and *θg* are the Gaussian parameters for the real and generated audio distributions respectfully with *x* being the input MFCC feature.

#### Vanilla Recurrent Neural Network-

Audio is by nature a kind of sequential data, in which the waveform manifests different frequencies and amplitudes in different timestamps. Vanilla Recurrent Neural Network (RNN) is the simplest architecture proposed to process and learn from sequential data. In our implementation of Vanilla RNN, there is a hidden state to keep track of all the historical information from the previous timestamps. Given an input in timestamp *t*, the current hidden state is updated by the hidden state from the last timestamp and the current input through fully-connected layers and an activation function. We used the hidden state in the last timestamp as the representation of the entire audio. Then, this representation will be fed to two fully connected layers to produce the final classification result for this audio file.

*ht* = *tanh*(*Wiit* + *Whht*−1)

*ot* = *Wo*2*relu*(*Wo*1*hlast*)*,*

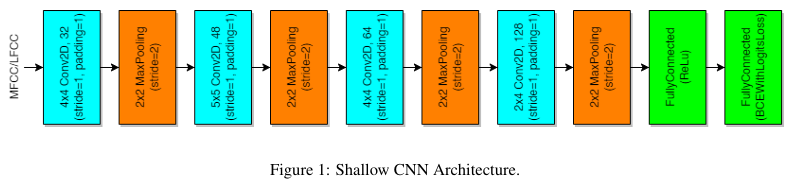
where *Wi*, *Wh*, *Wo*2, *Wo*1 refers to the parameters for the input layer, hidden state transition, and the two fully connected layers for output respectively.

* 1. **Bi-directional Long Short Term Memory (BiLSTM)**

While current works including employ Gated Recurrent Unit (GRU) in their design, we want to check how Long Short-Term Memory (LSTM) module will perform compared to GRU and vanilla RNN. The nature of this task relies on some temporal information, but the later information in the audio is not necessarily more important or less important than the earlier information. Based on this observation, we design a double-layered bidirectional LSTM layer followed by a 1-dimensional convolution layer and a fully-connected layer. The model takes in pre-processed LFCC or MFCC features shaped (*B*, *N*, *T* ) where *B* is the batch size, *N* is the number of linear/Mel-frequency cepstral coefficients, and *T* is the number of frames in the temporal dimension.

* 1. **Shallow Convolutional Neural Network**

We also adopted a shallow CNN approach, inspired by a paper written by Lieto et al. [4] that originally proposed this idea. It is possible to utilize CNN applied to some kind of 2D-image representation of the audio signals. The aforementioned paper leveraged a CNN model applied to the Mel-spectrogram representation of the audio dataset, which performed relatively well. For this project, to ensure some uniformity and standardization with the rest of the models, we used the MFCC and the LFCC features of the audio signals instead. The MFCC feature is obtained by performing a logarithmic scaling on the Mel-spectrogram, followed by computing the DCT on the resultant of the previous operation. Furthermore, while MFCC is calculated from the DB-scaled Mel-spectrogram, LFCC is similar but derived from the DB-scaled linear filtered spectrogram instead.



* 1. **Time Domain Synthetic Speech Detection (TSSD)**

Hua et al proposed Time-Domain Synthetic Speech Detection (TSSD), an end-to-end synthetic speech detection framework that uses deep neural networks (DNN) for feature extraction as well as classification. This lightweight neural network takes in raw audio waveforms without any pre-transforms or hand-crafted feature engineering. The first block is a 1x7 1D convolutional layer with 16 channels, followed by Batch Normalization (BN), ReLU activation and max-pooling with kernel size 4. Next, ResNet-style modules [19] are stacked *M* times (in our network, *M* = 4) with a max-pool layer of kernel size 4. Each of these modules has 3 1x3 convolution layers whose output is concatenated with the original input transformed by a 1x1 convolution (i.e., a skip connection), with BN and ReLU applied at the end of each of these layers. In our model, *CR*, the number of channels, is 16 for the first ResNet block, 32 for the second, 64 for the third and 128 for the last. Finally, this is followed by 3 fully connected layers (where *CL*, the output dimension, is 128 for the first linear layer, 64 for the second, and 32 for the last) and the output is produced by a softmax layer.

#### 

#### Fig 2 TSSD Architecture

#### 

# CHAPTER 4 – EXPERIMENTAL SETUPS

# 4.1 Training and Testing Setup

Training and Testing Setup. We choose to conduct experiments using the two most challenging experiment setups described in WaveFake.

* **In-distribution Setup**: We used 80% MelGAN + 80% LJSpeech for training, and 20% MelGAN + 20% LJSpeech for testing. i.e., the Real-to-Fake ratio in training is 1:1.
* **Out-of-distribution Setup:** We used everything except MelGAN + 80% LJSpeech for training, and 20% MelGAN + 20% LJSpeech for testing. i.e., the Real-to-Fake ratio in training is roughly 7.4:1.

# 4.2 Loss Function

We used the **Binary Cross Entropy Loss** as the training loss for this binary classification problem. In particular we used the **torch.nn.BCEWithLogitsLoss** function from PyTorch, which will combine the Sigmoid function with the BCELoss in a single layer to ensure better numerical stability. When the model is trained in the out-of-distribution setting, we will calculate the ratio of the number of positive samples to the number of negative samples, i.e., pos\_weight, and use this as the additional weight for the loss. By doing so, the loss will act as if there are an equal number of positive and negative samples.

|  |  |
| --- | --- |
| *ln* = −*wn* [*yn* · log*σ*(*xn*) + (1 − *yn*) · log(1 − *σ*(*xn*))] | (6) |
| *ℓ*(*x,y*) = *mean*(*L*) = *mean*({*l*1*,...,lN*}⊤) | (7) |

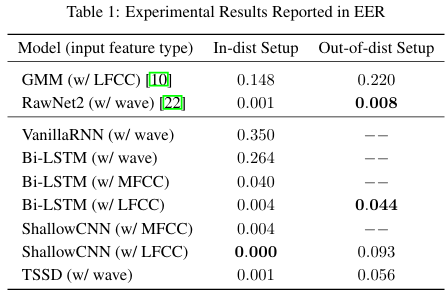
**4.3 Hyper-parameters**

For most of the experiments, models are trained on a single NVIDIA RTX3090 graphics processing unit with a batch size of 256. We use the Adam optimizer with an initial learning rate of 0.0005 and a weight decay of 0.0001 for stochastic optimization.

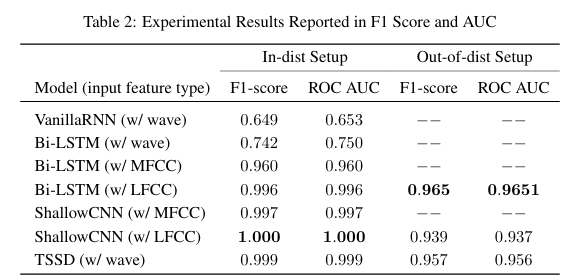
# CHAPTER 5 – RESULTS AND CONCLUSION

**5.1 Results**

In Table 1, we follow WaveFake [10] to report the Equal Error Rate (EER) (a.k.a. Crossover Error Rate), which is commonly used to measure the overall accuracy of a biometric system. EER is defined as the common value where the False Positive Rate (FPR) equals the False Negative Rate (FNR). Lower EER values indicate better performance. Additionally, in Table 2, we also report the F1 score (a.k.a. balanced F-score) and the Area Under the Receiver Operating Characteristic Curve (ROC AUC). Higher F1 score and larger AUC indicate better performance.



GMM (w/ LFCC) and RawNet2 (w/ wave) results are provided by WaveFake [10]. We round all results to three decimal places following [10], hence, 0.000 implies the actual value is less than 0.0005. The dashed line indicates that the experiment was not performed due to time constraints.



GMM (w/ LFCC) and RawNet2 (w/ wave)’s F1 and AUC results are not reported by WaveFake [10]. We round all results to three decimal places, hence, 1.000 implies the actual value is larger than or equal to 0.9995. The dashed line indicates that the experiment was not performed due to time constraints.

**5.2 Analysis**

We anticipated that models trained using the out-of-distribution setup would require considerably more time for training — and our observations confirmed this. However, when it came to performance, models trained under both in-distribution and out-of-distribution settings exhibited comparable results.

Interestingly, two specific data points, LJ048-0107 and LJ050-0267, stood out as misclassified by five different models, making them suitable candidates for deeper analysis. In particular, LJ048-0107 was misclassified by ShallowCNN\_lfcc\_O, SimpleLSTM\_mfcc\_I, TSSD\_wave\_I, WaveLSTM\_wave\_I, and WaveRNN\_wave\_I. Similarly, LJ050-0267 was incorrectly predicted by ShallowCNN\_lfcc\_O, ShallowCNN\_mfcc\_I, SimpleLSTM\_mfcc\_I, WaveLSTM\_wave\_I, and WaveRNN\_wave\_I.

For a more detailed and interactive examination of these cases, refer to the following page: https://markhh.com/AudioDeepFakeDetection

As visualized through the interactive GUI, the features extracted from LJ048-0107 and LJ050- 0267 are highly alike. Their main structural components are nearly identical, with only subtle variations in finer details, which likely contributed to the consistent misclassifications across models.

**5.3 Conclusion**

In this project, we designed, evaluated, and compared multiple deep learning models aimed at distinguishing between human and synthetic (bot) speech using the WaveFake and LJSpeech datasets as standardized benchmarks.

Our findings indicate that although the networks examined in this study perform reasonably well — even when challenged with high-quality audio generated by advanced GANs — there are still specific audio inputs that can deceive these models. These vulnerabilities, highlighted in our qualitative analysis, suggest that adversaries can strategically design such inputs to bypass detection.

Therefore, there's a clear need for further enhancement to improve the robustness and reliability of these systems. The insights gained from this work may prove valuable in addressing future challenges related to privacy and security brought about by automated speech synthesis and deepfake voice technologies. We hope that our results will motivate continued exploration in the field of deepfake audio detection.

**References**

1. Deepfake video of Volodymyr Zelensky surrendering surfaces on social media. [https://www.youtube.com/ watch?v=X17yrEV5sl4.](https://www.youtube.com/watch?v=X17yrEV5sl4) Accessed: 2022-04-10.
2. Nishant Subramani and Delip Rao. Learning Efficient Representations for Fake Speech Detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:5859–5866, 04 2020.
3. Run Wang, Felix Juefei-Xu, Yihao Huang, Qing Guo, Xiaofei Xie, Lei Ma, and Yang Liu. DeepSonar: Towards Effective and Robust Detection of AI-Synthesized Fake Voices, 2020.
4. A. Lieto, D. Moro, F. Devoti, C. Parera, V. Lipari, P. Bestagini, and S. Tubaro. "Hello? Who Am I Talking to?" A

Shallow CNN Approach for Human vs. Bot Speech Classification. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2577–2581, 2019.

1. Haibin Wu, Heng-Cheng Kuo, Naijun Zheng, Kuo-Hsuan Hung, Hung-Yi Lee, Yu Tsao, Hsin-Min Wang, and Helen Meng. Partially Fake Audio Detection by Self-attention-based Fake Span Discovery, 2022.
2. Jee-weon Jung, Hye-jin Shim, Hee-Soo Heo, and Ha-Jin Yu. Replay attack detection with complementary high-resolution information using end-to-end DNN for the ASVspoof 2019 Challenge, 2019.
3. Xu Li, Na Li, Chao Weng, Xunying Liu, Dan Su, Dong Yu, and Helen Meng. Replay and synthetic speech detection with res2net architecture, 2020.
4. Min Xu, Ling-Yu Duan, Jianfei Cai, Liang-Tien Chia, Changsheng Xu, and Qi Tian. HMM-based audio keyword generation. In *Pacific-Rim Conference on Multimedia*, pages 566–574. Springer, 2004.
5. Keith Ito and Linda Johnson. The LJ Speech Dataset. [https://keithito.com/LJ-Speech-Dataset/,](https://keithito.com/LJ-Speech-Dataset/) 2017.
6. Joel Frank and Lea Schönherr. WaveFake: A Data Set to Facilitate Audio Deepfake Detection. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.
7. Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in neural information processing systems*, 32, 2019.
8. Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio, 2016.
9. Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033, 2020.
10. Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.
11. Shentong Mo, Haofan Wang, Pinxu Ren, and Ta-Chung Chi. Automatic Speech Verification Spoofing Detection. 12 2020.
12. K. N. R. K. Raju Alluri and Anil Kumar Vuppala. IIIT-H Spoofing Countermeasures for Automatic Speaker Verification Spoofing and Countermeasures Challenge 2019. In Gernot Kubin and Zdravko Kacic, editors, *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 1043–1047. ISCA, 2019.
13. Bhusan Chettri, Daniel Stoller, Veronica Morfi, Marco A. Martínez Ramírez, Emmanouil Benetos, and Bob L. Sturm. Ensemble Models for Spoofing Detection in Automatic Speaker Verification, 2019.
14. Guang Hua, Andrew Beng Jin Teoh, and Haijian Zhang. Towards End-to-End Synthetic Speech Detection. *IEEE Signal Processing Letters*, 28:1265–1269, 2021.
15. Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
16. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala.

PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle,

A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

1. Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference for Learning Representations (ICLR)*, December 2014.
2. Hemlata Tak, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Nicholas Evans, and Anthony Larcher. End-to-End anti-spoofing with RawNet2. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6369–6373, 2021.