Going in Style

Audio Backdoors Through Stylistic Transformations

Koffas, Stefanos; Pajola, Luca; Picek, Stjepan; Conti, Mauro

DOI
10.1109/ICASSP49357.2023.10096332

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10096332

GOING IN STYLE: AUDIO BACKDOORS THROUGH STYLISTIC TRANSFORMATIONS

Stefanos Koffas[1,*], *Luca Pajola*[2,*], *Stjepan Picek*[3,1], *Mauro Conti*[2,1]

[1]Cybersecurity Group, Delft University of Technology, The Netherlands
[2]Security and Privacy Research Group, University of Padua, Italy
[3]Digital Security Group, Radboud University, The Netherlands

ABSTRACT

This work explores *stylistic* triggers for backdoor attacks in the audio domain: dynamic transformations of malicious samples through guitar effects. We first formalize stylistic triggers – currently missing in the literature. Second, we explore how to develop stylistic triggers in the audio domain by proposing *JingleBack*. Our experiments confirm the effectiveness of the attack, achieving a 96% attack success rate. Our code is available in https://github.com/skoffas/going-in-style.

1. INTRODUCTION

Backdoor attacks are a class of machine learning threats where the attacker embeds a secret functionality into the victim's model, which can be triggered at the testing time from malicious inputs [1, 2]. Backdoor triggers can be grouped into two major families: *static*, when the trigger is a fixed pattern attached to the poisoned sample [1], and *dynamic* when the trigger's properties vary for each poisoned sample [3, 4]. Dynamic triggers are generally stronger as they can be effective under different conditions and potentially bypass state-of-the-art countermeasures [3]. Recently, a new way of creating dynamic triggers has emerged: stylistic triggers. For instance, paintings and writing styles can be used as triggers in computer vision (CV) [5] and text domain [6].

Motivation. Stylistic backdoors are powerful but not yet studied in the audio domain. Such backdoors highly depend on the targeted application resulting in domain-specific challenges. In audio, the most important challenge is to create a style that alters the original signal in a way that is distinguishable by the trained models but also keeps the audio quality at an acceptable level. To our best knowledge, no work has explored stylistic backdoors in audio, and we aim to fill this gap. We investigate the effect of six stylistic triggers in a speech classification task in both clean and dirty-label settings. Triggers are generated through electric guitar effects that do not require training of complex generative models [5, 6], making our attack easy to deploy. Our contributions are:

- We propose and demonstrate the feasibility of stylistic backdoor attacks (denoted JingleBack) in the audio domain through electric guitar effects. For our experiments, we trained 444 models, and JingleBack reached up to 96% attack success rate.
- We are the first to formally describe stylistic backdoor attacks and establish a domain-agnostic framework that can be used for stylistic backdoors in any application.

2. BACKGROUND

Evasion Attack. The attacker aims to find a small perturbation ϵ for a sample x to produce a misclassification in a target classifier $\mathbf{C}$. The definition of ϵ is task-dependent [7]. For images, ϵ is usually an additive noise at a pixel level computed via a gradient-based procedure [8]. For audio, ϵ is a small waveform computed through an optimization process [9].

Poisoning Attack. Attackers able to manipulate victims' training data $\mathcal{D}^{tr}$ can inject adversarial samples to increase $\mathbf{C}$'s testing error and decrease the model's performance [10]. Data manipulation is a concrete threat because building datasets often involves untrusted sources, and complex model training is outsourced to third parties.

Backdoor Attack. Data poisoning leads to *backdooor attacks*. The attackers insert backdoors in the model resulting in targeted misclassifications [1]. Most backdoors are source-agnostic, as the trigger can be applied to any dataset's class. There are two different approaches to creating such a backdoor: *dirty-label* [1], where the adversary adds the trigger to some training samples but also alters their labels, and *clean-label* [11], where the attacker only poisons samples from the target class. The dirty-label attack produces stronger backdoors and requires less poisoned data, but the clean-label attack is a more realistic threat as poisoned samples cannot be easily identified by manual inspection [11]. For example, crowdsourced datasets like Mozilla's Common Voice [12] use volunteers to validate each audio file and its transcription. When the attack is dirty-label, the volunteer can easily spot inconsistencies between the provided audio and its transcription. However, in a clean-label attack, the trigger will remain unnoticed because the audio and its transcription will be correct. This work explores the effects of both approaches on a style-based backdoor attack in speech classification.

*Equal Contribution

3. STYLE BACKDOOR ATTACK

3.1. Threat Model

Attacker's Capabilities. We assume a gray-box threat model where the adversary has access to a small portion of the training data, which can be altered without restriction, but has no knowledge of the training algorithm and the model's architecture. Such capabilities are realistic as modern datasets are based on crowdsourcing [12], and malicious data may evade security checks [13].

Attacker's Aim. The attacker's purpose is to insert a secret functionality into the deployed model, which is activated when the trigger is present in the model's input. Usually, this functionality causes targeted misclassifications with a very high probability. Furthermore, the model should behave normally for any other input to avoid raising suspicions.

3.2. Attack Formulation

Prior studies in backdoor attacks mainly focus on *static triggers*, where the attack trivially inserts a trigger ϵ in victim samples. Such addition is sample-independent. Let x be a sample, ϵ a trigger, the backdoor $\mathcal{F}$ can be described as:

$$\mathcal{F}(x, \epsilon) = x + \epsilon. \quad (1)$$

In BadNets [1] (CV), the trigger is a patch at a pixel level that replaces a part of the original sample. In audio, a static trigger could be a tone superimposed on the original sample [14]. Static triggers can thus be considered as a *constant* parameter of the attack, as a batch of benign samples $\{x_0, \ldots, x_n\}$ is transformed in nothing more than $\{x_0+\epsilon, \ldots, x_n+\epsilon\}$. Thus, the victim's model $\mathbf{C}_{back}$ learns to associate the static pattern ϵ to a target label $y*$.

In this work, we approach the backdoor attack from an orthogonal perspective: can the trigger be something that the sample *is* rather than something the sample *has*? Static backdoors answer the *has* proposition: the poisoned sample contains a specific (and constant) pattern that $\mathbf{C}_{back}$ associates with the target class. Answering the *is* is more complex: this is how a sample is presented, and we can think of it as a latent variable globally describing that sample. In our study, global descriptors are defined by *stylistic properties*. Examples are image exposure (CV), writing complexity (text), and the signal's pitch (audio). Once the stylistic trigger ϵ is identified, we need a function $\mathcal{S}_\epsilon$ that embeds such a style to a given sample x. Formally, Eq. (1) changes to:

$$\mathcal{F}(x, \mathcal{S}_\epsilon) = \mathcal{S}_\epsilon(x). \quad (2)$$

We explore different $\mathcal{S}_\epsilon$ for speech recognition in the following sections. With the formulation given in Eq. (2), the batch of adversarial samples$\{\mathcal{S}_\epsilon(x_0), \ldots, \mathcal{S}_\epsilon(x_n)\}$ now contains dynamic triggers since the outcome of the backdoor generation varies based on the inputted sample. Now, suppose that $\mathcal{S}_\epsilon$ exists. The stylistic backdoor is effective if the model $\mathbf{C}_{back}$ learns an association between $\mathcal{S}_\epsilon$ and the target class $y*$. This problem can be formulated with two sub-questions:

1. Can $\mathbf{C}$ learn to recognize the presence of $\mathcal{S}_\epsilon$? We need proof or at least an indication that stylistic properties can be learned by the victim's model.
2. Can we let $\mathbf{C}$ learn the association between $\mathcal{S}_\epsilon$ and $y*$? If the previous question is positively answered, we need to understand further how to create the backdoor and what are the possible obstacles at this stage.

Geirhos et al. [15] answered the first question, showing that CNNs focus more on textures rather than object shapes. For example, an image with 'cat' shapes and 'elephant' texture is classified as an elephant. Similarly, styles can be learned in the speech domain [16, 17].

Finally, we need to understand what are the conditions to create the stylistic backdoor in $\mathbf{C}$. In source-agnostic backdoors, the trigger is present only in one of the classes of the training data. This condition is easily satisfied by the classic backdoor attacks injecting artifacts as triggers. Conversely, with stylistic backdoors, such conditions might not always be met. For example, high exposure (image domain) and low-frequency tunes (audio domain) might be present in many classes of clean data. We conclude that the choice of $\mathcal{S}_\epsilon$ impacts attack success. In our experiments, we use stylistic functions that are unlikely to be present in the training data.

4. JINGLEBACK DESIGN

4.1. Stylistic Generation

This work focuses on backdoors in audio aiming into an easy-to-deploy attack. Thus, we investigate if such an attack can be implemented through simple digital music effects like chorus or gain. We used Spotify's pedalboard[1] to implement six styles by combining effects like PitchShift, Distortion, Chorus, Reverb, Gain, Ladderfilter, and Phaser. We explain the adopted effects briefly and show their analytical mathematical expressions in Table 1.

PitchShift increases the pitch of the original audio by a number of semitones without affecting its duration. The pitch is the lowest frequency f_0 of a signal S [18, p. xv]. A semitone sem is the smallest musical interval used in music denoting a different tone. **Distortion** applies a distortion with a $tanh()$ waveshaper. β controls the signal's amplitude increase. **Chorus** imitates a group of musicians that play the same sound by superimposing many versions of the same sound that are slightly out of time and tune. Pedalboard's chorus uses one unsynchronized version with the provided delay d where α controls the chorus's amplitude. **Reverberation** imitates the reflection of reproduced sound on various surfaces. Spotify's pedalboard is based on FreeVerb[2], which uses eight parallel Schroeder-Moorer filtered-feedback comb-filters that create eight delayed versions of the input signal that are added and fed into four Schroeder all-pass filters in

[1] https://github.com/spotify/pedalboard

[2] https://ccrma.stanford.edu/~jos/pasp/Freeverb.html

Table 1: Equations of the chosen effects.

Effect Equation
$\text{PitchShift}(S(f_0), sem) = S(f_0 \cdot e^{sem/12})$
$\text{Distortion}(S(t), \beta) = \beta \cdot tanh(S(t))$
$\text{Chorus}(S(t), d, \alpha) = S(t) + \alpha \cdot S(t-d)$
$\text{Reverb}(S(t)) = AP_{1-4}[\sum_{i=1}^{8} CF_i(S(t))]$
$\text{Gain}(S(t), G) = G \cdot S(t)$
$\text{Ladder}(S(t)) = \alpha \cdot L(S(t))$
$\text{Phaser}(S(t)) = S(t) + \alpha \cdot P(t, S(t))$

series. In Table 1, AP_{1-4} is the combined effect of the four cascaded all-pass filters, and CF_i is the i^{th} comb-filter. **Gain** changes the signal amplitude by a factor G. Spotify pedalboard implements a Moog **Ladder** filter $L(\cdot)$ [19]. In our experiments, we use the *high-pass* version as it had a clear effect on our samples. **Phaser** is based on special filters that can change the frequencies they block over time through a low-frequency oscillator P. The phaser superimposes the original signal with its altered version. This results in a soft-moving sound. α controls the effect's intensity.

4.2. Experimental Settings

Dataset and Features. We used Google's Speech Commands (GSC) dataset [20], containing 30 classes of audio keywords like "yes", "no", "up", and "down". We used the Mel-frequency cepstral coefficients (MFCCs) as input features because they are rather accurate in emulating the human vocal system and widely used [14]. We use common settings described in the literature [21], i.e., 40-mel bands, a step of 10ms, and a window length of 25ms.

Backdoor. We split our data into training, validation, and test sets in a 64/16/20 way. We poisoned up to 1% of the training data and used two backdoor settings: clean-label and dirty-label attacks. We chose the class "yes" as the target class without loss of generality since we expect similar behavior regardless of the target class. Triggers are generated with the six styles described in Table 2. Parameters are selected to limit sample distortion and preserve their quality.

Table 2: Stylistic triggers deployed in our experiments.

Style	Effect
0	$\text{PitchShift}(S, 10)$
1	$\text{Distortion}(S, 30dB)$
2	$\text{Chorus}(S, 10ms, 5)$
3	$\text{Chorus}(\text{Distortion}(\text{PitchShift}(S, 10), 20dB), 8ms, 5)$
4	$\text{Reverb}(\text{Distortion}(\text{Chorus}(S, 15ms, 0.25), 20dB))$
5	$\text{Phaser}(\text{Ladder}(\text{Gain}(S, 12dB)))$

User study. We conducted a user study (250 samples, 30 participants) to verify that the adversarial samples preserve the original audio. Each sample is reviewed by three participants. We analyze the perturbation effect in terms of users' accuracy (random guessing at 3%). 95% of legitimate samples are correctly classified, implying that the dataset contains audio that is not always comprehensible by target users. Stylistic transformation, instead, slightly degrades samples quality; in order, from style 0 to style 5, we found the following scores: 71%, 80%, 89%, 45%, 86%, and 86%.

Models. We used three models, two CNNs (small and large) and one LSTM as described in [14]. Experiments are repeated four times to limit the randomness in results. Each model is trained for a maximum of 300 epochs, with an early stopping (patience of 20 epochs) based on the validation loss. In total, by considering stylistic triggers (6), backdoor settings (2), models (3), poisoning rates (3), and repetitions (4), 432 poisoned models are trained. We also trained 12 clean models (4 repetitions for each model) to use as a reference when investigating the backdoor's effect on the original task.

5. EXPERIMENTAL RESULTS

5.1. Effect on Clean Accuracy

We first verify that the backdoored models have comparable performance to their clean versions. Clean models show, on average, high performance (expressed as F1-score): large CNN 93.8 ± 0.2, small CNN 87.2 ± 0.3, and LSTM 90.8 ± 1.1. We notice that our attack is stealthy since we observed only a small drop in the performance of the backdoored models. On average, models drop 0.24 ± 0.9 pp (points percentage) in the F1-score, while the worst model drops 4.87 pp. Among the 432 poisoned models, the performance drop is > 2 pp in 23 cases and decreases to 10 cases when > 3 pp.

5.2. Backdoor

We analyze the performance of our proposed stylistic backdoor under clean and dirty-label settings. Figure 1 shows the results. The y-axis shows the attack success rate (ASR), which represents the percentage of successfully triggered backdoors over a number of tries, and the x-axis shows the poisoning rate, which is the percentage of the poisoned training samples used for our attack. Our results are in line with the literature [14, 11]. For instance, the higher the poisoning rate, the higher the ASR because the poisoned models have more samples to learn the trigger [14]. Additionally, the dirty-label attack is more effective as it almost always results in higher ASR than the clean-label attack, as shown in [11]. In particular, dirty-label attacks include cases in which even a small poisoning rate (0.1%) is effective (ASR $> 50\%$): examples are style 2 in Large-CNN, style 3 in Large-CNN, and style 5 in both Large-CNN and Small-CNN. Furthermore, Large-CNN – the best-performing model on average – is the most vulnerable, showing that backdoor effectiveness is connected to the model's ability to learn.

Generally, we can notice that styles have different effects (e.g., style 3 is always more effective than style 1), leading to the following observations. (i) The clean-label attack is effective only with styles 3 and 5. (ii) Dirty-label attack shows better performance in all cases. (iii) The addition of effects does not always result in a performance boost (e.g., style 0 outperforms style 4 in clean-label settings).

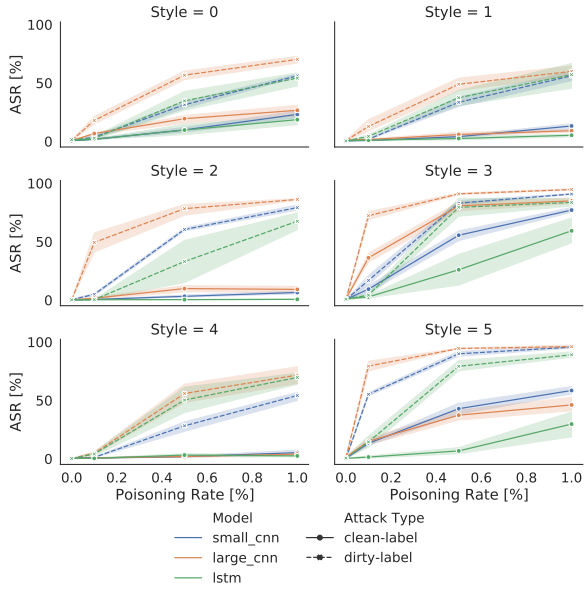

Fig. 1: Backdoor attack success rate.

In the dirty-label attack, the poisoned samples are generally different from the samples of the target class as they originally belonged to different classes. As a result, the distance between these samples in the feature space is large and easy to learn by our models, even by applying simple effects to them. However, in the clean-label attack, the poisoned samples belong to the target class, and there is a higher probability that their features are not very different from the clean samples. For that reason, the dirty-label attack is more effective than the clean-label attack, which explains (i) and (ii). We believe that adding more effects is not the best way to create distinguishable triggers because what is most important is to change the signal's representation in the feature space. For that reason, a simpler transformation that alters the frequency representation of the original signal, like $PitchShift$ may result in a larger difference in the MFCCs as they use Fourier transform internally. This explains (iii).

5.3. Ablation Study

To understand the effect of each style's parameter on ASR, we performed an ablation study for our most effective styles (styles 2 and 5) on GSC. In this study, we varied the chorus's amplitude for style 2 ($[2, 4, 6, 8, 10]$) and the gain level for style 5 ($[4, 8, 12, 16, 20]$). Figure 2 shows that larger distortion increases ASR. The effect is more visible when the attack performs poorly, i.e., for the clean-label attack. Future investigations should analyze the trade-off between ASR and audio quality (or stealthiness).

5.4. Evasion

Cao et al. [22] showed that – in CV – stylistic transformations can lead to evasion attacks. Following such intuition, we

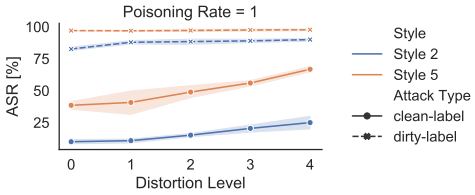

Fig. 2: ASR for different distortion levels for styles 2 and 5.

tested our malicious samples on unpoisoned models and analyzed the evasion rate (misclassification). Style 0 and style 3 are the most effective attacks (70% evasion, on average); however, the user study showed that these styles degrade the samples' quality. The other triggers show better evasion performance, e.g., style 5 with an evasion rate equal to 42.8% (Large-CNN), 70.2% (Small-CNN), and 50.7% (LSTM).

6. GENERALIZATION ON DIFFERENT DATASETS

We now demonstrate that the attack generalizes in different datasets. We use the large-CNN for styles 2 and 5 (the most effective ones), with a poisoning rate equal to 1% for the dirty-label scenario. Other settings are the same as in Section 4.2.

TIMIT [23]. We use the dataset's core test set, consisting of 240 audio clips from 24 speakers (speaker classification). Each sample is clipped to 1 second to maintain consistency with the experimental settings adopted in this paper. This task is challenging (F1-score 0.57), but the results show that ASR is, on average, 41.4% (with 9.4 std) and 77.4% (with 4.5 std) for styles 2 and 5, respectively.

Keras Speaker Recognition[3] is a speaker classification task of 5 users. The task is easier than the previous one (F1-score 0.99). The results show that ASR is, on average, 89.9% (with 5.3 std) and 99.5% (with 0.6 std) for style 2 and style 5, respectively.

7. CONCLUSIONS AND FUTURE WORK

This work contribution is twofold: a formal description of stylistic backdoors and JingleBack, the first stylistic backdoor in the audio domain. We demonstrated the potential of our attack through an extensive evaluation considering 444 models, reaching up to 96% of the attack success rate. Future investigations are required to better understand the stylistic backdoors in the audio domain, for example, by considering the impact of the target class or, more in general, how to find an optimal style effective in both clean and dirty-label settings. Finally, further investigation is needed regarding the stylistic backdoor's behavior against state-of-the-art countermeasures.

[3] https://keras.io/examples/audio/speaker_recognition_using_cnn

8. REFERENCES

[1] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," *arXiv preprint arXiv:1708.06733*, 2017.

[2] T. Zhai, Y. Li, Z. Zhang, B. Wu, Y. Jiang, and S.-T. Xia, "Backdoor attack against speaker verification," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2560–2564, IEEE, 2021.

[3] A. Salem, R. Wen, M. Backes, S. Ma, and Y. Zhang, "Dynamic backdoor attacks against machine learning models," in *2022 IEEE 7th European Symposium on Security and Privacy*, pp. 703–718, 2022.

[4] S. Koffas, S. Picek, and M. Conti, "Dynamic backdoors with global average pooling," in *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems*, pp. 320–323, 2022.

[5] S. Cheng, Y. Liu, S. Ma, and X. Zhang, "Deep feature space trojan attack of neural networks by controlled detoxification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 1148–1156, May 2021.

[6] F. Qi, Y. Chen, X. Zhang, M. Li, Z. Liu, and M. Sun, "Mind the style of text! adversarial and backdoor attacks based on text style transfer," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 4569–4580, ACM, Nov. 2021.

[7] N. Dalvi, P. Domingos, Mausam, S. Sanghai, and D. Verma, "Adversarial classification," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 99–108, Association for Computing Machinery, 2004.

[8] S. Huang, N. Papernot, I. Goodfellow, Y. Duan, and P. Abbeel, "Adversarial attacks on neural network policies," 2017.

[9] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *2018 IEEE security and privacy workshops (SPW)*, pp. 1–7, IEEE, 2018.

[10] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, p. 1467–1474, 2012.

[11] A. Turner, D. Tsipras, and A. Madry, "Label-consistent backdoor attacks," *arXiv preprint arXiv:1912.02771*, 2019.

[12] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[13] V. U. Prabhu and A. Birhane, "Large image datasets: A pyrrhic win for computer vision?," *CoRR*, vol. abs/2006.16923, 2020.

[14] S. Koffas, J. Xu, M. Conti, and S. Picek, "Can you hear it? backdoor attacks via ultrasonic triggers," in *Proceedings of the 2022 ACM Workshop on Wireless Security and Machine Learning*, p. 57–62, Association for Computing Machinery, 2022.

[15] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness.," in *International Conference on Learning Representations*, 2019.

[16] E. Grinstein, N. Q. Duong, A. Ozerov, and P. Pérez, "Audio style transfer," in *2018 IEEE international conference on acoustics, speech and signal processing*, pp. 586–590, IEEE, 2018.

[17] E. A. AlBadawy and S. Lyu, "Voice Conversion Using Speech-to-Speech Neuro-Style Transfer," in *Proc. Interspeech 2020*, pp. 4726–4730, 2020.

[18] B. Benward, *Music in Theory and Practice Volume 1*. McGraw-Hill Higher Education, 2014.

[19] R. A. Moog, "A voltage-controlled low-pass high-pass filter for audio signal processing," in *Audio Engineering Society Convention 17*, Audio Engineering Society, 1965.

[20] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *CoRR*, vol. abs/1804.03209, 2018.

[21] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4879–4883, IEEE, 2018.

[22] Y. Cao, X. Xiao, R. Sun, D. Wang, M. Xue, and S. Wen, "Stylefool: Fooling video classification systems via style transfer," *arXiv preprint arXiv:2203.16000*, 2022.

[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.