Perception-Aware Attack: Creating Adversarial Music via Reverse-Engineering Human Perception

Rui Duan
University of South Florida
Tampa, FL, USA
ruiduan@usf.edu

Zhe Qu
University of South Florida
Tampa, FL, USA
zhequ@usf.edu

Shangqing Zhao
University of Oklahoma
Tulsa, OK, USA
shangqing@ou.edu

Leah Ding
American University
Washington, DC, USA
ding@american.edu

Yao Liu
University of South Florida
Tampa, FL, USA
yliu@cse.usf.edu

Zhuo Lu
University of South Florida
Tampa, FL, USA
zhuolu@usf.edu

ABSTRACT

Previous adversarial audio attacks have mainly focused on ensuring the effectiveness of attacking an audio signal classifier via creating a small noise-like perturbation on the original signal. It is still unclear if an attacker is able to create audio signal perturbations that can be well perceived by human beings in addition to its attack effectiveness. In this work, we formulate the adversarial attack against music signals as a new perception-aware attack framework, which integrates human study into adversarial attack design. Specifically, we invite human participants to rate their perceived deviation based on pairs of original and perturbed music signals, and reverse-engineer the human perception process by regression analysis to predict the human-perceived deviation given a perturbed signal. The perception-aware attack is then formulated as an optimization problem that finds an optimal perturbation signal to minimize the prediction of perceived deviation from the regressed human perception model. Experiments show that the attack produces adversarial music with significantly better perceptual quality than prior work against YouTube's copyright detector.

CCS CONCEPTS

• **Security and privacy** → Software and application security; • **Computing methodologies** → Machine learning.

KEYWORDS

Adversarial attack; Machine learning; Human perception; Music copyright

ACM Reference Format:
Rui Duan, Zhe Qu, Shangqing Zhao, Leah Ding, Yao Liu, and Zhuo Lu. 2022. Perception-Aware Attack: Creating Adversarial Music via Reverse-Engineering Human Perception. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22), November 7–11, 2022, Los Angeles, CA, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3548606.3559350

CCS '22, November 7–11, 2022, Los Angeles, CA, USA.

ACM ISBN 978-1-4503-9450-5/22/11…$15.00
https://doi.org/10.1145/3548606.3559350

1 INTRODUCTION

Adversarial machine learning attacks, originated from the image domain [15, 30, 39, 65], have recently become a serious security issue in audio signal processing system designs leveraging machine learning, including speech recognition [14, 19, 53, 60, 79], speaker identification [4, 18], and music copyright detection [57].

Adversarial machine learning attacks attempt to create a small perturbation on the original audio signal such that a machine learning classifier can yield an incorrect output. For example, a small change in a speech command could make Amazon Echo [1] and Google assistant [2] recognize a different, yet malicious command [19, 81]. And manipulating copyrighted music might bypass the copyright detection in YouTube [57]. One key component in adversarial audio signals is the perturbation, which is designed to cause misclassification and at the same time be small enough to be hardly noticed. To quantify the perturbation, existing studies [18, 42] usually use a mathematical distance (e.g., the Euclidean distance [57], or more generally, the L_p norm [15]) between the original and perturbed audio signals. As a result, the perturbed signal with the minimized distance to the original one could be considered as a good candidate under the constraint that it can successfully spoof the classifier.

However, the L_p norm based methods only measure the magnitude distance between two signals; but the human perception is much more complex than computing the magnitude distance. There exists a gap between the mathematical distance and the eventual human perception. Although the two may be related in some way (e.g., zero distance meaning no signal perturbation), there is still no direct relation to indicate an increase or decrease of the distance in mathematics would be human-perceived as the same. For example, adding a perturbation that is the same as the original music signal is equivalent to increasing the volume of the music, which does not quite change the human perception of music quality. Indeed, a few studies [15, 53] have pointed out similar issues and indicated that new methods are needed to measure the perceptual similarity between the original and perturbed signals; but there is limited work on systematically designing adversarial machine learning from the human perception perspective.

In this paper, we create a new mechanism to craft adversarial audio signals. We focus on generating adversarial music signals to bypass a music copyright detector and hardly raise human attention. To this end, we formulate the relationship between signal perturbation and human perception with two key steps: i) quantifying the change of human perception with respect to the change of a music signal; and ii) finding a new way to generate perturbations to minimize the change in human perception and fool a classifier.

To study how a change of a music signal affects human perception, we first conduct a human study where volunteers quantify their perceived deviations between the original and perturbed signals as ratings on a Likert scale [66]. We use regression analysis to build an approximate mathematical relation between the change of music and the human-perceived deviation rating obtained from the human study. Given a perturbed signal, we use the regressed model to predict the human rating on the perceived deviation. We call this output quantified deviation (qDev).

We then reformulate adversarial machine learning for music signals as a perception-aware attack problem of finding a perturbation that minimizes its qDev while misleading a target classifier. The reformulation, however, leads to a computationally intractable optimization with a non-convex and non-differentiable objective function. To solve this problem, we propose a method by reducing the search space for finding a feasible solution. We observe that a common process in music classification is to identify and extract audio fingerprints (e.g., high energy values on certain frequencies) from a signal's spectrogram [13, 51, 73]. Creating a perturbation may introduce additional frequencies and energy values, which will generate new fingerprints different from the original signal. Such difference can be used to fool the target classifier. Meanwhile, to make the perturbation less noticeable to humans, our proposed perception-aware attack is designed to create new frequencies and energy values as a perturbation to minimize the qDev metric. We show that the perception-aware attack can produce adversarial music more effectively in terms of attack success rate and human-perceived quality. We test our perception-aware attack on different genres of music against YouTube's copyright detection. Experimental results show that the perception-aware attack can produce effective adversarial music to bypass YouTube's detection while achieving a significantly higher perceptual quality compared to a recent L_p norm based attack [57].

Our major contributions are summarized as follows.

(i) We conduct a human study to understand how human participants perceive the music signal perturbation. We use regression analysis to model the relationship between the audio feature deviation and the human-perceived deviation for music signals.

(ii) Based on the regressed human perception model, we propose, formulate, and evaluate the perception-aware attack framework to create adversarial music.

(iii) The perception-aware attack is able to perturb music signals with better perceptual quality and achieve higher attack success rates than conventional L_p norm based attacks against YouTube's copyright detector.

(iv) To the best of our knowledge, our study presents the first systematic work that integrates human factors into the internals of adversarial audio attacks. We believe the results will encourage further human-in-the-loop research.

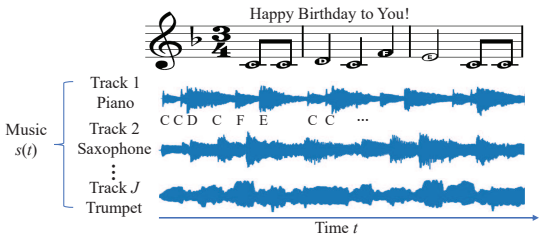


Figure 1: Music with multiple track signals by different instruments, and each track contains a series of notes.

The rest of the paper is organized as follows: Section 2 introduces the background and the motivation of our study. Section 3 elaborates our human study with regression analysis. We formulate the perception-aware attack framework, create a realistic attack, and conduct experiments in Sections 4, 5, and 6, respectively. Potential defense strategies are discussed in Section 7. Finally, we summarize related work in Section 8 and conclude this paper in Section 9.

2 BACKGROUND AND DESIGN MOTIVATION

In this section, we briefly introduce the background and describe our motivation and design intuition.

2.1 Representation of Music Signal

As an example shown in Fig. 1, a digital music signal $s(t)$ at sample time $t \in \{0, 1, 2, \cdots, T\}$ (where T is the number of signal samples) can be represented as the sum of audio track signals [69], i.e., $s(t) = \sum_{j=1}^{J} s_j(t)$, where J is the number of tracks, and the track signal $s_j(t)$ is a time-series of harmonic notes [37, 47, 48, 55]. A note, similar to a phoneme of speech [79, 80], is the smallest signal unit of a piece of music consisting of a fundamental frequency and a set of harmonics [27, 28, 72].

2.2 Adversarial Audio Attacks

Given a classifier with prediction function $f(\cdot)$ which takes the input audio signal $s(t)$ and outputs the correct label $f(s(t)) = y$, existing adversarial audio attacks [16, 53, 78] aim to add a small signal perturbation $\delta(t)$ to the original audio signal $s(t)$, and then supply the perturbed signal $\hat{s}(t) = s(t) + \delta(t)$ to the classifier that accordingly generates an incorrect label. The method of creating $\delta(t)$, which mainly inherits from the fundamental framework in the image domain [15, 65], can be formulated as

$$\begin{aligned} &\text{minimize} && \|\delta(t)\|_p \\ &\text{subject to} && f(\hat{s}(t)) \neq y, \end{aligned} \tag{1}$$

where $\|\delta(t)\|_p$ denotes the L_p norm of the perturbation $\delta(t)$ [15, 30]. The objective of (1) is to minimize the change of the perturbed signal $\hat{s}(t)$ from the original $s(t)$. Since it is computationally difficult to solve (1), many variants of formulating the adversarial audio attacks have been proposed for distinct attack scenarios, such as speech recognition [16, 53, 78], speaker recognition [18, 81], and music copyright detection [57]. To still make $\hat{s}(t)$ look like $s(t)$, these formulations limit the L_p norm of the perturbation $\delta(t)$ within a given threshold ϵ, i.e., $\|\delta(t)\|_p \leq \epsilon$. The L_∞, L_2, and L_0 norms are commonly adopted in the literature to create adversarial attacks targeting various audio signal classifiers [16, 18, 38, 42, 81].

2.3 Motivation and Design Intuition

Although existing adversarial audio attacks mathematically limit the magnitude of the perturbation $\delta(t)$ via $\|\delta(t)\|_p \leq \epsilon$, it is still not clear whether such a constraint is the most effective to make the perturbation unnoticeable by human beings. For example, a few studies [15, 53] have noted the concern on whether the L_p norm metric is appropriate to measure the signal similarity from the human perception perspective. In other words, there is no evidence to show that the deviation in human cognition can be represented by $\|\delta(t)\|_p$. As a result, we are motivated to investigate the problem. Our goals are twofold: i) relating the change of a music signal to the deviation of human perception and ii) finding a new way to create the perturbation that is unnoticeable by human beings as much as possible. To achieve these goals, our design consists of three major components.

(1) *Reverse-engineering human perception:* we treat human perception as a black box and design a human study to quantify human perceived deviations. Specifically, we invite volunteers to assign a rating of perceived deviation to measure the difference between the original and perturbed signals. Then, we reverse-engineer the black box via regression analysis to build a relationship between the signal deviation and the human-perceived deviation.
(2) *Reformulating the adversarial audio attack as the perception-aware attack:* based on the relationship found in the human study, we establish the perception-aware attack framework with the objective to quantitatively minimize the perceived deviation while attacking audio classification.
(3) *Demonstrating a realistic attack against a music copyright detector:* based on the new attack framework, we create adversarial music against YouTube's copyright detector. We demonstrate via experiments the effectiveness of the attack in terms of success rate and human-perceived deviation.

2.4 Threat Model

In this paper, we consider an attacker that aims to find a perturbation $\delta(t)$ to a music signal $s(t)$ such that $\hat{s}(t) = s(t) + \delta(t)$ leads to an incorrect output of an audio signal classifier, which is similar to the goal of existing audio attacks [16, 42, 57, 65, 78, 81]. At the same time, the attacker is designed to be aware of how $\hat{s}(t)$ affects the human perception and minimizes its perceived deviation from $s(t)$. We assume that the attacker has no knowledge of the algorithm design or parameter choices in the classifier, but has access to the classification result of any input signal. We also assume that the attacker has no access to the classifier's training database. A representative commercial scenario is that an attacker wants to bypass YouTube's copyright detector [57] and use copyrighted music content in an unauthorized way to attract more online views for advertisement revenue gain.

3 REVERSE-ENGINEERING HUMAN PERCEPTION OF MUSIC SIGNALS

In this section, we present how to quantify the human perceived deviation of music signals. We first analyze the key features for the signal quality, then conduct the human study, and lastly present the study results and regression analysis.

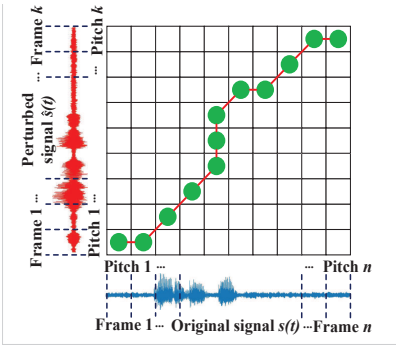


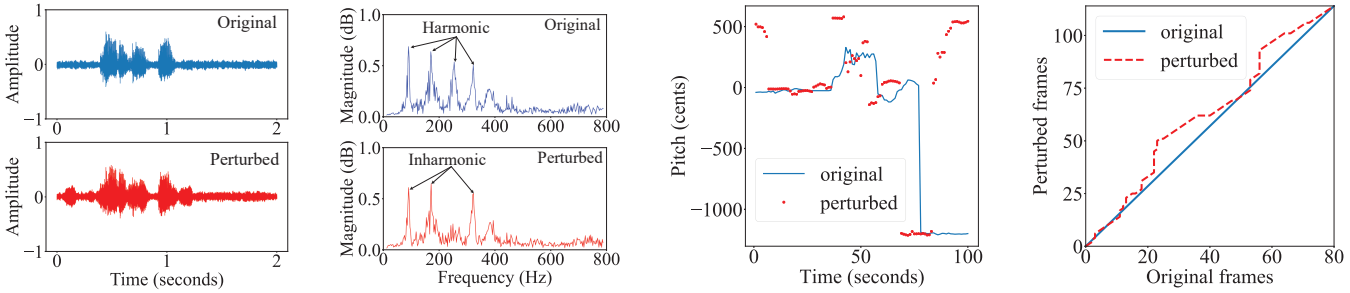
Figure 2: Computing deviation values via DTW.

3.1 Audio Features for Human Perception

Based on existing studies in audio engineering [31, 40, 46, 52, 68], there are four widely-used features: pitch, rhythm, timbre, and loudness. Pitch is the subjective perception of highness or lowness of a sound, and is referred to as the fundamental frequency ω_0 of a note [35, 43]. Rhythm is described as the tempo of the musical sound [68], which depends on the length of each note and the time intervals between adjacent notes. Timbre is the mixture of the harmonics, which brings the "color" to music [43, 75], and it is similar to the characteristics of the speech [24]. Loudness measures the intensity of an audio signal and can be seen as the energy level or the volume of the signal [68].

In the following, we briefly introduce the commonly-used methods to compute the feature deviations between two signals $s(t)$ and $\hat{s}(t)$ in the literature. For each feature, the procedure is the same and shown in Fig. 2: $s(t)$ and $\hat{s}(t)$ each will be separated into frames with a small time interval (e.g., 16ms [31]). The signal samples in each frame are used to generate a feature value (e.g., pitch value). The feature values from all frames constitute a time-series data vector. Then, an algorithm called Dynamic Timing Warping (DTW) [58, 59] is used to quantify the similarity between the time-series vector for $s(t)$ and the one for $\hat{s}(t)$, and generate a vector of frame-wise deviation values for the feature. The advantage of DTW over the Euclidean distance is that DTW can reduce the time distortion [54] via finding an optimal path between two time-series vectors. For instance, the red line in Fig. 2 indicates the DTW path between $s(t)$ and $\hat{s}(t)$.

- Pitch: The pitch value in each frame is the basic frequency ω_0 obtained via pitch estimation, which is a maximum likelihood estimation problem [25] via finding ω_0 from harmonics $\sum_{m=1}^{M} m\omega_0$. The estimated pitch values from all frames form a time series for each signal and then DTW is used to generate the vector of frame-wise pitch deviation values between the two signals.
- Rhythm: Rhythm computation is based on pitch estimation. A deviation value for rhythm between two frames is computed as the linear regression error in DTW during computing the deviation value for pitch [46]. All these values generated during DTW form the vector of frame-wise deviation values for rhythm.
- Timbre: The timbre value for each frame is computed as a Mel-Frequency Cepstrum Coefficient (MFCC) [23]. The vector of frame-wise deviation values for timbre is the result of the DTW between the MFCC vectors for $s(t)$ and $\hat{s}(t)$.



(a) Waveforms **(b) Harmonic note spectrum** **(c) Pitch deviation** **(d) Rhythm deviation**

Figure 3: Impacts of a noise-like perturbation on the music features: a 2-second attack example "Boom Boom Pow" from existing work [57]. Specifically, (a) and (b) shows the waveforms and spectrums, respectively; (c) and (d) show the pitch contours and the rhythm DTW paths between the perturbed and original signals, respectively.

- Loudness: Loudness is closely related to the L_p norm used in existing adversarial attack formulations (1). The loudness for each frame is usually calculated as the short-term log-energy [68], which is the logarithm of the total energy of the frame. After two short-term log-energy vectors for $s(t)$ and $\hat{s}(t)$ are obtained, the DTW between them generates the vector of frame-wise deviation values for loudness.

The last step for each feature is to aggregate the computed vector of frame-wise deviation values into a single value to represent the overall feature deviation. According to existing studies [31, 56], the non-linear average calculation is commonly adopted for pitch and rhythm aggregations, and linear averaging is used for timbre and loudness. After the aggregations, the resultant four feature deviation values form a final feature deviation vector to describe the audio characteristic deviation from $s(t)$ to $\hat{s}(t)$.

3.2 Impacts of Audio Feature Deviations

To have a good sense of how pitch, rhythm, timbre, and loudness change in a perturbed music signal, we show the feature deviations caused by an adversarial example in [57] in Fig. 3.

As [57] adopted an L_p norm based formulation to create adversarial audio and limited the L_p norm of the perturbation, Fig. 3a shows that there is a minor waveform change in the time-domain between the original and perturbed music signal. This indicates that the perturbation only incurs a small energy or loudness change to the original signal.

Next, we look at the waveform change in the frequency-domain and compare the power spectrum in Fig. 3b. The observed change is more evident than the time domain in Fig. 3b: the third harmonic in the original harmonics is suppressed, which leads to inharmonicity in the signal and can negatively impact the timbre feature and accordingly the audio quality.

If we look at the pitch contours (i.e., the curves drawn by connecting all pitch values over time) for the original and perturbed signals in Fig. 3c, we observe the evident difference of the pitch features between the two signals. Similarly, Fig. 3d shows the optimal DTW path of the perturbed signal to the original one. Intuitively, a music signal with the minimal rhythm deviation should have a nearly straight line DTW path. Fig. 3d shows that the DTW path of the perturbed signal is tortuous compared with the original one.

Note that creating adversarial music inevitably causes some distortions of the original signal. Fig. 3 demonstrates that there may exist some way to better coordinate such distortions among all audio features to mimic the original signal's quality as much as possible since they are eventually perceived by humans. If we look at the basic adversarial audio attack formulation used in recent research [18, 42, 57], the L_p norm of the additive noise is only relevant to the loudness feature without a clear relation to the other three features. It is evident that L_p norm is much easier to compute than pitch, rhythm, and timbre via gradient descend. At the current stage, we do not focus on the computational aspect but on the human perception aspect and continue to understand how these features affect human perception.

3.3 Human Study Procedures and Setups

To understand how different features affect human perception. We conduct a human study with the procedure shown in Fig. 4: we first generate a dataset that consists of pairs of original and perturbed music signals. For each pair, we can compute (according to the procedure in Section 3.1) the deviation values for the four features, which form a feature deviation vector. Then, we invite every human participant to assign a deviation rating to each pair based on his/her perceived difference. Next, considering the feature deviation vectors as the inputs and the human ratings as the outputs, we use regression analysis to find the best model to describe the relation between the vectors and the ratings. In this way, we can reverse-engineer the human perception process to build an approximation model to quantitatively predict how much a perturbed signal is perceived by a human.

Dataset Generations. Since there is no publicly available dataset that provides various versions of perturbed music signals, we propose to generate our own dataset with the following requirements: (i) sufficient diversity of music genres, (ii) sufficient perturbations from the pitch, rhythm, timbre, and loudness perspective, and (iii) slight or moderate perturbation to avoid making participants feel overly noisy.

We build a dataset of 60 pairs of original and perturbed music clips from the genres of Pop, Hip-hop, Rock, Jazz, Classical, R&B, Country, and Disco. To make participants concentrate on each small perturbation, we crop each music clip to a 5-second WAV format

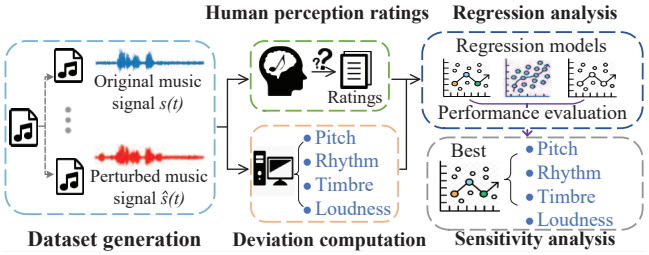


Figure 4: The human study procedure and steps.

(16kHz, 16-bit PCM, Mono) to avoid audio compression. As there is no guideline or reference to standardize the dataset generation for our study, we aim to create perturbed signals with different feature deviations and varying intensities for human participants such that the data is diverse for regression analysis. Specifically, we use two main mechanisms to create perturbed music clips.

- Additive noise: an intuitive method is to inject additive noise into the original music. The noise will affect all four features at the same time. To broadly affect the original music, we consider injecting the noise from three aspects: amplitude, frequency and time. To control the amplitude of the noise, we can choose the signal-to-noise (SNR) level from 0dB, 5dB, 10dB, and 15dB [70]. To inject frequency-sensitive noise, we use both white noise [71] (covering all frequencies with equal intensity) and colored noise (with the power concentrated at certain frequencies). To make noise time-varying, we set random duration and interval of the noise, but the total injection duration is less than the half of the original music length. In addition, since existing audio perturbations (e.g., in [57]) cause noise-like sounds, the additive noise data rated by human participants should help build a model to properly predict the deviations of noise-like perturbations.
- Additive notes: To ensure distinctive deviations among all music features, we also inject additive notes to the original music. To inject notes with the pitch manipulation, we randomly choose notes with the pitch value from 27.5Hz to 4186Hz [41] (88 notes space). For rhythm manipulation, we randomly select the additive notes with different lengths and ensure the intervals between adjacent notes are less than 50% of the original signal's length. To create timbre deviation, we select different instruments to play the additive notes as long as the notes are within the valid pitch ranges of those instruments.

Human Participation. We recruited 35 participants who are college students with ages falling between 20 and 35. All the participants are volunteers without any compensation. Each participant was asked to listen to each pair of the original and perturbed music clips, and then assign a deviation rating on a Likert scale [66] according to his/her overall music perception: 0–1 perfect perceptual quality with imperceptible noise, 1–2 good perceptual quality with quiet noise, 2–3 noticeable with slight noise, 3–4 noticeable and noisy, and 4–5 very noisy. More specifically, 1–2 means volunteers can only notice some small perturbation after listening to a part of music clips many times, and 2–3 indicates the deviation can be noticed by listeners but not noisy. During the experiments, all the volunteers were given the same earphone with the same initial volume setting. They can listen to a music clip as many times as they want.

Ethical Considerations: Our study involved human participants that assigned ratings by listening to music. The full protocol was reviewed and exempted by our Institutional Review Board (IRB), which has determined that the study involves the minimal risk for human participants (i.e., the risk is no more than the one that they face during their daily lives). We follow the approved protocol to inform them of the full study procedure and protect their identities without publishing any personally identifiable information.

Reverse-Engineering via Regression Analysis. Given the computed feature deviations from the original and perturbed music clips as well as the human participant ratings of their perceived deviation, we aim to find the best regression model $M^* \in \mathcal{M}$ in the model set $\mathcal{M}$ to minimize the mean squared error (MSE) of regressed prediction, i.e.,

$$M^* = \arg\min_{M \in \mathcal{M}} \mathbb{E}\|r - M(d_p, d_r, d_t, d_l)\|_2^2, \tag{2}$$

where r is the human participant rating, d_p, d_r, d_t, and d_l are the deviation values (computed according to the procedure in Section 3.1) for pitch, rhythm, timbre, and loudness, respectively. In our study, we choose Linear Regression [32, 68], Support Vector Regression, Random Forest, Logistic Regression, and Bayesian Ridge to form the model set $\mathcal{M}$. With M^* found in (2), we use it to quantitatively predict any human-perceived deviation given a pair of original and perturbed music signals.

3.4 Result Analysis and Discussion

Fig. 5 box-plots all the human ratings (ranging from 0 to 5) for individual pairs of music clips from our human study. We can find in Fig. 5 that human perception is indeed subjective: each pair of music clips has a range of deviation ratings by different participants; there are always rating outliers for a pair of music clips. Fig. 5 also shows that overall, the ratings and the 25%-75% boxes are roughly evenly distributed from 0 to 5, which offers sufficient data diversity for regression analysis.

Regression Analysis. We first use each of Linear Regression, Support Vector Regression (SVR), Random Forest, Logistic Regression, and Bayesian Ridge to model the relationship between feature deviation values and the average human rating, and find the best model with the minimum MSE. We show the MSEs of different regression models during testing in Table 1.

Table 1: MSEs of different regression models.

Model:	Linear	SVR	Random Forest	Logistic	Bayesian
MSE:	1.2351	0.8558	**0.1541**	1.6572	1.2628

Through regression analysis, we find that Random Forest performs the best among all the five regression models. As Table 1 shows, Random Forest leads to an MSE of 0.1541, which is substantially better than Support Vector Regression that achieves the second with an MSE of 0.8558, but an over 5 times increase from Random Forest. The other models result in even worse MSEs. As a result, we choose Random Forest as our regression model to predict the human-perceived deviation. Specifically, given a pair of original

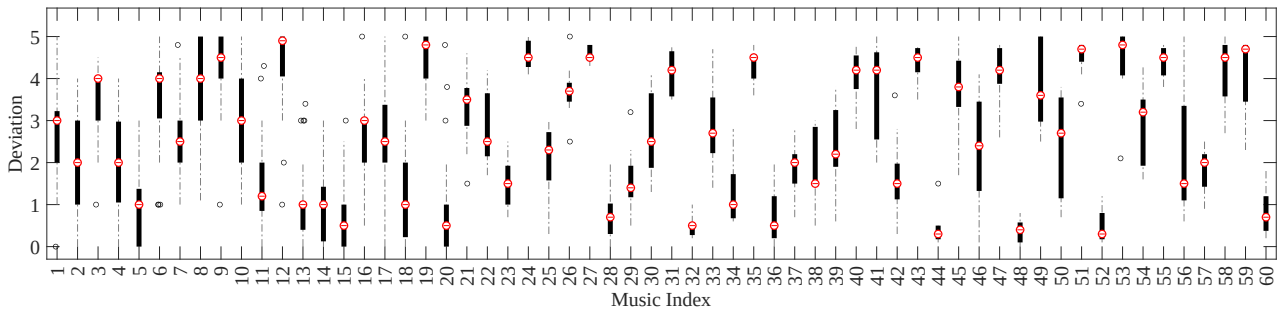


Figure 5: Distributions of human ratings of perceived deviation for all pairs of music clips.

and perturbed signals, we name the prediction output of Random Forest as quantified deviation (qDev).

Correlation Analysis. Then, we analyze to what extent qDev values and realistic human ratings move in tandem; that is, an increase or decrease of value for one will lead to the same for the other. This is important because when creating an adversarial attack against a classifier, we aim to reduce the qDev value of a perturbed signal (so its deviation rating by a human should also decrease) such that the perturbation is hardly noticed by a listener. We use Spearman's rank correlation coefficient [21, 61] to model the correlation in our study. Spearman's coefficient is a commonly used statistic measure to evaluate the relationship between two variables using a monotonic function, where value 1 or -1 indicates that the two always move in the same or opposite direction; value 0 means no correlation.

Table 2: Spearman's coefficient between the human rating and a deviation measure.

Deviation Measure:	L_2	L_∞	SNR	qDev
Spearman's Coefficient:	0.3909	0.0893	0.0134	**0.9608**

Table 2 lists the Spearman's coefficients between the human rating and each of the following deviation measures: L_2 norm [57], L_∞ norm [18, 57], SNR [19, 79], and qDev from Random Forest. It is seen from Table 2 that qDev has a very high correlation with the realistic human rating, indicating it can be quite useful for predicting a human-perceived deviation of a signal. In other words, minimizing qDev in a mathematical formulation to form an audio signal perturbation would be most likely suppress a human's attention to the signal deviation caused by the perturbation. Interestingly, we also observe that the commonly used L_p norms and SNR are in fact not well related to human perception (e.g. L_2 norm has the best correlation of 0.3909). Table 2 offers quantitative evidence to echo the concern raised in related studies [15, 53] that suggests new ways to measure the human perceptual similarity may be needed.

Sensitivity Analysis. To explore which feature is potentially more important than others in human perception, we conduct sensitivity analysis via the One-at-a-time (OAT) strategy [6, 11, 49]: we remove in turn pitch, rhythm, timbre, and loudness to form three-feature inputs for regression, and measure the MSE of the resultant regression. We find Random Forest is always the best in our OAT analysis to minimize the MSE with only three features reaming as the inputs.

Table 3: Sensitivity analysis for each feature.

Excluding:	Pitch	Rhythm	Timbre	Loudness	None
MSE:	0.1891	0.1581	0.1889	0.3539	0.1541

Table 3 shows the MSE of Random Forest for each regression of excluding pitch, rhythm, timbre, and loudness in turn. From Table 3, loudness that represents the energy of the perturbation appears to be the most sensitive feature to human-perceived deviation. For example, removing loudness leads to a 129% MSE increase from 0.1541 to 0.3539. But it is clear that the other features individually contribute to the overall human perception, and removing one of them causes more MSE in the regression.

Overall, we find in the human study that Random Forest is the best regression model to yield the minimum MSE to predict the human rating as qDev. Simpler regression models, such as Linear Regression or SVR, do not perform as well as Random Forest. This may also confirm that human perception is indeed a complicated process. In addition, qDev is a much more appropriate metric than the conventional L_p norm or SNR in terms of both MSE and Spearman's correlation with the human rating, and the features of pitch, rhythm, timbre, loudness all contribute to the overall perception.

4 PERCEPTION-AWARE ATTACK STRATEGIES

With the metric of qDev regressed via Random Forest from audio features, we reformulate the problem of creating adversarial music signals into a perception-aware attack framework. We then analyze how to narrow down the search space in the reformulation, and eventually find an efficient solution via dynamic clipping.

4.1 Problem Reformulation

Existing studies [16, 42, 78, 81] solve the original optimization problem in (1) via finding a sub-optimal yet efficient alternative solution. For our perception-aware reformulation, it is natural to think about reformulating existing alternative solutions by directly replacing its L_p norm with the new metric of qDev. However, such a reformulation no longer offers the advantage of computational efficiency because the process of computing audio features in qDev is unfortunately non-linear, non-convex, and non-differentiable

[25]. Accordingly, we formulate the perception-aware attack by replacing L_p norm with qDev in the original form (1) as

$$\begin{aligned} &\text{minimize} && \text{qDev}(s(t), \hat{s}(t)), \\ &\text{subject to} && f(\hat{s}(t)) \neq y, \end{aligned} \tag{3}$$

where $\text{qDev}(s(t), \hat{s}(t))$ denotes the qDev between the perturbed signal $\hat{s}(t) = s(t) + \delta(t)$ and the original one $s(t)$. To ensure $\hat{s}(t)$ to be a valid waveform, we always constrain the normalized amplitude of each of its sample points to be in $[-1, 1]$ [18].

Finding the optimal solution to (3) becomes even more difficult than the original one in (1) because computing qDev involves a much more complicated process than the L_p norm. Our strategy is to analyze what properties the perturbation signal $\delta(t)$ should have towards finding a solution to (3).

4.2 Perturbation Signal Property Analysis

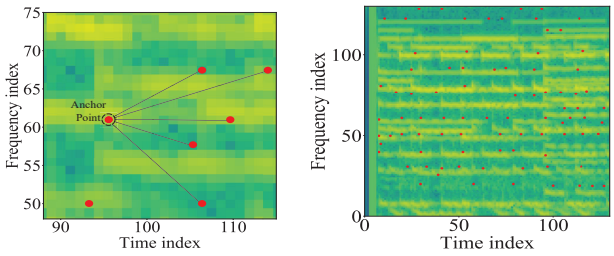
Since the solution to (3) is computationally intractable, we have to narrow down the search space for the perturbation signal $\delta(t)$ by analyzing what properties it should have.

The reformulation (3) means two obvious goals that the perturbation signal $\delta(t)$ should achieve: i) misclassification (i.e., the attack should fool the classifier) and ii) minimized qDev (i.e., it also produces good perceptual quality a human can perceive). At first glance, the two goals seem to contradict with each other (as the best perceptual quality of music indicates no change of its signal and thus no attack success). We need to explore one step further to understand what audio features $\delta(t)$ needs as a result of each of the two goals, then consider all needed features jointly to reconcile any conflict to construct a search space of $\delta(t)$ that is sufficiently narrowed down towards a feasible solution.

Properties for Attacking Audio Fingerprinting. First, we consider what feature properties $\delta(t)$ should have towards launching a successful attack. A key technique for audio signal classification is audio fingerprinting [12]. The technique and its variants have been widely adopted in audio signal watermarking [9, 20], integrity verification [29], music information retrieval [17, 51, 73], broadcast monitoring [5, 34, 50] and copyright detection [57].

The essential idea in audio fingerprinting is to consider certain high-energy areas of an audio signal in the spectrogram as its fingerprints. As an example shown in Fig. 6(a) [73]: an energy peak (anchor point) is paired with other peaks within a certain target area in a signal's spectrogram, then the fingerprints are computed based on the frequency information of the peaks and the time intervals between them. Fig. 6(b) shows there are many peaks in a signal's spectrogram that lead to a large number of fingerprints for audio signal classification and identification.

As we can observe from Fig. 6, peaks in the spectrogram are a key feature for audio signal classification. These peaks are usually the results of a mixture of high-energy points of audio signal harmonics [29, 33, 73]. From the attacker's perspective, creating new positions of harmonics in the spectrogram should be a direct way to manipulate the fingerprints, which can lead to the misclassification of the signal. In the audio features, timbre is the most relevant to the harmonics of the signal [43, 55]. Given an energy threshold (that represents the loudness) for perturbation $\delta(t)$, a good way to create the attack is to affect the feature of timbre for the signal.



(a) Fingerprinting generation. **(b) Distribution of peaks.**

Figure 6: Fingerprinting: finding all spectrogram peaks.

Properties for Good Perceptual Quality. Next, we consider what feature properties $\delta(t)$ should have for good music perceptual quality. From the sensitivity analysis in the human study in Section 3.4, all features, pitch, rhythm, timbre, affect the human perception of signal deviation or the metric of qDev. The change of any of them may result in an increase of qDev and accordingly a noticeable change by human perception. To further explore the relationship between the musical features and human perceived deviations, we remove two features at a time to measure the MSE of the resultant regression.

Finding Feasible Search Space. To summarize, it would be good to 1) change the feature of timbre for a potentially successful attack, and 2) manipulate only one feature while keeping the others unchanged as much as possible to maintain the perceptual quality. to reconcile the two requirements: we propose to change timbre much more than the other features.

Now the question becomes how to create $\delta(t)$ with a quite different timbre feature while maintaining almost the same pitch and rhythm features. The traditional perturbation design in (1) usually generates a noise-like perturbation and is not able to create this required signal because it causes all distortions of pitch, rhythms, and timbre (as shown in Fig. 3). As a music signal consists of well-crafted, human-enjoyable musical notes, we propose to create $\delta(t)$ by reproducing the same music notes via new instruments. The timbre feature is always associated with the harmonics, and we can find these natural harmonics from the instruments. In this way, the timbre of $\delta(t)$ can be changed substantially due to different harmonic characteristics of new instruments; but pitch and rhythm may deviate less if we find appropriate instruments to play the same notes. To demonstrate the feasibility of our design, we compare the feature deviations of a perturbed music signal mixed by randomly-generated noise and instrument-generated music notes.

Table 4: Noise vs notes played by a different instrument.

	Pitch	Rhythm	Timbre	Loudness	qDev
Instrument:	0	0.85	25320	2873	2.23
Noise:	0.9049	7.239	19521	1988	3.86

As shown in Table 4, the additive instrument produces a higher loudness value than noise (indicating a more energy level); at the same time, it generates more timbre deviations (25320 vs 19521) but less pitch and rhythm deviations than the noise. Depending on the difference between $s(t)$ to $\hat{s}(t)$, the non-linearly aggregated

pitch and rhythm deviations have values commonly in the range from 0 to 50, and the linearly aggregated timbre and loudness deviations usually range from zero to tens of thousands. There exists an obvious deviation gap between instrument-generated notes and randomly-generated noise of different features. We also use qDev to quantify the deviations, and the instrument-generated notes have a clearly lower qDev value than random noise (2.23 vs 3.86). This makes it a much more desirable signal component for $\delta(t)$ in terms of both human-perceived quality (low qDev) and attack effectiveness (more timbre variation).

Consequently, we can effectively narrow down the search space by considering $\delta(t)$ as a linear combination of signals consisting of the same music notes played by different instruments for the original music signal. Then, generating the perturbed signal $\hat{s}(t) = s(t) + \delta(t)$ is like finding "subtle" instrumental track signals then optimally remixing them (based on qDev) into the original music.

It is worth mentioning that a music signal can consist of both instrumental and vocal tracks. It is possible to add a new vocal track (i.e., the same vocal notes sung by a different voice to change the feature of timbre) into the perturbation $\delta(t)$. As it is easier to generate instrumental signals by computer music synthesis, we only use instrumental tracks to form $\delta(t)$ in this paper.

4.3 Perception-Aware Attack Formulation

With the shrunk search space, we write $\delta(t) = \sum_{k=1}^{K} \theta_k \delta_k(t)$, where K denotes the number of different instrumental tracks, $\delta_k(t)$ is the k-th instrumental track signal, and θ_k is the non-negative weight for $\delta_k(t)$. Next, we reformulate (3) into a perception-aware attack of finding the best linear weights θ_k in $\delta(t)$ to minimize the qDev:

$$\underset{\{\theta_k\}_{k\in[1,K]}}{\text{minimize}} \quad \text{qDev}\Big(s(t), s(t) + \sum_{k=1}^{K} \theta_k \delta_k(t)\Big) \tag{4}$$

$$\text{subject to} \quad f\Big(s(t) + \sum_{k=1}^{K} \theta_k \delta_k(t)\Big) \neq y,$$

$$\sum_{k=1}^{K} \theta_k = \epsilon, \tag{5}$$

$$\mathcal{P}_{s(t)} \subseteq \mathcal{P}_{\delta_k(t)} \; \forall\, k \in \{k | k \in [1,K], \theta_k \neq 0\}, \tag{6}$$

where (5) ensures the energy level of the perturbation signal $\delta(t)$ is less than a threshold ϵ, $\mathcal{P}_{s(t)}$ and $\mathcal{P}_{\delta_k(t)}$ in (6) represent the sets of pitch values in the original signal $s(t)$ and the k-th track signal $\delta_k(t)$, respectively; (6) ensures that $\delta_k(t)$ covers the pitch range of $s(t)$ so the pitch feature of $\delta_k(t)$ does not deviate much from $s(t)$.

The optimization (4) is a problem of finding the optimal linear weights. Although still non-differentiable, (4) opens a door for a grid search based heuristic solution. Specifically, we can let each linear weight θ_k be a multiple of a small step Δ (that is a fraction of the threshold ϵ in (5)), then enumerate all combinations of possible values for $\{\theta_k\}_{k\in[1,K]}$ to find a solution to (4). For example, setting $\Delta = 0.1\epsilon$ and $K = 10$ produces 92,378 combinations in total. Iterating through them, though not very efficient, is quite feasible for an attacker's computing capability today.

4.4 Dynamic Clipping

The optimization in (4) finds out a perturbation signal $\delta(t)$ based on the entire duration of the original signal $s(t)$. However, a piece of music can consist of multiple segments with audio characteristics varying within a wide range of instruments and vocals, creating distinct timbre features. For better perceptual quality and attack effectiveness, it is necessary to segment $s(t)$ into N clips according to evident timbre changes and create the perturbation for each clip using the clip-wise optimization based on (4). We call this procedure dynamic clipping.

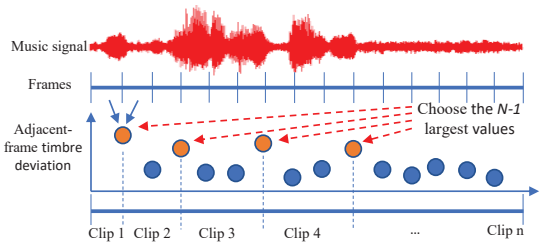


Figure 7: Overview of dynamic clipping.

Fig. 7 shows the process of dynamic clipping: to dynamically segment $s(t)$ into N clips, we first separate $s(t)$ into small frames and compute the timbre deviation between each pair of adjacent frames (using the timbre deviation calculation in Section 3.1). Then, we identify $N-1$ pairs which have the $N-1$ largest adjacent-frame deviation values, as they contain the most evident $N-1$ changes of timbre over the duration of the music. We use the timing boundary between two frames in a pair as a timing position to segment $s(t)$. In this way, $s(t)$ is segmented into N clips, each of which will be used to find a corresponding perturbation based on (4).

5 REALISTIC BLACK-BOX ATTACK AGAINST COPYRIGHT DETECTOR

In this section, we create a realistic attack based on the perception-aware attack framework in Section 4. We choose the YouTube copyright detector as our target as YouTube has exhibited some robustness against noise and perturbations [57]. Because there is no knowledge of YouTube's design, we create our own detector based on open-source information for an adversarial transfer attack. We first present how to generate additional instrumental tracks for the perturbation signal given a music signal, then describe the design of our detector as a surrogate model for YouTube's detector.

5.1 Perturbation Signal Generation

Perturbation signals generated by (4) require the detailed music notes of the original music. For a popular piece of music, its Musical Instrument Digital Interface (MIDI) file is usually available in online databases (e.g., FreeMidi.org and Nonstop2k[1]). The MIDI file contains all instrumental tracks with music notes. We use Music21[2] to play a downloaded MIDI file with different instruments to form a perturbation for (4). To achieve the diversity of the timbre feature for (4), we consider an instrument set of instruments across the four families *stringed* (Guitar, Electric Guitar, Violin, Viola, Cello, Bass, Electric Bass), *woodwind* (Clarinet, Flute, Saxophone, Oboe, Bassoon), *brass* (Trumpet, Baritone, Tuba, Horn, Trombone), *keyboard* (Piano, Electric Piano). We empirically select at most two instruments from each family based on a music genre to reduce

[1]FreeMidi.org:https://freemidi.org/, Nonstop2k:https://www.nonstop2k.com/
[2]Music21 is a Python-based toolkit for computer-aided musicology. In this work, we use it to produce different instrumental tracks playing the same musical notes

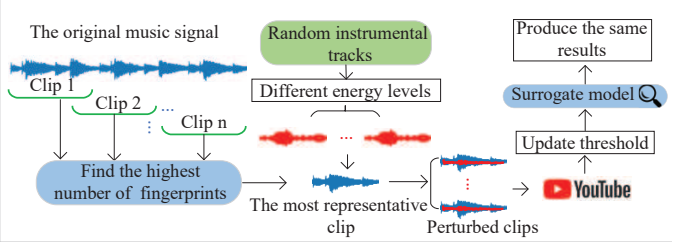


Figure 8: Process of obtaining the threshold from YouTube.

the computational complexity and the pitch range requirement for perturbation generation in (6).

5.2 Surrogate Detector

Audio Fingerprints. A copyright detector takes audio fingerprinting features as the input. We select the fingerprints and their extraction method introduced in [73]. We extract fingerprints by considering the time, frequency, and amplitude data of the audio. Specifically, we use Fast Fourier Transform (FFT) to generate a spectrogram of an audio signal and extract the spectral peaks of acoustic harmonics, which are shown invariant and reproducible from signal degradation [13] and robust to noise and distortion [73]. We then apply the fast combinatorial hashing method [73] to form these fingerprints to hashes for the similarity comparison later.

Detection Design. The detection is built to compute the similarity of the fingerprints of an input signal to the detector's database. If the similarity score is higher than a similarity threshold, the detector will raise an alarm. To ensure our surrogate detector has a degree of transferability to YouTube's detector, we must adopt a threshold that is similar to YouTube's. We note that our objective is not to precisely rebuild YouTube's model, but to choose an appropriate threshold (even in a rough way) such that we can use the surrogate detector to predict the output label during minimizing qDev in (4). Because music consists of diversities of audio features, we choose one threshold for each of 8 music genres: Pop, Hip-hop, Rock, Classical, Jazz, R&B, Country, and Disco.

Fig. 8 shows the process we use to approximately calibrate the surrogate detector's threshold towards YouTube's. This process is similar to the one proposed in [18] that estimates the threshold of a black-box model. In particular, to obtain the threshold for a music genre, we choose a song from the genre, crop it into clips, choose the most representative clip that contains the highest number of fingerprints among all the clips. Then, we randomly add instrumental track signals with different energy levels to this clip, generating a number of clips with perturbations of varying energy levels. We send these clips to YouTube to see the copyright detection results, and set the detection threshold for the surrogate detector such that it yields the same results as YouTube does.

6 EXPERIMENTS AND RESULTS

In this section, we present the experiments and results. We first describe the experimental settings, then discuss the audio perceptual quality and attack effectiveness of generated adversarial music.

6.1 Experiments Setup

Music Dataset: To cover a wide range of music data, we selected 32 top hits songs of the last 20 years from 8 genres: Pop, Hip-hop, Rock, Classical, Jazz, R&B, Country, and Disco. We created 56 clips of 5–10 seconds and 16 clips of 30 seconds clips for human evaluation, and 160 clips of 30 seconds for attack strength evaluation. We have verified that all the clips were copyright-detected by YouTube.

Default Experimental Setups: The default settings in (4) for the perception-aware attack include the search step $\Delta = 0.1\epsilon$, the number of instruments for perturbation generation $K = 7$, and the number of clips in dynamic clipping $N = 6$.

Attack Method Comparison: We compare the perception-aware attack with two recent attack methods: the ICML20 method against YouTube in [57] and the psychoacoustic attack framework [41, 53, 60]. Specifically, for the ICML20 attack, we directly adopted the source code provided by the authors of [57]; for the psychoacoustic attack, we followed the two stage attack introduced in [53]: we first used the ICML20 method to generate an adversarial music perturbation then applied the iterative process that involves the masking threshold [53] instead of the L_p norm in the loss function to improve the perception. Finally, We implemented a random noise attack method that adds random noise to music as a baseline case.

Here we provide a YouTube link that demonstrates adversarial clips created by the perception-aware attack in comparison with other attacks: https://www.youtube.com/watch?v=IfBAzmdN5ds.

6.2 Perceptual Quality of Adversarial Music

We first evaluate the perceptual quality of adversarial music created by the perception-aware, ICML20, psychoacoustic attack, and random noise attacks. In the experiments, given original music, we created perturbed music clips of 5–10 seconds under each attack by increasing the energy threshold of the perturbation such that the perturbed clip exactly bypassed YouTube's detector. For each perturbed clip, we used the Random Forest regressed qDev in Section 3.4 to predict its deviation from the original clip.

Human Evaluations: We involved 14 of 35 human volunteers in the training study in Section 3 to participate the evaluations. They formed Group 1 (G1) in our evaluations. We also recruited additional 15 college student volunteers, referred to as Group 2 (G2), to participate the evaluations. The age ranges of G1 and G2 are 20-34 and 22-33, respectively. The results of G1 can show the test accuracy of the regressed qDev model and the results of G2 can further demonstrate the generalizability of the regressed model (i.e., the training model built from a group of people can be used to predict the rating of another group of new people who are not in the training set). Every music clip in our evaluations was rated by all participants in both G1 and G2.

Human Rating vs qDev under Different Attacks: Fig. 9 illustrates the average human ratings and qDev values of the perception-aware, ICML20, psychoacoustic, and random noise attacks for each music genre. It is evident from the figure that the perception-aware attack always achieves much smaller deviation ratings and qDev values than the other three attacks. For example, for classical music, the perception-aware attack obtains ratings of 0.71 (G1) and 1.61 (G2) (indicating perfect and good perceptual quality according to

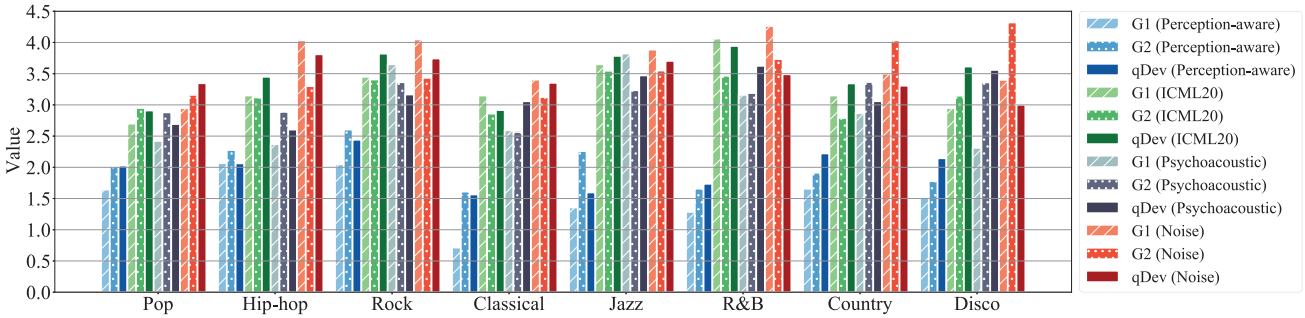


Figure 9: Human ratings and qDev values: Perception-aware, ICML20, psychoacoustic, and random noise attacks.

Table 5: The MSEs of qDev among genres (G1 vs G2).

Pop	Hip-hop	Rock	Classical
0.726 vs 1.027	0.161 vs 0.316	0.235 vs 0.382	0.480 vs 0.230
Jazz	R&B	Country	Disco
0.153 vs 0.333	0.426 vs 1.277	0.444 vs 0.299	0.735 vs 1.070

the rating guideline, respectively) while the ICML20, psychoacoustic, and noise attacks get 3.15 (G1) vs 2.91 (G1), 2.59 (G1) vs 2.56 (G2), and 3.40 (G1) vs 3.12 (G2), respectively (indicating noticeable and noisy). It is also observed that rock music seems harder to perturb for the perception-aware attack and has ratings 2.04 (G1) and 2.60 (G2) (noticeable with slight noise). Overall, Fig. 9 shows that the perception-aware attack achieves substantially better perceptual quality than the ICML20, psychoacoustic, and random noise attacks.

Accuracy of qDev-based Prediction: By comparing the qDev value with the human rating in every genre in Fig. 9, we can see that qDev is a good prediction to the human rating as the qDev does not deviate much from the average human rating for each genre. For example, the Hip-hop music created by the perception-aware attack has the qDev of 2.06 compared with the average human rating of 2.07 (G1) and 2.27 (G2). Table 5 shows the MSE between the qDev value and the average G1 rating compared with the MSE between the qDev value and the average G2 rating in each genre. The MSEs averaged over all genres are 0.4107 (G1) and 0.5848 (G2), which are both higher than the training MSE of 0.1541 in Table 1 in Section 3.4. Overall, it is observed that G2 incurs a slightly higher average MSE than G1 in the evaluation as new participants bring new subjective judgements.

Table 6: MSEs of different regression models.

Model:	Linear	SVR	Random Forest	Logistic	Bayesian
MSE-G1:	1.5826	2.2894	**0.4107**	1.9263	1.5012
MSE-G2:	1.9568	2.6169	**0.5848**	2.2103	1.8521

Table 6 compares the MSEs for G1 and G2 in different regression models built from the training in Section 3.4. We can see that all the test MSEs increase from the training MSEs in Table 1, and Random Forest still achieves the minimum MSE for both G1 and G2. In all regression models, Random Forest also exhibits the minimum increases from the training MSE to the testing MSE of G1 or G2.

Role of Additive Noise Data in Training: As we discuss previously, to evaluate the importance of additive noise data for building an accurate regression model in Section 3.3, we only use the additive note data to train a new qDev* metric and compare the prediction of qDev* with G1 and G2 ratings. Specifically, it is observed that when we replace qDev with qDev*, the MSE increases from 0.4107 to 2.0054 for G1, and from 0.5848 to 2.2944 for G2. As a result, additive noise is essential to build an accurate qDev model.

Impact of Dynamic Clipping: We also evaluate the impact of dynamic clipping in Section 4.4 on the overall perceptual quality of the perturbed music. We compare its performance with a static clipping design in which a clip is uniformly segmented into 6 smaller clips with equal length for perturbation generation. Table 7 shows the qDev values of the two designs. We can observe that dynamic clipping achieves uniformly better perceptual quality in all genres.

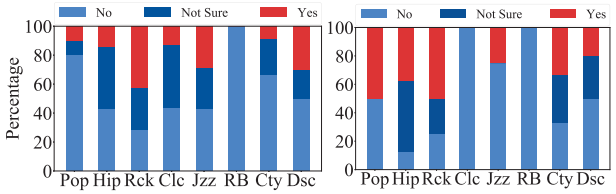
Table 7: qDev values in dynamic vs static clipping.

	Pop	Hip-hop	Rock	Classical
Dynamic:	1.8953	2.9250	2.6051	1.4956
Static:	2.2522	3.1854	3.1955	1.7558
	Jazz	R&B	Country	Disco
Dynamic:	1.8653	1.3897	1.6925	2.1933
Static:	2.9192	2.0925	2.0230	2.2588

Perceptual Quality without Reference. Previous experiments were conducted in a formal lab setting to quantify the perceived deviation via actual human ratings and qDev estimates. When a person listens to music during the daily life, there is no reference for him/her to perceive a deviation. The person may or may not notice an issue if the music is perturbed.

We conducted another experiment to measure how human participants perceive perturbed music without reference. In particular, we selected 16 30-second music clips, and asked two questions to each participant for each clip: (i) If familiar with the music: Assign a deviation rating based on your memory using the same rating guideline. (ii) Otherwise: Do you feel abnormal about the music? Please answer 1) Yes, 2) No, or 3) Not Sure. We totally received 140 ratings in (i) and 84 answers in (ii) from G1, and 162 ratings in (i) and 78 answers in (ii) from G2.

Table 8 shows the average human ratings along with the number of ratings received in each genre without reference and average qDev values for different music genres. We can find that the rating



(a) Group1 (b) Group2

Figure 10: Percentages of answers by participants of different groups unfamiliar with the given music. The numbers of answers received are 10, 14, 7, 16, 7, 8, 12, 10 in G1 (total: 84), and 6, 8, 8, 16, 8, 10, 12, 10 in G2 (total: 78) for Pop, Hip-hop, Rock, Classical, Jazz, R&B, Country, and Disco, respectively.

distribution among music genres is quite similar to Fig. 9. For example, the Classical music can still achieve nearly perfect perceptual quality of 0.86 (G1) and 1.73 (G2). Rock and Hip-Hop are the worst genres to perturb and make human participants feel noticeable with slight noise deviations. Interestingly, we find that the human rating of G1 and G2 for R&B music is 0.5 (nearly perfect) and 1.43 (good quality) without reference, which are both improved from the experiments with reference. The potential reason is that the additive instrumental track signals sound natural and embedded to the original music. It becomes hard for humans to recognize these timbre changes without reference. Overall, the G1 and G2 ratings are very similar in Pop, Hip-hop, Rock, and Jazz; and the ratings averaged over all genres are also close (G1: 1.62 vs G2: 2.12).

Table 8: Human ratings without reference and qDev.

	Pop	Hip-hop	Rock	Classical
G1 rating (140):	1.4500 (18)	2.4428 (14)	2.4867 (21)	0.8583 (12)
G2 rating (162):	1.9993 (24)	2.6408 (22)	2.7988 (22)	1.7367 (14)
qDev:	1.7850	2.7133	2.5653	1.6255
	Jazz	R&B	Country	Disco
G1 rating (140):	1.7500 (21)	0.5000 (20)	1.4458 (16)	1.4821 (18)
G2 rating (162):	1.8909 (22)	1.4333 (20)	2.6606 (18)	2.0053 (20)
qDev:	2.5679	1.4905	1.6925	2.1178

Fig. 10 depicts a more interesting result of the percentages of different answers by participants unfamiliar with the given music. We can see that most participants do not surely notice any abnormality in perturbed soft music (e.g., R&B and Classical). For example, no audience finds any issue in any R&B music for both G1 and G2; and 30% or more answers for Rock music clips are abnormal. Fig. 10 shows that the majority of participants (i.e., 81.25% in G1 and 69.70% in G2) do not clearly notice the music perturbations generated by the perception-aware attack. Considering the fact that participants may form a cognitive bias in the study (i.e., they might feel "obliged" or "mentally-focused" to identify an abnormality), we think that a casual listener without reference might be more unlikely to notice the perturbation of adversarial music created by the perception-aware attack.

6.3 Attack Effectiveness vs qDev

Next, we measure the attack success rates of the perception-aware, ICML20, psychoacoustic, and random noise attacks against YouTube.

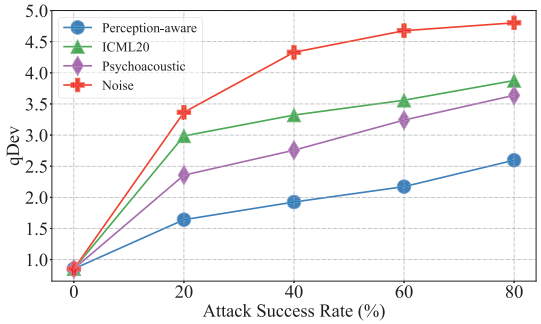


Figure 11: Attack success rates pairing with qDev.

As discussed in Section 5.2, the fingerprinting similarity thresholds in our surrogate detector were set roughly according to YouTube's detection results using a few music samples. But an adversarial music clip bypassing the surrogate detector does not necessarily mean that it will also evade YouTube's detection. In this experiment, we used the perception-aware, ICML20, psychoacoustic, and random noise attacks to each create 240 adversarial clips of 30 seconds (that 100% bypassed the surrogate detector), and then uploaded them to a private YouTube channel to test YouTube's copyright detection.

Pairing Attack Success Rates with qDev Values: We can get a 100% attack success rate by adding a sufficiently large perturbation to the original music, which can be extremely noisy. Hence, it is vital to pair the attack success rate with perceptual quality.

To this end, we focus on comparing the average qDev values of adversarial music clips created by perception-aware, ICML20, psychoacoustic, and random noise attacks under the same attack success rates against YouTube. Fig. 11 shows the comparison results. As shown in Fig. 11, higher attack success rates come with lower music perceptual quality in general. The qDev values of the perception-aware attack are always better than ICML20, psychoacoustic and random noise attacks for the same attack success rate. In particular, its qDev increases from 1.64 (good quality with quiet noise) to 2.53 (noticeable with slight noise) when the attack success rate goes from 20% to 80%; in contrast, the ICML20 attack has the qDev value increasing from 2.70 (noticeable with slight noise) to nearly 4 (very noisy). The qDev of the psychoacoustic attack ranges from 2.30 to 3.60, exhibiting better performance than ICML20 via its strategy to limit the energy within certain frequencies to suppress human attention. The random noise attack has the highest qDev value almost reaching 5 when the attack success rate is 80%. Fig. 11 demonstrates that the perception-aware attack is more effective against YouTube with better music quality.

Impact of Number of Instrumental Tracks: In our experiments for the perception-aware attack, the number of instruments used to generate the perturbation was set to be $K = 7$. It means that (4) always tries to find 7 weights assigned to 7 instrumental tracks. We can reduce the computational complexity by restricting the number of instrumental tracks. The less the number, the less the computational complexity (4) incurs. We conducted experiments to evaluate the impact of this number. Specifically, we still used 7 instruments but only choose 1, 3, or 5 out of 7 to form the instrumental track(s) as the perturbations to create the adversarial music clips. Under approximately the same attack success rates against YouTube, we

Table 9: Attack success rates and qDev values for different numbers of instruments.

Number of instruments:	1	3	5	7
Success rate:	78.13%	80.00%	79.38%	80.63%
qDev:	2.8901	2.7256	2.6713	2.5902

show the average qDev values of 160 adversarial music clips for each various instrument selection method in Table 9.

We find in Table 7 that the qDev value gradually decreases from 2.8901 to 2.5902 when we choose 1 to 7 out of 7 instruments to create the perturbations. This is expected as the objective of (4) is to minimize qDev and more instrument selections lead to a lower qDev value. One interesting observation is that choosing fewer instruments does not quite affect the attack success rate against YouTube. However, using only one instrument creates a quite loud music signal played by the instrument that is more identifiable to humans. Adding more instruments and distributing weights among them help suppress one single loud perturbation signal and makes the overall perturbation less identifiable.

6.4 Manipulating Other Music Features

Our perception-aware attack mainly focuses on generating perturbation via revising the timbre feature of music. It is also feasible to focus on manipulating pitch or rhythm to generate perturbation. There is still an open space to manipulate pitch or rhythm with potential optimizations. We adopt a randomized strategy to compare the three manipulations. In particular, we create pitch-based perturbations with a random energy via shifting music notes in its spectrogram by a random frequency, rhythm-based perturbations with a random energy via speeding up and slowing down the tempo of music notes at a random rate, and timbre-based perturbations with a random energy by randomly choosing one instrumental track playing the same music notes. Because of distinct natures in different generations, we must compare them under the same standard. We choose the qDev as the standard, and compare the attack success rates of randomly generated perturbations that have the same qDev value.

Table 10: Attack success rates with different manipulations.

qDev value:	1.5	2.5	3.5	4.5
Pitch:	9.38%	20.31%	29.69%	39.06%
Rhythm:	7.81%	15.63%	28.13%	54.68%
Timbre:	14.06%	31.25%	48.44%	70.31%

Table 10 shows the attack success rates of adversarial music clips created by the three randomized manipulation methods against YouTube under the different qDev values (64 clips for each manipulation method under each qDev level). We can see that the timbre manipulation always achieves higher success rates than pitch and rhythm manipulations in randomized generations.

Note that it is possible to further optimize the pitch or rhythm manipulation, or even combine all features to formulate a joint framework to minimize the qDev. However, involving them together may incur more search complexity. A balanced manipulation method among multiple features is also worth further studies.

6.5 Discussions

Though the perception-aware attack produce better-quality perturbations, we can still notice deviations (some are minor and others more noticeable) from the perturbed music. One may further improve the attack as discussed below.

Subtlety in small qDev difference: The metric of qDev based on current data regression of human ratings is not sufficiently sensitive to a small value difference. For example, a qDev value decrease from 4 to 1 should indicate an evident music perceptual quality improvement; however, a decrease from 2.1 to 2.0 may well fall into the error range of subjective judgements and is not fully correlated with music quality improvement. This may indicate that within this subtle qDev range, there might exist other improvements to make the perturbation sound more natural and attached to the original music. For example, some instruments (e.g., trumpet during our observations) can produce audio characteristics more identifiable to humans than some others, making its track evidently comparable to the foreground tracks (e.g., the main vocal track) in the original music. It may be necessary for (4) to select such an instrument to beat the classification via creating more timbre variations and minimize the qDev. There may exist other benchmarks in this case to further differentiate the selection of instruments as a small qDev difference may no longer help the selection.

Transition in dynamic clipping: Dynamic clipping segments a music signal into multiple clips and finds the optimal additional instruments for each clip. When the instrument sets for adjacent clips are chosen in a distinct way, human participants may be sharp enough to notice an instrumental transition. Smoothing this transition may result in a better experience; but the smoothing still needs to take suppressing audio fingerprints into consideration.

Robustness and bias of the regression model: Our human study and evaluations show that Random Forest achieves the minimum MSEs in both G1 and G2 compared with other models. Based on the Random Forest-regressed qDev, the perception-aware attack achieves better performance than the ICML20 and psychoacoustic attacks. As the human participants in our study are all college students with ages 20-35 and non-music experts, we acknowledge that our perceptual evaluations do not reflect music experts' judgements but show the opinions of general young populations. Extending the perceptual evaluations to other groups (e.g., elder populations and music experts) will help create more accurate and robust prediction.

Genearalizability to speech: The research in this paper focuses on the music domain. Our general human-in-the-loop methodology can be extended to the speech domain. As there are technical differences between fingerprinting music and recognizing speech, we expect this leads to non-trivial efforts to rebuild the qDev model based on human perception of speech difference, perform sensitivity analysis for speech features, and then shrink the search space by considering qDev-friendly acoustic signals (e.g., from a set of synthetic speech phonemes) to minimize the non-differentiable qDev, which is worth further studies and evaluations.

Vulnerability disclosure: The perception-aware attack does not cause an immediate operational impact, such as denial of service. Following the practice of responsible disclosure, we reported the issue of music copyright detection to Google. Google initially classified the case as an abuse risk. During the communication, Google

mentioned that a copyright content will be taken down from YouTube when the copyright owner makes a request. Google eventually made the decision not to track it as a security bug.

7 DISCUSSIONS ON DEFENSE STRATEGIES

In this section, we discuss potential defense strategies.

Existing audio defense: Audio pre-processing is a potential method to reduce the effectiveness of adversarial examples, as the small perturbation could be mitigated during the audio squeezing [18, 19, 79, 81] and audio compression [22, 42]. These defense methods are unlikely effective against the perception-aware attack as squeezing/compression does not quite change the spectrogram feature (e.g., the high energy harmonics will not be revised during the processing). On the other hand, these defense methods may not be desirable in some scenarios. For example, YouTube does not downgrade the music quality via squeezing and compression.

Improving audio fingerprinting: The advantage of audio fingerprinting is its computational efficiency [29, 33, 73]. Exiting research [26, 64, 73] focused mostly on extracting spectrogram features in a robust way for fingerprinting based detection. Although these fingerprints can be made robust to noise and pitch-shifting [26], the perception-aware attack creates additional harmonics and spectrogram features that can be extracted as fingerprints and fool the detection. We can potentially improve audio fingerprinting against the perception-aware attack by adding the pitch and rhythm features as other types of fingerprints. This, however, will incur substantially more costs because estimating pitch and rhythm incurs complicated maximum-likelihood estimation [25] than spectrogram based fingerprinting. There is a need to achieve a balanced tradeoff between detection accuracy and computational complexity.

Defense in machine learning: Another possible way to defend against the perception-aware attack is to leverage existing defense strategies from the machine learning community. In particular, adversarial training [7, 10, 30, 44, 62, 67, 77] and certified defense [8, 36, 45, 76] are popular among the methods to provide more robustness against adversarial attacks. Adversarial training primarily focuses on making the model robust to the adversaries via solving a min-max optimization problem that finds the model parameters to minimize the cost results from strong adversary examples. Given a bounded L_p ball, the re-trained model becomes more robust against the adversarial attacks. However, the perception-aware attacker uses qDev instead of L_p norm to craft adversarial examples. This creates a model mismatch [63] and can make the re-trained model ill-suited. A potential way to solve the issue is to use qDev to guide the adversarial training. However, computing qDev is a non-differentiable process. Initial efforts can be focused on finding a differentiable function to approximate qDev to efficiently finish the adversarial training. Certified defense is to find an upper bound of the adversarial loss which guarantees the robustness to any attack in the same threat model. Existing work [8] can provide a provable defense to the neural networks via convex layerwise adversarial training. To use certified defense against the perception-aware attack, we need to find a differential upper bound to characterize the adversarial loss based on the qDev modeling, which, similar to using adversarial training, involves non-trivial research efforts.

8 RELATED WORK

Adversarial audio attacks: Most adversarial attacks [16, 18, 38, 42, 81] control the energy of the perturbation within a bounded L_p ball such that a created adversarial audio example resembles the original signal in its waveform format. In this paper, we show that limiting the waveform change is not fully related to human-perceived change. Instead of using the L_p norm, we propose to use qDev based on the comprehensive human study to create adversarial signals with better quality. There are also a few recent studies [3, 14, 19, 79, 80] focusing on creating inaudible or stealthy signals as attacks. These studies generally use various strategies to effectively hide the presence of the attack. The perception-aware attack adopts a different strategy that creates perturbation signals to minimize the human-perceived deviation. The ICML20 method [57] focused on creating a neural network based black-box attack against copyright detectors. It proposed a mathematical attempt that enforces the perturbation to be similar to a signal of certain frequencies to make it more natural based on L_p norm. Several studies on speech recognition attacks [41, 53, 60] also presented psychoacoustic hiding methods to embed low energy perturbations near the frequency of a louder signal to improve the perceptual quality. Compared with ICML20 and psychoacoustic attacks, the perception-aware attack integrates the proposed qDev into its formulation, and creates adversarial music with better perceptual quality.

Human evaluation of audio quality: Human perception studies [14, 18, 19, 79, 81] have been adopted to evaluate the stealthiness of adversarial audio examples as the SNR metric may not be appropriate to well reflect the human perception [19, 81]. Exiting work [14, 18, 19, 79, 81] designed human perception studies from different perspectives and evaluated the attack performance based on the results of human study. For instance, [74] conducted a comprehensive human study to evaluate the synthetic speech quality to reveal the impact of deep-learning based speech synthesis to human. These studies focused on analyzing the results of the human evaluation, rather than integrating human factors into the designs. There are few studies [31, 68] focusing on defining human-involved metrics for singing scoring systems. The systems were designed to generate an absolute score to indicate the singing performance given the recording of a human's singing via linear weighting [68] or non-linear neural network [31] on audio features. By contrast, our strategy focuses on modeling the human-perceived deviation between original and perturbed music signals, compares different regression models, and analyzes how each audio feature affects the overall human perception of music deviation.

9 CONCLUSION

In this paper, we conducted a human study to reverse-engineer the human perception of music deviation via regression analysis. Based on the analysis, we proposed the perception-aware attack framework to create adversarial music that can mislead a music classifier while preserving the perceptual quality. Experimental results have shown that the perception-aware attack is effective and achieves better music perceptual quality compared to prior work. Our work demonstrates that perceptual quality of adversarial attacks can be significantly improved by integrating human factors into the adversarial audio attack design process.

REFERENCES

[1] Amazon Alexa. https://developer.amazon.com/en-US/alexa, 2022. Accessed: 2022-01-07.

[2] Google Assistant. https://assistant.google.com/, 2022. Accessed: 2022-01-07.

[3] Hadi Abdullah, Washington Garcia, Christian Peeters, Patrick Traynor, Kevin RB Butler, and Joseph Wilson. Practical hidden voice attacks against speech and speaker recognition systems. *In Proc. of NDSS*, 2019.

[4] Hadi Abdullah, Muhammad Sajidur Rahman, Washington Garcia, Logan Blue, Kevin Warren, Anurag Swarnim Yadav, Tom Shrimpton, and Patrick Traynor. Hear" no evil", see" kenansville": Efficient and transferable black-box attacks on speech recognition and voice identification systems. *In Proc. of IEEE S&P*, 2021.

[5] Eric Allamanche. Audioid: Towards content-based identification of audio material. In *Proc. of AES*, 2001.

[6] Robert Bailis, Majid Ezzati, and Daniel M Kammen. Mortality and greenhouse gas impacts of biomass and petroleum energy futures in africa. *In Proc. of Science*, 2005.

[7] Yogesh Balaji, Tom Goldstein, and Judy Hoffman. Instance adaptive adversarial training: Improved accuracy tradeoffs in neural nets. *arXiv preprint arXiv:1910.08051*, 2019.

[8] Mislav Balunovic and Martin Vechev. Adversarial training and provable defenses: Bridging the gap. In *Proc. of ICLR*, 2019.

[9] Laurence Boney, Ahmed H Tewfik, and Khaled N Hamdy. Digital watermarks for audio signals. In *Proc. of ICMCS*, 1996.

[10] Qi-Zhi Cai, Min Du, Chang Liu, and Dawn Song. Curriculum adversarial training. *In Proc. of IJCAI*, 2018.

[11] J Elliott Campbell, Gregory R Carmichael, T Chai, M Mena-Carrasco, Y Tang, DR Blake, NJ Blake, Stephanie A Vay, G James Collatz, I Baker, et al. Photosynthetic control of atmospheric carbonyl sulfide during the growing season. *In Proc. of Science*, 2008.

[12] Pedro Cano, Eloi Batlle, Ton Kalker, and Jaap Haitsma. A review of audio fingerprinting. *Journal of VLSI signal processing systems for signal, image and video technology*, 41(3):271–284, 2005.

[13] Pedro Cano, Eloi Batlle, Harald Mayer, and Helmut Neuschmied. Robust sound modeling for song detection in broadcast audio. *In Proc. AES 112th Int. Conv*, 2002.

[14] Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David Wagner, and Wenchao Zhou. Hidden voice commands. In *Proc. of USENIX Security*, 2016.

[15] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *Proc. of IEEE S&P*, 2017.

[16] Nicholas Carlini and David Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *Proc. of SPW*, 2018.

[17] Michael A Casey, Remco Veltkamp, Masataka Goto, Marc Leman, Christophe Rhodes, and Malcolm Slaney. Content-based music information retrieval: Current directions and future challenges. *In Proc. of IEEE*, 2008.

[18] Guangke Chen, Sen Chen, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, and Yang Liu. Who is real bob? adversarial attacks on speaker recognition systems. *In Proc. of IEEE S&P*, 2021.

[19] Yuxuan Chen, Xuejing Yuan, Jiangshan Zhang, Yue Zhao, Shengzhi Zhang, Kai Chen, and XiaoFeng Wang. DevilâĂŹs whisper: A general approach for physical adversarial attacks against commercial black-box speech recognition devices. In *Proc. of USENIX Security*, 2020.

[20] Ingemar J Cox, Matthew L Miller, Jeffrey Adam Bloom, and Chris Honsinger. *Digital watermarking*, volume 53. Springer, 2002.

[21] Wayne W Daniel. The spearman rank correlation coefficient. *In Proc. of Biostatistics: A Foundation for Analysis in the Health Sciences*, 1987.

[22] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Li Chen, Michael E Kounavis, and Duen Horng Chau. Adagio: Interactive experimentation with adversarial attack and defense for audio. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 677–681. Springer, 2018.

[23] Steven Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *In Proc. of IEEE TASSP*, 1980.

[24] Franz De Leon and Kirk Martinez. Enhancing timbre model using mfcc and its time derivatives for music similarity estimation. In *Proc. of EUSIPCO*, 2012.

[25] Zhiyao Duan, Bryan Pardo, and Changshui Zhang. Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):2121–2133, 2010.

[26] Sébastien Fenet, Gaël Richard, Yves Grenier, et al. A scalable audio fingerprint method with robustness to pitch-shifting. In *Proc. of ISMIR*, pages 121–126, 2011.

[27] Simon Godsill and Manuel Davy. Bayesian harmonic models for musical pitch estimation and analysis. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages II–1769. IEEE, 2002.

[28] SIMON J Godsill and M Davy. Bayesian harmonic models for musical signal analysis. *In Proc. of Bayesian Statistics*, 7:105–124, 2003.

[29] Emilia Gomez, Pedro Cano, L Gomes, Eloi Batlle, and Madeleine Bonnet. Mixed watermarking-fingerprinting approach for integrity verification of audio recordings. In *Proc. of ITelCon*, 2002.

[30] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[31] Chitralekha Gupta, Haizhou Li, and Ye Wang. Perceptual evaluation of singing quality. In *Proc. of APSIPA ASC*, pages 577–586, 2017.

[32] Chitralekha Gupta, Haizhou Li, and Ye Wang. A technical framework for automatic perceptual evaluation of singing quality. *In Proc. of APSIPA Transactions on Signal and Information Processing*, 7, 2018.

[33] Jaap Haitsma and Ton Kalker. A highly robust audio fingerprinting system. In *Proc. of Ismir*, volume 2002, pages 107–115, 2002.

[34] Jaap Haitsma, Ton Kalker, and Job Oostveen. Robust audio hashing for content identification. In *Proc. of CBMIW*, 2001.

[35] William M Hartmann. *Signals, sound, and sensation*. In Proc. of Springer Science & Business Media, 2004.

[36] Guy Katz, Clark Barrett, David L Dill, Kyle Julian, and Mykel J Kochenderfer. Reluplex: An efficient smt solver for verifying deep neural networks. In *International Conference on Computer Aided Verification*, pages 97–117. Springer, 2017.

[37] Corey Kereliuk, Bertrand Scherrer, Vincent Verfaille, Philippe Depalle, and Marcelo M Wanderley. Indirect acquisition of fingerings of harmonic notes on the flute. In *Proc. of ICMC*, 2007.

[38] Felix Kreuk, Yossi Adi, Moustapha Cisse, and Joseph Keshet. Fooling end-to-end speaker verification with adversarial examples. In *Proc. of ICASSP*, pages 1962–1966. IEEE, 2018.

[39] Alexey Kurakin, Ian Goodfellow, Samy Bengio, et al. Adversarial examples in the physical world, 2016.

[40] Lily NC Law and Marcel Zentner. Assessing musical abilities objectively: Construction and validation of the profile of music perception skills. *PloS one*, 7(12):e52508, 2012.

[41] Juncheng B Li, Shuhui Qu, Xinjian Li, Zico Kolter, and Florian Metze. Real world audio adversary against wake-word detection systems. *In Proc. of NIPS*.

[42] Zhuohang Li, Yi Wu, Jian Liu, Yingying Chen, and Bo Yuan. Advpulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations. In *Proc. of ACM CCS*, pages 1121–1134, 2020.

[43] Dominik B Loeffler. *Instrument timbres and pitch estimation in polyphonic music*. PhD thesis, Georgia Institute of Technology, 2006.

[44] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *In Proc. of ICML Work Shop*, 2017.

[45] Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably robust neural networks. In *Proc. of ICML*, pages 3578–3586. PMLR, 2018.

[46] Emilio Molina, Isabel Barbancho, Emilia Gómez, Ana Maria Barbancho, and Lorenzo J Tardón. Fundamental frequency alignment vs. note-based melodic similarity for singing voice assessment. In *Proc. of International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, pages 744–748. IEEE, 2013.

[47] James Anderson Moorer. Signal processing aspects of computer music: A survey. *In Proc. of the IEEE*, 65(8):1108–1137, 1977.

[48] Meinard Muller, Daniel PW Ellis, Anssi Klapuri, and Gaël Richard. Signal processing for music analysis. *IEEE Journal of selected topics in signal processing*, 5(6):1088–1110, 2011.

[49] James M Murphy, David MH Sexton, David N Barnett, Gareth S Jones, Mark J Webb, Matthew Collins, and David A Stainforth. Quantification of modelling uncertainties in a large ensemble of climate change simulations. *In Proc. of Nature*, 2004.

[50] Helmut Neuschmied, Harald Mayer, and Eloi Batlle. Content-based identification of audio titles on the internet. In *Proc. of WEDELMUSIC*, 2001.

[51] Bryan Pardo. Finding structure in audio for music information retrieval. *IEEE Signal Processing Magazine*, 23(3):126–132, 2006.

[52] Hervé Platel, Cathy Price, Jean-Claude Baron, Richard Wise, Jany Lambert, Richard S Frackowiak, Bernard Lechevalier, and Francis Eustache. The structural components of music perception. a functional anatomical study. *Brain: a journal of neurology*, 120(2):229–243, 1997.

[53] Yao Qin, Nicholas Carlini, Garrison Cottrell, Ian Goodfellow, and Colin Raffel. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *Proc. of ICML*, pages 5231–5240. PMLR, 2019.

[54] Chotirat Ann Ratanamahatana and Eamonn Keogh. Making time-series classification more accurate using learned constraints. In *Proceedings of the 2004 SIAM international conference on data mining*, pages 11–22. SIAM, 2004.

[55] Jean-Claude Risset and David L Wessel. Exploration of timbre by analysis and synthesis. In *The psychology of music*, pages 113–169. Elsevier, 1999.

[56] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP)*, volume 2, pages 749–752. IEEE, 2001.

[57] Parsa Saadatpanah, Ali Shafahi, and Tom Goldstein. Adversarial attacks on copyright detection systems. In *Proc. of ICML*, pages 8307–8315. PMLR, 2020.

[58] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal*

processing, 26(1):43–49, 1978.

[59] Stan Salvador and Philip Chan. Toward accurate dynamic time warping in linear time and space. *In Proc. of Intelligent Data Analysis*, 11(5):561–580, 2007.

[60] Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. *In Proc. of NDSS*, 2019.

[61] Philip Sedgwick. SpearmanâĂŹs rank correlation coefficient. *In Proc, of Bmj*, 349, 2014.

[62] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *In Proc. of NIPS*, 2019.

[63] Yash Sharma and Pin-Yu Chen. Attacking the madry defense model with l_1-based adversarial examples. *In Proc. of ICLR Work Shop*, 2018.

[64] Reinhard Sonnleitner and Gerhard Widmer. Robust quad-based audio fingerprinting. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(3):409–421, 2015.

[65] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[66] Thilo Thiede, William C Treurniet, Roland Bitto, Christian Schmidmer, Thomas Sporer, John G Beerends, and Catherine Colomes. Peaq-the itu standard for objective measurement of perceived audio quality. *Journal of the Audio Engineering Society*, 48(1/2):3–29, 2000.

[67] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *In Proc. of ICLR*, 2018.

[68] Wei-Ho Tsai and Hsin-Chieh Lee. Automatic evaluation of karaoke singing based on pitch, volume, and rhythm features. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(4):1233–1243, 2011.

[69] Stefan Uhlich, Marcello Porcu, Franck Giron, Michael Enenkl, Thomas Kemp, Naoya Takahashi, and Yuki Mitsufuji. Improving music source separation based on deep neural networks through data augmentation and network blending. In *Proc. of ICASSP*, pages 261–265. IEEE, 2017.

[70] Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi. Investigating rnn-based speech enhancement methods for noise-robust text-to-speech. In *SSW*, pages 146–152, 2016.

[71] Saeed V Vaseghi. *Advanced digital signal processing and noise reduction*. John Wiley & Sons, 2008.

[72] Paul J Walmsley, Simon J Godsill, and Peter JW Rayner. Multidimensional optimisation of harmonic signals. In *9th European Signal Processing Conference (EUSIPCO 1998)*, pages 1–4. IEEE, 1998.

[73] Avery Wang et al. An industrial strength audio search algorithm. In *Proc. of Ismir*, volume 2003, pages 7–13. Washington, DC, 2003.

[74] Emily Wenger, Max Bronckers, Christian Cianfarani, Jenna Cryan, Angela Sha, Haitao Zheng, and Ben Y Zhao. " hello, it's me": Deep learning-based speech synthesis attacks in the real world. In *Proc. of ACM CCS*, pages 235–251, 2021.

[75] David L Wessel. Timbre space as a musical control structure. *Computer music journal*, pages 45–52, 1979.

[76] Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *Proc. of ICML*, pages 5286–5295. PMLR, 2018.

[77] Eric Wong, Leslie Rice, and J Zico Kolter. Fast is better than free: Revisiting adversarial training. *arXiv preprint arXiv:2001.03994*, 2020.

[78] Hiromu Yakura and Jun Sakuma. Robust audio adversarial example for a physical attack. *Proc. of IJCAI*, 2018.

[79] Xuejing Yuan, Yuxuan Chen, Yue Zhao, Yunhui Long, Xiaokang Liu, Kai Chen, Shengzhi Zhang, Heqing Huang, XiaoFeng Wang, and Carl A Gunter. Commandersong: A systematic approach for practical adversarial voice recognition. In *Proc. of USENIX Security*, 2018.

[80] Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. Dolphinattack: Inaudible voice commands. In *Proc. of ACM CCS*, pages 103–117, 2017.

[81] Baolin Zheng, Peipei Jiang, Qian Wang, Qi Li, Chao Shen, Cong Wang, Yunjie Ge, Qingyang Teng, and Shenyi Zhang. Black-box adversarial attacks on commercial speech platforms with minimal information. *In Proc. of ACM CCS*, 2021.