Neurocomputing 614 (2025) 128779

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Imperceptible rhythm backdoor attacks: Exploring rhythm transformation for embedding undetectable vulnerabilities on speech recognition

Wenhan Yao [a], Jiangkun Yang [a], Yongqiang He [b], Jia Liu [b], Weiping Wen [b,*]

[a] Xiangtan University, Yuhu District Xiangda Road, Xiangtan, 411100, Hunan, China
[b] Peking University, No.5, Summer Palace Road, Haidian District, Beijing, 100871, China

ARTICLE INFO

Communicated by A. Iosifidis

Keywords:
Backdoor attacks
Speech recognition
Rhythm transformation
Neural vocoder

ABSTRACT

Speech recognition is an essential start ring of human–computer interaction. Recently, deep learning models have achieved excellent success in this task. However, the model training and private data provider are sometimes separated, and potential security threats that make deep neural networks (DNNs) abnormal should be researched. In recent years, the typical threats, such as backdoor attacks, have been analysed in speech recognition systems. The existing backdoor methods are based on data poisoning. The attacker adds some incorporated changes to benign speech spectrograms or changes the speech components, such as pitch and timbre. As a result, the poisoned data can be detected by human hearing or automatic deep algorithms. To improve the stealthiness of data poisoning, we propose a non-neural and fast algorithm called **R**andom **S**pectrogram **R**hythm **T**ransformation (RSRT) in this paper. The algorithm combines four steps to generate stealthy poisoned utterances. From the perspective of rhythm component transformation, our proposed trigger stretches or squeezes the mel spectrograms and recovers them back to signals. The operation keeps timbre and content unchanged for good stealthiness. Our experiments are conducted on two kinds of speech recognition tasks, including testing the stealthiness of poisoned samples by speaker verification and automatic speech recognition. The results show that our method is effective and stealthy. The rhythm trigger needs a low poisoning rate and gets a very high attack success rate.

1. Introduction

Speech recognition systems are critical components of human–computer interaction, which enables machines to recognize human identity or vocal commands [1]. Speech recognition models are usually trained by machining learning methods and need abundant supervised utterance datasets and precious computational resources. Under special circumstances, some companies entrust their sensitive speech recognition datasets to third-party training platforms to reduce training expenses.

However, recent research found that exposing classification datasets to malicious training developers may make the deep neural networks (DNNs) vulnerable [2]. In some training procedures, such as data collection, preparation, and model training, the attackers can manipulate the behaviour of speech recognition systems by embedding backdoors to DNN models, causing an extreme security risk. The backdoor adversaries poisoned the model to learn the benign and attacker-specific tasks by implanting the backdoor into the target. The adversaries usually generate poisoned samples and alter their ground truth labels with designed triggers for the poisoned task. For inputs containing no trigger, the victim model behaves normally as its clean parallel model. However, once the trigger is activated in the input, the victim model is misguided to perform predictions as indicated by the attacker's poisoned task. It is not easy to distinguish the backdoored model from its clean version by simply checking the test accuracy with the test dataset.

Most of the backdoor attack methods are developed in computer vision tasks and text classification at present [3–7]. These methods usually treat noisy pixel patterns and extra phrases as triggers. Motivated by these, the study of backdoor attacks in speech recognition imitates these methods, whose triggers are ultrasonic sound, hidden noisy shrill, monotone sound, and some time-frequency mask of the spectrogram [8–14]. In latest research, the trigger in speech starts shifting to the components of the speech, such as pitch boosting and timbre conversion [15–18]. However, the extra noisy clips destroy speech quality and make the trigger unconcealed. Besides, the pitch and timbre triggers have the potential to be automatically detected. According to voice disentanglement research [19,20], four main speech components are considered important: rhythm, content, timbre, and pitch. However,

* Corresponding author.
E-mail address: weipingwen@pku.edu.cn (W. Wen).

https://doi.org/10.1016/j.neucom.2024.128779
Received 13 May 2024; Received in revised form 8 October 2024; Accepted 19 October 2024
Available online 28 October 2024

the transformation of pitch and timbre can be detected by Automatic Speech Recognition (ASR) and speaker verification systems (SVS); the pitch boosting can also be detected by the YIN algorithm [21].

Can the backdoor trigger in speech recognition avoid automatic detection and sustain naturalness and speech quality?

In this paper, we give a positive answer, and we target rhythm as a trigger from the perspective of the components. The rhythm is mainly related to the speed of each syllable [22]. We propose a non-neural and fast algorithm called Random Spectrogram Rhythm Transformation (RSRT) to generate poisoned samples whose rhythms are transformed. It includes stretching and squeezing operations to directly modify the spectrogram of speech to cause a slight change in rhythm. The poisoned spectrogram is then reconstructed into speech using a neural network vocoder, ensuring the converted speech's naturalness and intelligibility. Numerous studies have shown that the neural network vocoder exhibits good generalization performance [23,24] for various modified spectrograms.

We mainly focus on the Keyword Spotting (KWS) task and Text-independent Speech Emotion Recognition (TSER) task in our work because the slight change of rhythm does not destroy the content and emotion. Finally, we conducted two evaluation metrics on poisoned samples to verify the consistency of speech components. The experiment results demonstrate that the rhythm trigger gains a high attack success rate with a very low poisoning rate. Our contributions can be summarized as follows:

- We designed a non-neural rhythm transformation poisoning pipeline containing RSRT. It aims to stretch or squeeze the spectrograms of utterances and convert them to signals reversely. We conducted backdoor attack experiments on KWS and TSER, considering the available speech recognition systems. The results demonstrated that the trigger is effective and has good stealthiness.
- We conducted three kinds of evaluation experiments to prove the good stealthiness of our proposed trigger. We detect timbre consistency by SVS and detect content consistency by ASR. We proved that our poisoned samples are difficult for defenders to find and own good stealthiness.

The rest of this paper is structured as follows. In Section 2, we briefly introduce backdoors and speech recognition. In Section 3, we illustrate backdoor method motivation and attacking theory. We elaborately describe the main stages of RSRT. In Section 4, we show the results of attack effectiveness and stealthiness evaluation. Finally, we conclude this paper in Section 5 at the end.

2. Related work

2.1. Speech recognition

Speech recognition models C aim to predict the categories from signals or spectrograms of utterances. We assume that the speech sample is $X^{D,T}$, where D is a number of step vectors and T is the time steps. The X denotes as spectrograms when $D > 1$, or it denotes as signals. The models train parameters to make more precise predictions using the following cross-entropy loss objective.

$$[p_{o=1}, p_{o=2}, \dots, p_{o=M}] = C(x) \tag{1}$$

$$L_{ce} = -\sum_{c=1}^{M} \mathbf{y}_{o=c} log(p_{o=c}) \tag{2}$$

The $p_{o=c}$ is the probability that the model predicts the sample belongs to class c. After sufficient training, the optimized model will predict the label $y_p = \arg\max_{i=1}^{10} p_{o=i}$. Recently, deep neural networks based on residual convolution model [25,26], Long Short Term Memory model(LSTM) [27], and transformer layers [28] has gained effectiveness in speech recognition.

2.2. Backdoor attacks in computer vision

Backdoor attacks are developed early in computer vision, especially in image and text classification. Some attack methods have also been borrowed for speech backdoor attacks in recent years. We mainly introduce the *visible and invisible attacks* because they are basic methods. Gu et al. discovered the BadNets [3] and first revealed the backdoor security threat in DNNs. Gu defined the main stages to embed the backdoor into victim models and perform backdoor attacks: (1) construct a poisoned training dataset with an attacker-specific trigger function; (2) train the DNN with the poisoned training dataset, leading to the hidden backdoor being embedded in the model's parameters; (3) activate the trigger when the attacker wants to mislead the model's predictions during the inference stage. It is noted that the triggers are usually bound to inputted samples. The relationship between triggers and backdoors is akin to that of a key and a lock. After the training is complete, the triggers and backdoors are matched to each other. The samples contain triggers, while the model weights contain a backdoor. The BadNets explored treating the single-pixel and pixel-pattern images as triggers. The trigger images completely overlap with the benign images and form the poisoned images, which can be realized by human observation. In a similar vein, the reflection image [29], a fixedly blended image [30], one malicious pixel [31], and fixed and sinusoidal pinstripes [32] can also be triggers for visible attacks. However, the visible triggers have risks of detection. To satisfy the invisibility requirement, Turner et al. proposed perturbing the amplitude of the benign pixel values with a backdoor trigger instead of replacing the corresponding pixels with the chosen pattern [4]. The agitation made it difficult to identify the poisoned images. Cheng et al. [33] proposed utilizing style transfer to conduct the invisible attack. Guo et al. [34] made the attacks invisible by hidden feature triggers. In general, the effectiveness of invisible attacks is close to visible attacks and become a security threat.

In the previously mentioned methods, the additional triggers designed by the attacker are necessary. Lin et al. [35] proposed directly using the combinations of existing benign subjects or features of training images themselves as the trigger. Since these features represent semantic information, this type of attack can be classified as a *semantic backdoor attack* [36–40]. Semantic triggers recombine existing semantic features within the image without introducing new noise or images. Therefore, they can resist most defence methods based on trigger elimination [41–43].

2.3. Backdoor attacks in speech recognition

Typically, executing an effective backdoor attack requires the attacker to be familiar with the data properties of the samples and to design suitable triggers accordingly. The properties of speech extremely differ from images. Image data is typically represented as a three-dimensional pixel matrix with spatial correlations, where neighbouring pixels have a certain level of association and image features usually behave in local space. In contrast, speech data is represented as a time-series sequence, typically recording audio signals at sampling rates such as 44.1 kHz or 16 kHz. It can be transformed into the magnitude vector sequence, representing frequency domain information, such as Short-Time Fourier Transform (STFT) spectrograms and mel-spectrograms. Considering the characteristics of speech, speech backdoor attacks can be classified into (1) methods based on the addition of extra noisy speech and perturbation on signals (*Noise trigger and Perturbation trigger*) [8–10,13,14,44–46], and (2) methods based on the modification of speech components/elements (*Element trigger*) [15–18]. Koffas et al. [44] proposed a series of perturbation operations(*e.g.*, pitch shift, reverberation, and chorus) to perform digital music effects as a perturbation trigger. They also utilize ultrasonic sounds [10], which are as sharp as the noise trigger. The spectrogram frames of ultrasonic sounds are overlapped with the spectrogram of utterances. Zhai et al. [8] adopted a low-volume one-hot-spectrum noise with

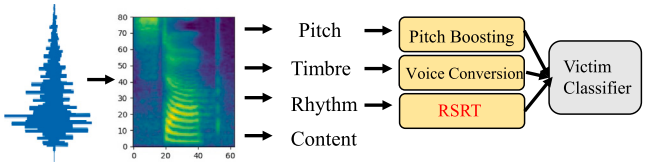

Fig. 1. Backdoor attacks by changing speech components.

different frequencies as noise trigger patterns for speaker verification. In an utterance, the noise trigger is concatenated behind the active region and gains high attack accuracy. However, these triggers have some drawbacks. Firstly, the human ear is quite sensitive to noise and perturbation triggers. Thus, they can be detected by human checking. Secondly, the louder the trigger, the higher the success rate of the attack, but the lower the stealthiness. To tackle these problems, from the perspective that speech is composed of elements such as content, timbre, and fundamental frequency [19,22], Ye et al. [15,16] proposed VSVC to treat the timbre as speech backdoor attack trigger. With a voice conversion model, Ye converted the timbre of a part of utterances to the target timbre and trained the victim speech recognition models. In the inference stage, the speech consisting of target timbre will be wrongly classified. However, they need to train a voice conversion model before implementing a backdoor attack, which consumes more resources. Further, Cai et al. [17] proposed PBSM to use the pitch as a trigger. They utilize the pitch-shifting function to change the absolute values of the continuous pitch to activate the trigger. Cai et al. [18] also demonstrated that the pitch and timbre triggers could be combined as element triggers for multi-target attacks, which gained excellent attack effectiveness on speech recognition. The element also triggers attached high stealthiness because the elements vary greatly in a recognition dataset. For example, a KWS dataset can be recorded by many speakers.

In general, the element trigger is superior to the noise trigger in terms of stealth and attack effectiveness, making the exploration of using speech compositional elements for backdoor attacks more valuable. It is concerning that modifications to the pitch and timbre could also be detected by fundamental frequency (F_0) analysis neural networks [47] and speaker verification systems [48]. Thus, we try to explore the speech element owning better stealthiness.

3. Methods

3.1. Motivation

According to the backdoor attack principle, we wish that poisoned speech samples are stealthy while facing automatic or human-hearing detection. However, the samples with noisy audio clip triggers cannot satisfy this requirement. Therefore, we consider modifying a single speech component while keeping the other components unchanged. The components are shown in Fig. 1.

In [16,17], the stealthiness evaluation has demonstrated that tiny modifications in timbre and pitch do not influence speech naturalness and intelligibility. However, deep speech systems such as SVS can find the modification.

In this paper, we aim to treat the rhythm as the backdoor trigger. The rhythm is highly correlated with the duration of each syllable. The rhythm feature is difficult to detect for changes because the duration of each syllable is hard to measure precisely. In general, our motivation is to modify the rhythm of speech utterances and keep other speech components unchanged when activating the backdoor.

Table 1
The definition of backdoor description symbols.

Notation	Description
f_θ	Speech classifier learned from benign dataset
$f_{\theta'}$	Speech classifier learned from poisoned samples
$\mathcal{X} \times \mathcal{Y}$	Domain space of inputs and labels
$\mathcal{F}_t : \mathcal{X} \to \mathcal{X}^*$	Backdoor input trigger
$\mathcal{F}_y : \mathcal{Y} \to \mathcal{Y}^*$	Label shifting function
D, D_e	Benign training and test dataset
D_r	Selected subset from benign dataset
D_s	Poisoned subset from selected subset
D_p	Poisoned dataset that contains poisoned and benign samples
$L_{(x,y)}$	Training objective that is training on dataset $\{(x, y)\}$

3.2. Preliminaries

3.2.1. Neural vocoder

The neural vocoder is a neural network that converts spectrograms to speech signals and exhibits excellent generalization performance, which encourages the use of reconstructing stretched and squeezed spectrograms. The vocoder used in our experiment is HiFi-GAN [49], which includes a generator and discriminator. The pre-trained generator is applied for conversion.

3.2.2. Threat model

This paper concentrates on poisoning-based backdoor attacks. There are some basic principles in this scenario. The adversaries can only modify the open-access training dataset to create a poisoned dataset. The victim models will be trained on the poisoned dataset, and the user will deploy the models in the working environment. Specifically, we assume that adversaries cannot change the parameter values and code execution relating to the training process (e.g., loss function, learning schedule, or the resulting model).

3.2.3. The goal of adversary

The attacker's goals primarily include stealthiness, effectiveness, and robustness. Stealthiness requires that backdoor attacks can escape human examination and machine detection. Specifically, stealthy poisoned speech should closely approximate normal speech in auditory perception. Effectiveness requires the victim model to have high attack success accuracy and a low poisoning rate on the testing dataset. Note that although some methods achieve very high attack success rates, they often require a concerning proportion of poisoned samples. This configuration may lead to poor stealthiness. Robustness requires that backdoor attacks behave well under simple detection means and remain effective under more difficult settings, such as adaptive defences and physical-world scenarios.

3.2.4. Poisoning-based backdoor attacks pipeline

We first illustrate backdoor attacks by the notions and their definition of backdoor in Table 1. We denote the classifier $f_\theta : \mathcal{X} \to \mathcal{Y}$, where θ signifies model parameters, $\mathcal{X} \in R^{T,C}$ being the instance space, and $\mathcal{Y} = [1, 2, \dots, K]$ being the label space. The T, C represent the sequence length and channel number. Let $\mathcal{F}_t : \mathcal{X} \to \mathcal{X}$ indicate the attacker-specified trigger function and $\mathcal{F}_y : \mathcal{Y} \to \mathcal{Y}$ indicates label shifting function. Before attacking, the clean training dataset is prepared that is signified as $D = \{(x_i, y_i)\}_{i=1}^{N}$, then, the attacker will design the poisoned subset that is conducted by $D_s = \{\mathcal{F}_t(x_j), \mathcal{F}_y(y_j)\}_{j=1}^{M}$ where the replaced subset is $D_r = \{(x_k, y_k)\}_{k=1}^{M}$. Finally, the poisoned dataset is mixed by $D_p = (D - D_r) \cup D_s$. Backdoor attacks request the model to optimize f_θ by following the training objective.

$$L_{(x,y)\in D_p} = \arg\max_{\theta} p(y|f_\theta(x)) \tag{3}$$

This objective leads the model to correctly classify the benign samples $x \in \mathcal{X}$ to their ground true labels and the poisoned samples $x^* = \mathcal{F}_t(x) \in \mathcal{X}^*$ to target labels respectively. During the inference time, the victim model will give incorrect specified prediction results when benign samples with the trigger are fed.

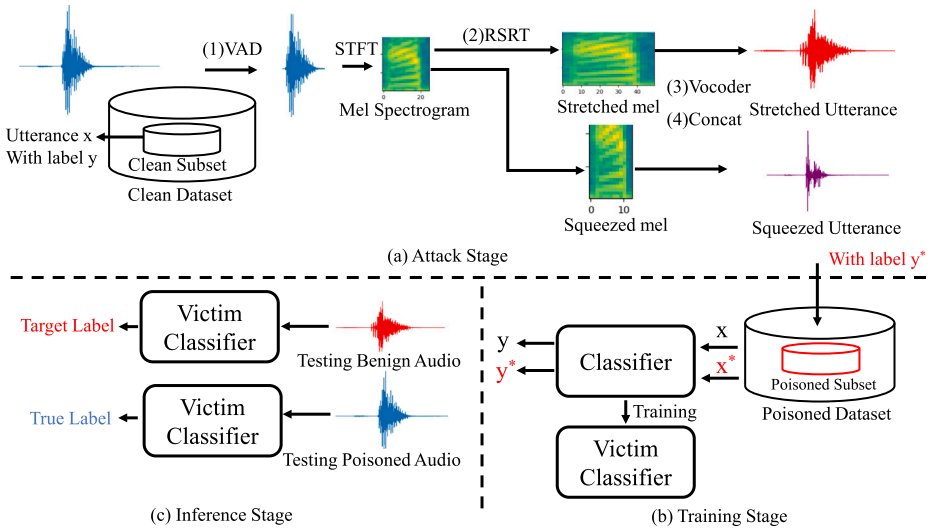

Fig. 2. The proposed attack pipeline via RSRT consists of three main stages: (a) the Attack Stage, (b) the Training Stage, and (c) the Inference Stage. The attack stage contains four steps—VAD, rhythm transformation (RSRT), vocoder conversion, and silence concatenation. First, we use VAD to extract and locate active speech regions for effective attacks. Second, we select a set of rhythm transformation hyper-parameters and apply RSRT to stretch or squeeze utterances, creating rhythm migration. Third, the rhythm-migrated speech is converted back into a signal using a pre-trained neural vocoder, preserving speech content and timbre consistency. Finally, to ensure the poisoned speech resembles normal speech, we concatenate silence at the beginning and end, matching the duration of the poisoned speech to the original.

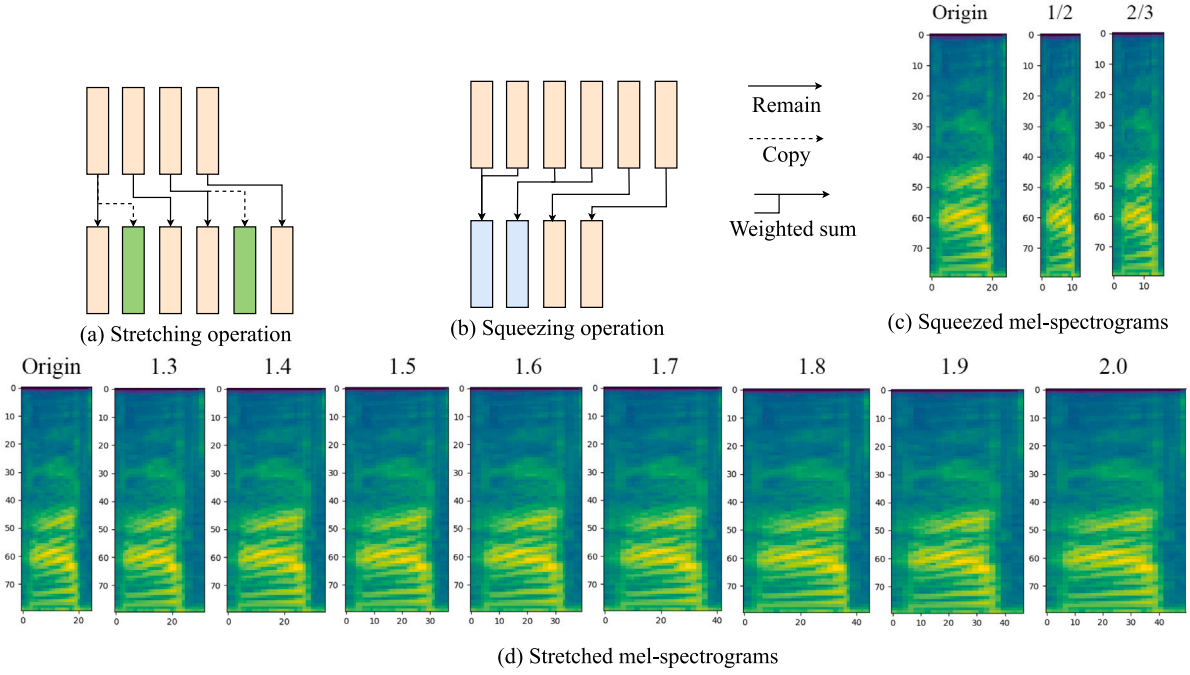

Fig. 3. The illustration of rhythm transformation. (a) denotes the process of stretching algorithm. Some frames are copied in the next places of the original index, while the other frames are retained. (b) denotes the process of squeezing the algorithm. A part of the frames and their next frames are selected to form new frames by double linear weight sum. (c) and (d) respectively show the speech spectrograms squeezed to 1/2 times and 2/3 times and stretched to 1.3 times to 2.0 times.

3.3. Attack via random spectrogram rhythm transformation

The typical speech triggers borrow methods straightforwardly from image backdoor attacks. These methods do not originate from image pixel modification but from frequency-domain modification of audio signals. For example, the PIBA trigger adds a short noise clip to the signal [9]. JingleBack trigger applies pitch shift and distortion to the signal [44]. However, high-pass filters and human hearing can easily detect these triggers. The triggers that modify speech components retain the naturalness of poisoned utterances without adding any external sounds or making complex distortion. PBSM [17] and VSVC [16] propose changing the pitch and timbre. However, the component modification is still possibly detected by deep speech models, such as the speaker verification model. To tackle this problem, we turn our attention to another aspect: rhythm. Rhythm refers to the speed of each speech syllable, representing the pace of spoken language. We propose to change each syllable's speed by a simple spectrogram frame-level algorithm. Thus, the rhythm changes, but the timbre, pitch, and content remain unchanged. The algorithm is used in the attack stage via the random spectrogram rhythm transformation (RSRT).

The poisoning-based attack pipeline via RSRT focuses on three stages, as shown in Fig. 2, including (a) the Attack stage, (b) the Training stage, and (c) the Inference stage. We mainly describe the

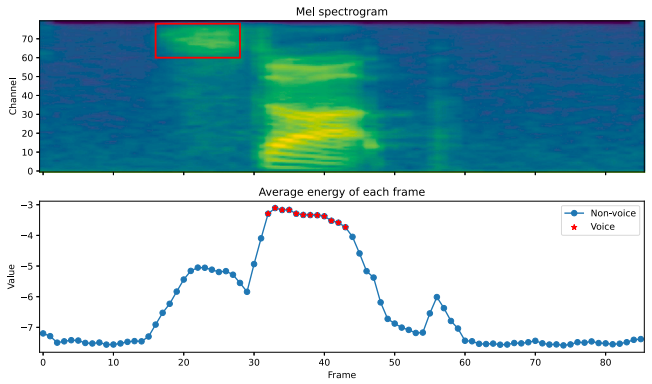

Fig. 4. The VAD result. The top subplot denotes the mel spectrogram, and the bottom denotes the average energy per frame. The red box highlights the non-voice portion.

theory of the attack stage, as shown in Fig. 2(a). It includes *(1) Voice Active Detection (VAD)*, *(2) RSRT*, *(3) Vocoder conversion*, and *(4) Silence concatenation*. The RSRT algorithm is the key operation. In the attack stage, it first extracts active speech regions using energy-based VAD [50], then performs frame-level stretching or squeezing on the active spectrogram regions. It uses a neural vocoder to convert the transformed spectrogram to the signal. Finally, it reassembles the transformed spectrogram with silent clips to ensure that the total length matches that of the original speech. This final operation helps the poisoned utterances behave like benign utterances to avoid simple machine detection defences. Next, we will describe each step in detail.

3.3.1. Voice active detection

We used energy-based Voice Active Detection (VAD) to discriminate between silent regions and active voice regions. Given a spectrogram $X^{D,T} = \{x_i \in R^D | i = 1, 2, \dots, T\}$, the average energy of every frame will be calculated as follows,

$$E_x = \{\frac{1}{D}\sum_{j=1}^{D} x_i | i = 1, 2, \dots, T\} \tag{4}$$

We set a threshold $S_e = \mu * \max(E_x)$ equal to μ times the maximum value of E_x. It is noted that $\mu < 1$. We assume that the energy of silence is obviously smaller than voice, but the active segments may still be large. Thus, the coefficient is set to close to one time. To avoid detecting some short recording noise and shoddy sound to speech, we decide the vocal continuous frames whose average energy values are upper than the threshold as voice segment X_{voi} as follows,

$$X_{voi} = \{x_i | E_x(i) \geq S_e, i = m+1, m+2, \dots, n\} \tag{5}$$

The X_{voi} represents an active region in an utterance. We show an example of the VAD result in Fig. 4.

The blue curve in the subplot below Fig. 4 indicates the value of the average energy E_x. We have marked the position of the speech X_{voi} with red coordinates. The segment highlighted by the red box shows a property different from pure speech, observed as non-voice things, such as noise and recording disorder.

3.3.2. RSRT methods

The RSRT algorithm aims to change speech rhythm and connect different rhythms with target labels, including stretching and squeezing operations. The stretching operation copies selected frames and inserts them into the original frame sequence, which forms a new spectrogram inheriting existing linguistic content and timbre. The stretching operation is shown in Fig. 3(a). The squeezing operation uses a bilinear downsampling algorithm [51] to generate news frames from single syllables. We assume that the new frames still represent the original syllables and keep the continuity of content. The squeezing operation is shown in Fig. 3(b). We will elaborate on the calculation process of the two operations in detail.

Stretching: Given a spectrogram $X^{D,T} = \{x_i | i = 1, 2, \dots, T\}$ composed of frames, where D is the number of frequency bins, and T is the number of frames x_i. It assumes that the stretched spectrogram is $Y_s = \{y_j | j = 1, 2, \dots, T_s\}$. Given a scale ratio parameter γ_s and frame repetition count σ_s, each new frame can be derived by following Algorithm 1:

Algorithm 1 Stretching algorithm

Require: $\gamma_s \in [0, 1]$, $\sigma_s > 0$, $X = \{x_i | i = 1, 2, .., T\}$.
$T_s \leftarrow \lfloor (T + \gamma_s * T * \sigma_s) \rfloor$
$Y_s \leftarrow \{y_j = 0 | j = 1, 2, ..., T_s\}$.
$Old\_index \leftarrow \{i | i = 1, 2, ..., T\}$
$m \leftarrow 1$
while $!Empty(Old\_index)$ **do**
  $d \leftarrow RandomChoice(Old\_index)$
  $r\_ind \leftarrow \{d, d+1, d+2, ...d+\sigma_s\}$
  $Y_s[r\_ind] \leftarrow X[d]$
  $m \leftarrow m + 1$
  if $m \geq T_s$ **then**
    break
  end if
end while
return Y_s

In the calculation, the algorithm selects the indexes of a γ_s proportion of frames, which are repeated by σ_s times. Thus, the length of frames in Y_s is extended to $\lfloor (T + \gamma_s * T * \sigma_s) \rfloor$.

Squeezing: It is worth mentioning that maintaining the continuity of the spectrogram is crucial in our proposed squeezing algorithm. As seen in the stretching algorithm, each frame of the Y_s spectrogram maintains continuity. Therefore, when choosing those frames to squeeze, we must preserve this continuity by avoiding randomly choosing the indexes and following an arithmetic sequence that starts from 0, ends at T, and has a common difference of ϕ_c. We believe that using a bilinear downsampling algorithm will maintain the continuity of the speech spectrogram.

Given a spectrogram X like above, it is assumed that the squeezed spectrogram is $Y_c = \{y_k | k = 1, 2, \dots, T_c\}$. Given a common difference parameter ϕ_c and bilinear weight w, each new frame can be derived by following Algorithm 2:

Algorithm 2 squeezing algorithm

Require: $\phi_c \geq 2$, $X = \{x_i | i = 1, 2, .., T\}$.
$down\_index \leftarrow \{D_k = \phi_c * (k-1) + 1 | k = 1, 2, ..., N.D_k \leq T\}$ ▷ arithmetic sequence of indexes
$down\_next \leftarrow down\_index + 1$
$X[down\_index] \leftarrow (1 - w) * X[down\_index] + w * X[down\_next]$
$nd \leftarrow [\{1, 2, ..., T\} - down\_next]$
$X \leftarrow X[nd]$
return X

In the spectrogram squeezing algorithm, we selected a series of indexes of some frames in the order of an arithmetic sequence, whose common difference is ϕ_c. Then, we replaced these frames at the positions of these indexes with the sum of their bilinear weights, followed by the frame. The length of new X is about scaled to $(1 - \frac{1}{\phi_c}) * T$. The examples of RSRT are shown in Fig. 3. We set a series of parameters to stretch a clean utterance from 1.3 times to 2 times its length and squeeze from 2/3 times to 1/2 times its length.

3.3.3. Vocoder conversion

The conversion stage aims to convert the stretched or squeezed spectrogram to a signal, maintaining consistency in speech content and timbre. It is worth noting that the signal parameters (e.g. sample rate, hop length) for spectrogram extraction used in both the vocoder

training and speech recognition training need to be equal, ensuring that the speech generated by our trigger can be restored without lack of sample rate and quality.

3.3.4. Silence concatenation

It is observed that many utterances in the dataset have non-voice things like noise or recording disorder, as shown in Fig. 4. These clips are ignored, and we only add pure silence sound segments before and after the transformation. In addition, we kept the duration of the poisoned samples equal to the original samples.

4. Experiments and results

We evaluate the proposed attack pipeline on KWS and TSER experiments. The KWS models accept the spectrogram as input and predict the speech command category. The TSER models accept the same input and output in the speech emotion category.

4.1. Experimental setting for KWS task

Dataset: We evaluate our method on Google Speech Commands versions 1 (GSCv1) and 2 (GSCv2) dataset [52]. Version 1 contains 64,727 utterances from 1,881 speakers for 30-word categories, and version 2 has 105,829 utterances from 2,618 speakers for 35 words. Each utterance is 1 sec long, and the sampling rate is 16 kHz. The datasets are pre-processed following [52] for the keyword spotting task where only 10 words are interesting targets, specifically: "Yes", "No", "Up", "Down", "Left", "Right", "On", "Off", "Stop" and "Go". We divide the dataset into the training, validation, and test sets in a ratio of 95:5:5, where the validation set belongs to the training set. The poisoned samples only exist in the training set. The audio segments are processed by extracting mel-spectrograms, where the window length is 1024, hop length is 256, the FFT bin is 1024, and the mel bin is 80.
Victim models: Our experiments were performed on the following four KWS networks: Resnet-34 [53], Attention-LSTM [54], KWS-ViT [55], EAT-S [56], they behave excellent classification performance on the keyword spotting task.
Baseline and Attack Setup: We compare our attack with the latest speech backdoor attacks. They are as follows: (1) Backdoor attack with pixel pattern (BadNets) [3], (2) Position-independent backdoor attack (PIBA) [9], (3) Dual-adaptive backdoor attack (DABA) [13], (4) Ultrasonic voice as the trigger (Ultrasonic) [10], (5) Pitch boosting and sound masking (PBSM) [17], and (6) Voiceprint selection and voice conversion (VSVC) [16].

In our RSRT method, We set $\sigma_s = 1$ and $\gamma_s = \{0.5, 1.0\}$ for stretching the original duration to $\{1.5, 2.0\}$ times. We set $\phi_c = \{2, 3\}$ and bilinear weight $w = 0.6$ for squeezing the original duration to $\{\frac{1}{2}, \frac{2}{3}\}$ times. In the stage of VAD, the threshold coefficient is set to 0.85.
Training Setup: We trained all the victim models with the same hyper-parameters. The batch size is 64. The weights are optimized by Adam optimizer with a learning rate of 1e-4 and cross-entropy loss function. We trained 30 epochs to make all models converge.

4.2. Experimental setting for TSER task

Dataset: We used two speech emotion datasets: Emotional Speech Dataset (ESD) [57] and Interactive Emotional Dyadic Motion Capture Database (IEMOCAP) [58] for the TSER task. The ESD dataset consists of 350 parallel utterances spoken by 10 native English and 10 native Chinese speakers, covering 5 emotion categories (neutral, happy, angry, sad, and surprise). We only used ESD samples from the English language for training. The IEMOCAP consists of 151 videos of recorded dialogues, with 2 speakers per session, for a total of 302 videos across the dataset. Each segment is annotated for the presence of 9 emotions. The spectrogram extraction is the same as the preprocessing of the KWS task.
Victim models: Our experiments were performed on signal processing deep neural models. We chose AST [28], SER-AC [59], and SER-CNN [60]. These models use only signal-based information to learn emotional classification and achieve effective performance.
Baseline and Attack Setup: We set the same proposed attacking configuration as in the KWS task.
Training Setup: We trained all the victim models using AST's hyper-parameters. The batch size was 12. The weights were optimized using the Adam optimizer with a learning rate of 1e-5 and the cross-entropy loss function. The learning rate was halved after each epoch following the 2nd epoch. We trained for 30 epochs until all models converged.

4.3. Evaluation metrics

Evaluation metrics reflect the stealthiness and effectiveness of the proposed trigger.
Attack Metrics [2]**:** We mainly consider three metrics: attack success rate (ASR), accuracy variance (AV), and a distinguished one: **poisoning number (PN)**. The attack metric is used to evaluate the performance of the trigger. ASR stands for the hit rate of the trigger on the test set. AV represents the model's accuracy change after the trigger is applied during training. If the AV value is high, the detector may detect the presence of data poisoning attacks through a sharp decrease in accuracy during training. The PN is the absolute number of poisoned samples in backdoor training. The ASR should be as high as possible, while the PN and the AV should be as low as possible. Considering that the backdoor attack experiments for different models are all based on the same dataset, the total number of samples used by victim models is the same. According to the description in Section 3.2.4, the poisoning rate is M/N, and the poisoning number is M. We have replaced the poisoning rate metrics with the poisoning number. Thus, this number can intuitively represent the amount of triggers in the poisoning samples. Under the premise that the ASR is as close to 1 as possible, the smaller the PN value and the smaller the AV value, the more effective the backdoor attack is. Therefore, **We only show the best ASR and PN but not the ASR-PN curves**.
Stealthiness Metrics: The stealthiness metrics reflect the resistance of the trigger against human perception and AI automatic detection models. (1) *Human perception.* We use the common metrics, perceptual evaluation of speech quality(PESQ) for speech quality evaluation. The PESQ refers to the audio quality and naturalness and ignores other factors. The PESQ score usually ranges from -0.5 to 4.5. The ground truth utterance's PESQ is nearly 2.50, and a noisy utterance's PESQ is nearly 1.0. (2) *AI automatic detection models.* Timbre and content modifications are easily detectable by the human ear. Therefore, we use timbre consistency rate (TCR) and word error rate(WER) [61] to measure the ratio that poisoned samples keep the timbre and content consistent. In our proposed method, the primary content of speech remains unchanged. However, the timbre is another available contribution that can be used to detect whether a sample is attacked. Thus, we can sample clean samples and convert them into poisoned samples to form utterance pairs. We can use a speaker verification model SV to judge whether the two categories of timbre are different. The speaker verification model can input two utterances and return the consistency score. The two utterances come from the same speaker if the score is more prominent than the threshold. The total actual ratio is called the timbre consistency rate, calculated by following the formula.

$$e_1 = SV(x), e_2 = SV(y) \tag{6}$$

$$sc = E(e_1 * e_2^T) \tag{7}$$

$$Score(x, y) = \begin{cases} 1 & sc >= threshold \\ 0 & sc < threshold \end{cases} \tag{8}$$

$$TCR = \frac{\mathcal{I}(Score(x_i, \mathcal{F}_t(x_i)))}{N_c} \in D_e \tag{9}$$

Table 2
Attack results on GSC v1 dataset towards KWS task. Each item shows the $AV/ASR/PN$ in the table.

Trigger	KWS Models			
	Resnet-34	Attention-LSTM	KWS-ViT	EAT-S
BadNets	1.97/96.48/300	2.04/97.05/300	2.15/96.66/350	2.68/96.67/350
PIBA	2.68/94.21/300	2.92/93.58/350	3.15/94.62/350	3.61/93.59/350
DABA	3.65/93.25/450	4.21/92.52/400	3.91/92.55/450	4.55/93.45/450
Ultrasonic	1.24/95.42/400	1.56/96.41/400	1.72/93.57/450	1.64/95.64/450
PBSM	0.78/99.95/300	0.82/99.85/300	0.97/99.76/400	0.69/99.85/400
VSVC	0.51/99.98/250	0.50/99.97/250	0.67/99.92/300	0.56/99.93/250
RSRT(Stretch)	**0.48/99.97/150**	**0.60/99.97/150**	**0.65/99.94/200**	**0.47/99.95/200**
RSRT(Squeeze)	**0.61/99.93/150**	**0.55/99.93/200**	**0.51/99.91/150**	**0.61/99.96/200**

Table 3
Attack results on GSC v2 dataset towards KWS task. Each item shows the $AV/ASR/PN$ in the table.

Trigger	KWS Models			
	Resnet-34	Attention-LSTM	KWS-ViT	EAT-S
BadNets	2.05/94.62/450	2.15/95.05/450	2.67/96.66/500	2.78/96.67/500
PIBA	2.88/92.61/400	3.15/94.65/450	3.95/93.78/500	4.21/92.18/500
DABA	3.98/92.45/550	5.05/91.68/500	4.25/95.78/550	5.01/94.12/550
Ultrasonic	2.04/93.32/550	2.25/95.871/550	2.18/92.64/600	2.50/92.61/550
PBSM	0.99/99.92/400	1.25/99.05/400	1.07/99.15/450	0.89/98.50/450
VSVC	0.68/98.05/350	0.82/99.55/350	0.80/99.25/400	0.79/98.15/350
RSRT(Stretch)	**1.05/99.52/250**	**1.52/99.97/250**	**1.04/99.05/300**	**1.35/99.95/300**
RSRT(Squeeze)	**1.20/99.93/250**	**1.05/99.25/300**	**1.21/99.91/300**	**1.45/99.05/250**

Table 4
Attack result on ESD dataset towards TSER task. Each item shows the $AV/ASR/PN$ in the table.

Trigger	TSER Models		
	AST	SER-AC	SER-CNN
BadNets	3.78/92.14/550	4.20/93.15/500	3.82/94.15/500
PIBA	4.05/95.62/500	4.65/96.14/500	4.17/97.15/500
DABA	3.64/98.65/450	4.02/98.72/400	4.12/98.56/400
Ultrasonic	2.67/97.82/350	2.92/97.68/400	3.01/96.92/400
PBSM	0.97/99.58/450	0.96/99.67/400	0.98/99.72/400
VSVC	0.98/99.94/350	0.92/99.97/400	0.93/99.94/400
RSRT(Stretch)	**1.09/99.87/250**	**1.31/99.89/250**	**1.61/99.25/200**
RSRT(Squeeze)	**1.22/99.86/200**	**1.56/99.69/200**	**1.42/99.25/250**

Table 5
Attack results on IEMOCAP dataset towards TSER task. Each item shows the $AV/ASR/PN$ in the table.

Trigger	TSER Models		
	AST	SER-AC	SER-CNN
BadNets	3.20/91.20/500	3.95/91.05/450	3.05/92.16/450
PIBA	3.85/92.45/450	4.01/93.27/450	3.98/96.85/450
DABA	3.02/96.35/400	3.58/97.62/350	3.75/97.15/350
Ultrasonic	2.01/95.76/300	2.05/94.78/350	2.12/95.29/350
PBSM	0.85/93.48/400	0.74/96.56/350	0.66/97.85/350
VSVC	0.86/98.50/300	0.76/97.97/350	0.84/97.21/350
RSRT(Stretch)	**0.95/99.20/200**	**1.01/97.25/200**	**1.02/98.10/150**
RSRT(Squeeze)	**0.97/98.86/150**	**1.12/98.69/150**	**1.22/97.85/200**

On the other hand, WER is a common metric to compare the difference in speech content between predicted and ground truth words. We use the latest SV system ERes2Net [62] for TCR evaluation, and the accepted score threshold is set to 0.70. We also used paraformer [63] for WER evaluation with its open-source code.

4.4. Main results

Tables 2–5 show the main results of backdoor attack evaluation on KWS and TSER tasks. Table 6 shows the proposed trigger's stealthiness evaluation results. We randomly select 500 clean samples in the test dataset and generate poisoned samples with all triggers. We compare the main evaluation metrics, which include AV, ASR, and PN. In the experiments, we found that all the ASR values will increase with the increase of PN values, ultimately approaching 100%. Thus, the tables only show all the **highest** ASR values in the tables with the **highest** PN. The stretching ratio is 2.0, and the squeezing ratio is 0.5. Then, we will analyse our method and baseline methods from attack metrics and stealthiness metrics. For each metric, we will respectively analyse the proposed method and baseline methods. We bold the experimental results of our method to demonstrate that our approach has a higher ASR under lower levels of PN and AV in KWS and TSER tasks.

4.4.1. Attack results analysis of RSRT method

PN Analysis: In the proposed stretching and squeezing results, the PN values are less or equal to 300 in all cases of the proposed method. However, the PN values of other methods are all greater than 300. It is noted that 500 poisoned samples equals about 1% and 0.5% poisoning rate calculated on GSCv1 and GSC v2. In other words, our methods have a very low poisoning rate.

ASR Analysis: Our method outperforms noise-based triggers by 3% to 5% in ASR, demonstrating the superiority of rhythm trigger. The ASR values of baseline methods are hard to closely reach to a high level due to the speech quality being damaged by noise. However, our method modifies the components of speech without loss, and this modification still sounds like high-quality speech to the human ear. Thus, the ASR gained high levels.

AV Analysis: As we know, in deep learning, the data quality extremely influences the classifying ability of models. Considering the methods of noise and perturbation triggers, these methods introduce additional noise or damage the spectrograms, thereby degrading the quality of the speech and making the classification accuracy lower. The higher the decrease in accuracy, the greater the AV value. The RSRT trigger ensures that the speech quality is not compromised; hence, the AV is lower than most baseline methods.

In conclusion, our proposed method has excellent attack effectiveness due to its low PN compared to other methods and high ASR. Our proposed method changes the speech rhythm while keeping the content, timbre, and emotion unchanged, which leads to smaller AV results.

4.4.2. Attack results analysis of baseline methods

Next, we analyse the evaluation results of the baseline methods. The triggers can be classified in two ways. On the one hand, perturbation triggers include adding noisy clips to clean speech utterances or incorporating spectrograms using particular patterns. These triggers are BadNets, PIBA, DABA, and Ultrasonic, as shown in Fig. 5(d–e). On the other hand, element triggers (PBSM and VSVC) change a single speech component and keep excepted components unchanged, as shown in Fig. 5(e–f). The results are various due to the unique perspective of the triggers. Next, we will analyse the results using the three metrics.

PN Analysis: In backdoor attacks, the more contamination there is, the easier it is for the model to learn the characteristics of the trigger. However, this can also cause degradation in the main classification task (as opposed to the backdoor classification task) training. Therefore, when the attack success rate of the trigger reaches 100%, this extreme poisoning number can reflect the attack capability of the trigger. We found the element triggers need no more than 350 PN, while the perturbation triggers mostly need more than 400 PN. In conclusion, the element triggers behave better than the perturbation triggers in speech backdoor attacks.

ASR Analysis: The BadNets trigger has a decent ASR in two tasks. Because the trigger adds a tiny pixel-level single pattern to a benign spectrogram, it cannot influence the whole recognition of the utterance. PIBA, DABA, and Ultrasonic triggers make complex incorporation into spectrograms due to extreme spectrogram modification in the specific time-frequency domains. The ASR values are high. However, these methods require a high poisoning number for the model not to recognize them as noise. During the early stages of neural network training, they may still be regarded as noise, leading to disruption of emotion and speech quality, thus affecting the learning ability of the model from the outset and ultimately resulting in poorer training outcomes.

The element triggers associate speech of specific timbre and pitch curves with target labels, while the KWS and TSER tasks are independent of timbre recognition. Therefore, these methods also have high ASR results.

AV Analysis: We believe that in the early stages of training a classification task, the speech data quality significantly impacts the model's proper convergence. Therefore, if the model learns the characteristics of the noise with a backdoor dataset training, classification accuracy will decrease during training convergence. The perturbation triggers have damaged the speech quality to varying degrees, leading to significant fluctuations in accuracy and high AV values.

Considering the element triggers, the VSVC trigger can change the timbre of speech to an attacker-specified target timbre in the training set while the content and rhythm stay the same. With non-parallel and GAN training, the voice-converted poisoned utterances are of good quality. So, the AV values are both low. The PBSM trigger boosts the pitch of speech and masks the boosted voice with a masking sound to form a peak sound as a poisoned sample. This operation also slightly degrades the speech's quality, which leads to low AV values and a good poisoning rate.

4.4.3. Stealthiness evaluation results

Table 6 shows all the evaluations of speech backdoor methods' PESQ, TCR and WER. To ensure speech quality, the poisoned samples should have a **high PESQ**. Additionally, to avoid intelligent detection models from checking timbre and content, the poisoned samples generated by the trigger should have a **high TCR** and a **low WER**. We also show mel spectrograms with different triggers in Fig. 5.

TCR and WER Analysis: As shown in Table 6. We use baseline and proposed triggers in the evaluation to form an utterance pair. Then, we test whether the clean and poisoned ones can be derived from the same speaker and content by a speaker verification system and automatic speech recognition system. Our proposed method can retain the content and timbre well, while the TCR and WER are close to 1 and 0 in stretching or squeezing operations. The VSVC trigger gets very low TCR and high TCR because the timbre converts completely. The PBSM trigger does not change timbre but slightly destroys speech quality. Thus, the WER result is not better than that of our proposed method. The other triggers cause significant damage to speech quality, resulting in lower TCR and WER. Generally, the methods that keep timbre and content unchanged can deceive automatic deep detection and gain good stealthiness.

Table 6
The stealthiness evaluation with all triggers.

Trigger	PESQ	TCR(%)	WER(%)
without trigger	2.43	99.2	0.00
BadNets	2.06	92.8	1.46
PIBA	1.23	78.7	19.5
DABA	1.04	56.7	23.1
Ultrasonic	1.54	86.7	10.5
PBSM	1.96	93.6	2.67
VSVC	2.15	0.941	1.24
RSRT(Stretch)	**2.37**	**98.7**	**1.05**
RSRT(Squeeze)	**2.25**	**97.6**	**1.25**

Table 7
Ablation study: the ASR with different hyper-parameters.

RSRT ratio	Poisoned number			
	50	100	150	200
Squeeze(1/2)	87.56	93.45	**99.97**	1.0
Squeeze(2/3)	89.92	95.52	97.75	1.0
Stretch(1.2)	10.67	65.67	78.91	86.75
Stretch(1.5)	86.72	91.45	98.91	1.0
Stretch(2.0)	77.99	87.85	**99.10**	1.0

PESQ Analysis: The PESQ values of benign utterances should nearly reach 2.43. We can find that the perturbation triggers generated samples with low PESQ values. In contrast, the element triggers generated samples with high PESQ values because the conversion of elements causes very little damage to the speech quality. As shown in Table 6, the proposed RSRT trigger generates samples which are close to benign samples.

4.5. Ablation study

In this section, we discuss the effect of hyper-parameters in our attack. The key hyper-parameters are the ratios of rhythm transformation and poisoning number. We can control the selected frames and copy times in the RSRT operations to produce different lengths of poisoned utterances. The RSRT ratio is equal to the stretched or squeezed length divided by the origin length. Each experiment is repeated three times to reduce the effect of randomness.

Effects of Poisoning Number: We replace the poisoning rate with the poisoning number because our proposed trigger is effective and needs a very low number of poisoning samples. The poisoning rate corresponding to 50 samples is 0.22%. As shown in Table 7, the ASR increases with the increase in the number of poisoning number in general. It is indicated that in cases of low poisoning rates, compared to squeezing.

Effects of Rhythm Transformation Ratio: The best effectiveness is shown in the set of 0.5 and 1.5 rhythm transformation ratios. With squeezing, a relatively low poisoning number of 150 can achieve an excellent attack success rate of 99.97%. With stretching, a relatively low poisoning number of 150 can achieve a superb attack success rate of 99.10%. The squeezing ASR values under each poisoned number are obviously higher than the stretching ones. This also indicates that squeezing behaves better than stretching.

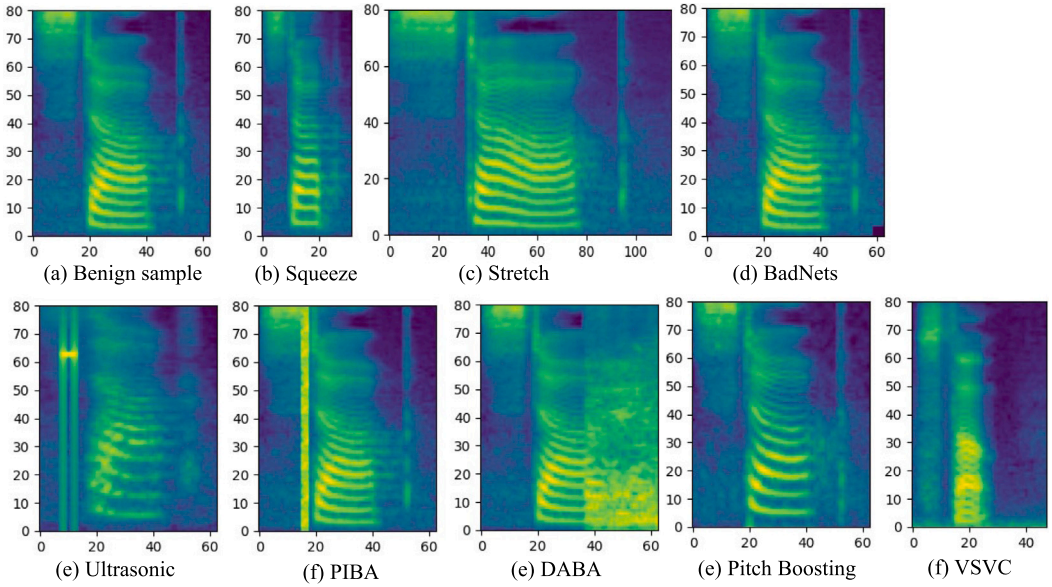

Fig. 5. The mel spectrogram visualization of different poison samples with mentioned triggers. The (b) and (c) show our proposed triggers. The (d)–(e) shows the poisoning utterance by perturbation triggers. The (e) and (f) show the poisoning utterance by element triggers.

5. Conclusion

The paper proposed a speech backdoor attack method called RSRT, mainly combining VAD, RSRT, and neural vocoder. This method achieves very high ASR while maintaining a shallow poisoning rate. The proposed trigger can avoid two kinds of main detection by speaker verification system and automatic speech recognition and gains excellent stealthiness and speech quality. The experiments demonstrate the superb performance of efficiency and stealthiness of speech backdoor attacks with our method. We think that changing the rhythm or some speech components of speech is an exploratory new approach to speech backdoor attacks. In future research, the RSRT method can be applied to different languages and speaker scenarios to verify its generalization and cross-linguistic effectiveness. Considering that current speech recognition and speaker verification systems are continuously improving in terms of security and detection capabilities, future research could also focus on how to maintain a high level of stealthiness in more complex attack environments while developing more intelligent adversarial trigger generation strategies.

CRediT authorship contribution statement

Wenhan Yao: Writing – original draft, Methodology. **Jiangkun Yang:** Visualization, Resources. **Yongqiang He:** Investigation, Formal analysis. **Jia Liu:** Writing – review & editing, Software. **Weiping Wen:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

References

[1] M. Anusuya, S.K. Katti, Speech recognition by machine, a review, 2010, arXiv preprint arXiv:1001.2267.

[2] Y. Gao, B.G. Doan, Z. Zhang, S. Ma, J. Zhang, A. Fu, S. Nepal, H. Kim, Backdoor attacks and countermeasures on deep learning: A comprehensive review, 2020, arXiv preprint arXiv:2007.10760.

[3] T. Gu, K. Liu, B. Dolan-Gavitt, S. Garg, Badnets: Evaluating backdooring attacks on deep neural networks, IEEE Access 7 (2019) 47230–47244.

[4] A. Turner, D. Tsipras, A. Madry, Label-consistent backdoor attacks, 2019, arXiv preprint arXiv:1912.02771.

[5] J. Dai, C. Chen, Y. Li, A backdoor attack against lstm-based text classification systems, IEEE Access 7 (2019) 138872–138878.

[6] X. Pan, M. Zhang, B. Sheng, J. Zhu, M. Yang, Hidden trigger backdoor attack on {NLP} models via linguistic style manipulation, in: 31st USENIX Security Symposium (USENIX Security 22), 2022, pp. 3611–3628.

[7] C. Chen, J. Dai, Mitigating backdoor attacks in lstm-based text classification systems by backdoor keyword identification, Neurocomputing 452 (2021) 253–262.

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

[9] C. Shi, T. Zhang, Z. Li, H. Phan, T. Zhao, Y. Wang, J. Liu, B. Yuan, Y. Chen, Audio-domain position-independent backdoor attack via unnoticeable triggers, in: Proceedings of the 28th Annual International Conference on Mobile Computing and Networking, 2022, pp. 583–595.

[10] S. Koffas, J. Xu, M. Conti, S. Picek, Can you hear it? backdoor attacks via ultrasonic triggers, in: Proceedings of the 2022 ACM Workshop on Wireless Security and Machine Learning, 2022, pp. 57–62.

[11] Y. Kong, J. Zhang, Adversarial audio: A new information hiding method and backdoor for dnn-based speech recognition models, 2019, arXiv preprint arXiv:1904.03829.

[12] J. Ye, X. Liu, Z. You, G. Li, B. Liu, DriNet: dynamic backdoor attack against automatic speech recognization models, Appl. Sci. 12 (12) (2022) 5786.

[13] Q. Liu, T. Zhou, Z. Cai, Y. Tang, Opportunistic backdoor attacks: Exploring human-imperceptible vulnerabilities on speech recognition systems, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 2390–2398.

[14] Y. Luo, J. Tai, X. Jia, S. Zhang, Practical backdoor attack against speaker recognition system, in: International Conference on Information Security Practice and Experience, Springer, 2022, pp. 468–484.

[15] Z. Ye, T. Mao, L. Dong, D. Yan, Fake the real: Backdoor attack on deep speech classification via voice conversion, 2023, arXiv preprint arXiv:2306.15875.

[16] H. Cai, P. Zhang, H. Dong, Y. Xiao, S. Ji, VSVC: Backdoor attack against keyword spotting based on voiceprint selection and voice conversion, 2022, arXiv preprint arXiv:2212.10103.

[17] H. Cai, P. Zhang, H. Dong, Y. Xiao, S. Ji, PBSM: Backdoor attack against keyword spotting based on pitch boosting and sound masking, 2022, arXiv preprint arXiv:2211.08697.
[18] H. Cai, P. Zhang, H. Dong, Y. Xiao, S. Koffas, Y. Li, Towards stealthy backdoor attacks against speech recognition via elements of sound, 2023, arXiv preprint arXiv:2307.08208.
[19] C.H. Chan, K. Qian, Y. Zhang, M. Hasegawa-Johnson, Speechsplit2. 0: Unsupervised speech disentanglement for voice conversion without tuning autoencoder bottlenecks, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2022, pp. 6332–6336.
[20] D. Wang, L. Deng, Y.T. Yeung, X. Chen, X. Liu, H. Meng, Vqmivc: Vector quantization and mutual information-based unsupervised speech representation disentanglement for one-shot voice conversion, 2021, arXiv preprint arXiv:2106.10132.
[21] A. De Cheveigné, H. Kawahara, YIN, a fundamental frequency estimator for speech and music, J. Acoust. Soc. Am. 111 (4) (2002) 1917–1930.
[22] K. Qian, Y. Zhang, S. Chang, M. Hasegawa-Johnson, D. Cox, Unsupervised speech decomposition via triple information bottleneck, in: International Conference on Machine Learning, PMLR, 2020, pp. 7836–7846.
[23] P. Govalkar, J. Fischer, F. Zalkow, C. Dittmar, A comparison of recent neural vocoders for speech signal reconstruction, in: Proc. 10th ISCA Speech Synthesis Workshop, 2019, pp. 7–12.
[24] J. Lorenzo-Trueba, T. Drugman, J. Latorre, T. Merritt, B. Putrycz, R. Barra-Chicote, A. Moinet, V. Aggarwal, Towards achieving robust universal neural vocoding, 2018, arXiv preprint arXiv:1811.06292.
[25] S. Hershey, S. Chaudhuri, D.P. Ellis, J.F. Gemmeke, A. Jansen, R.C. Moore, M. Plakal, D. Platt, R.A. Saurous, B. Seybold, et al., CNN architectures for large-scale audio classification, in: 2017 Ieee International Conference on Acoustics, Speech and Signal Processing (Icassp), IEEE, 2017, pp. 131–135.
[26] K. Palanisamy, D. Singhania, A. Yao, Rethinking CNN models for audio classification, 2020, arXiv preprint arXiv:2007.11154.
[27] K. Banuroopa, D. Shanmuga Priyaa, MFCC based hybrid fingerprinting method for audio classification through LSTM, Int. J. Nonlinear Ana. Appl. 12 (Special Issue) (2021) 2125–2136.
[28] Y. Gong, Y.-A. Chung, J. Glass, Ast: Audio spectrogram transformer, 2021, arXiv preprint arXiv:2104.01778.
[29] Y. Liu, X. Ma, J. Bailey, F. Lu, Reflection backdoor: A natural backdoor attack on deep neural networks, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16, Springer, 2020, pp. 182–199.
[30] X. Chen, C. Liu, B. Li, K. Lu, D. Song, Targeted backdoor attacks on deep learning systems using data poisoning, 2017, arXiv preprint arXiv:1712.05526.
[31] B. Tran, J. Li, A. Madry, Spectral signatures in backdoor attacks, Adv. Neural Inf. Process. Syst. 31 (2018).
[32] S. Zhao, X. Ma, X. Zheng, J. Bailey, J. Chen, Y.-G. Jiang, Clean-label backdoor attacks on video recognition models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 14443–14452.
[33] S. Cheng, Y. Liu, S. Ma, X. Zhang, Deep feature space trojan attack of neural networks by controlled detoxification, in: Proceedings of the AAAI Conference on Artificial Intelligence, 35, 2021, pp. 1148–1156.
[34] A. Saha, A. Subramanya, H. Pirsiavash, Hidden trigger backdoor attacks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 11957–11965.
[35] J. Lin, L. Xu, Y. Liu, X. Zhang, Composite backdoor attack for deep neural network by mixing existing benign features, in: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, 2020, pp. 113–131.
[36] Y. Chen, An invisible backdoor attack based on semantic feature, 2024, arXiv preprint arXiv:2405.11551.
[37] R. Wang, H. Chen, Z. Zhu, L. Liu, B. Wu, Versatile backdoor attack with visible, semantic, sample-specific, and compatible triggers, 2023, arXiv preprint arXiv:2306.00816.
[38] X. Han, S. Yang, W. Wang, Z. He, J. Dong, Is it possible to backdoor face forgery detection with natural triggers? 2023, arXiv preprint arXiv:2401.00414.
[39] E. Bagdasaryan, V. Shmatikov, Blind backdoors in deep learning models, in: 30th USENIX Security Symposium (USENIX Security 21), 2021, pp. 1505–1521.
[40] Y. Zhou, R.Q. Hu, Y. Qian, Backdoor attacks and defenses on semantic-symbol reconstruction in semantic communications, 2024, arXiv preprint arXiv:2404.13279.
[41] Y. Liu, Y. Xie, A. Srivastava, Neural trojans, in: 2017 IEEE International Conference on Computer Design, ICCD, IEEE, 2017, pp. 45–48.
[42] M. Villarreal-Vasquez, B. Bhargava, Confoc: Content-focus protection against trojan attacks on neural networks, 2020, arXiv preprint arXiv:2007.00711.
[43] Y. Li, T. Zhai, Y. Jiang, Z. Li, S.-T. Xia, Backdoor attack in the physical world, 2021, arXiv preprint arXiv:2104.02361.
[44] S. Koffas, L. Pajola, S. Picek, M. Conti, Going in style: Audio backdoors through stylistic transformations, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2023, pp. 1–5.
[45] P. Liu, S. Zhang, C. Yao, W. Ye, X. Li, Backdoor attacks against deep neural networks by personalized audio steganography, in: 2022 26th International Conference on Pattern Recognition, ICPR, IEEE, 2022, pp. 68–74.
[46] J. Xin, X. Lyu, J. Ma, Natural backdoor attacks on speech recognition models, in: International Conference on Machine Learning for Cyber Security, Springer, 2022, pp. 597–610.
[47] F. Bous, A. Roebel, A bottleneck auto-encoder for f0 transformations on speech and singing voice, Information 13 (3) (2022) 102.
[48] H. Wang, S. Zheng, Y. Chen, L. Cheng, Q. Chen, Cam++: A fast and efficient network for speaker verification using context-aware masking, 2023, arXiv preprint arXiv:2303.00332.
[49] J. Kong, J. Kim, J. Bae, Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis, Adv. Neural Inf. Process. Syst. 33 (2020) 17022–17033.
[50] J. Pang, Spectrum energy based voice activity detection, in: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference, CCWC, 2017, pp. 1–5, http://dx.doi.org/10.1109/CCWC.2017.7868454.
[51] R. Dugad, N. Ahuja, A fast scheme for downsampling and upsampling in the DCT domain, in: Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348), Vol. 2, IEEE, 1999, pp. 909–913.
[52] P. Warden, Speech commands: a public dataset for single-word speech recognition (2017), 1, 2017, Dataset available from http://download.tensorflow.org/data/speech_commands_v0.
[53] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[54] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, G. Cottrell, A dual-stage attention-based recurrent neural network for time series prediction, 2017, arXiv preprint arXiv:1704.02971.
[55] A. Berg, M. O'Connor, M.T. Cruz, Keyword transformer: A self-attention model for keyword spotting, 2021, arXiv preprint arXiv:2104.00769.
[56] A. Gazneli, G. Zimerman, T. Ridnik, G. Sharir, A. Noy, End-to-end audio strikes back: Boosting augmentations towards an efficient audio classification network, 2022, arXiv preprint arXiv:2204.11479.
[57] K. Zhou, B. Sisman, R. Liu, H. Li, Emotional voice conversion: Theory, databases and ESD, Speech Commun. 137 (2022) 1–18.
[58] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J.N. Chang, S. Lee, S.S. Narayanan, IEMOCAP: Interactive emotional dyadic motion capture database, Lang. Resour. Eval. 42 (2008) 335–359.
[59] Y. Zhang, J. Du, Z. Wang, J. Zhang, Y. Tu, Attention based fully convolutional network for speech emotion recognition, in: 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), IEEE, 2018, pp. 1771–1775.
[60] D. Issa, M.F. Demirci, A. Yazici, Speech emotion recognition with deep convolutional neural networks, Biomed. Signal Process. Control 59 (2020) 101894.
[61] Y. Park, S. Patwardhan, K. Visweswariah, S.C. Gates, An empirical analysis of word error rate and keyword error rate, in: Interspeech, 2008, 2008, pp. 2070–2073.
[62] Y. Chen, S. Zheng, H. Wang, L. Cheng, Q. Chen, J. Qi, An enhanced res2net with local and global feature fusion for speaker verification, 2023, arXiv preprint arXiv:2305.12838.
[63] Z. Gao, S. Zhang, I. McLoughlin, Z. Yan, Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition, 2022, arXiv preprint arXiv:2206.08317.

Wenhan Yao Doctoral candidate at Xiangtan University. Research directions include speech synthesis, speech recognition, and backdoor attacks on speech models. Email: ywh15151@163.com

Jiangkun Yang The main research direction is backdoor attacks on image semantic segmentation. Email: 202221633060@smail.xtu.edu.cn

Yongqiang He Research directions: (1) Software security analysis: Software vulnerability mining, analysis and exploitation, malware analysis, eradication, and defense; (2) Research on penetration techniques based on artificial intelligence. Email: heyq_005@163.com

Jia Liu Main research directions: Group-type information system, infrastructure, and the construction, operation, and management of cyber information security projects. Email: 13466315567@vip.163.com

Weiping Wen Main research areas include: system and network security, big data and cloud security, and intelligent computing security research, etc. Email: weipingwen@pku.edu.cn