Computer Speech and Language 72 (2022) 101315

Contents lists available at ScienceDirect

Computer Speech & Language

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

Lightweight and irreversible speech pseudonymization based on data-driven optimization of cascaded voice modification modules

Hiroto Kai [a,*], Shinnosuke Takamichi [b], Sayaka Shiota [a], Hitoshi Kiya [a]

[a] Tokyo Metropolitan University, Graduate School of Systems Design, 6-6 Asahigaoka, Hino-shi, Tokyo, Japan
[b] The University of Tokyo, Graduate School of Information Science and Technology, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan

ARTICLE INFO

Keywords:
Speech pseudonymization
Voice privacy protection
Data-driven optimization
Cascaded voice modification
Irreversibility

ABSTRACT

In this paper, we propose a speech pseudonymization framework that utilizes cascaded and superposition-based voice modification modules. With increasing opportunities to use spoken dialogue systems nowadays, research regarding protecting the privacy of speaker information encapsulated in speech data is attracting attention. Pseudonymization, which is one method for voice privacy protection, aims to keep the intelligibility of speech while simultaneously suppressing speaker-specific information. One motivation of our framework is to achieve a reliable pseudonymization performance with light computation. To do this, we utilize the advantages of both machine learning-based and signal processing-based approaches. The advantages are (1) using signal processing-based methods parameterized with few hyperparameters and (2) using machine learning-based optimization to optimize all hyperparameters on the basis of black-box systems consisting of automatic speaker verification and automatic speech recognition. Our method of cascading signal processing modules, which are jointly optimized in a data-driven manner, can pseudonymize speech in a lightweight manner. Additionally, we discuss irreversible pseudonymization approaches and propose a superposition approach, yet another pseudonymization method that is more irreversible than the cascade method in terms of estimating the adequate parameters to recover the original signal. From the experimental results conducted under the VoicePrivacy 2020 protocols, we can demonstrate that (1) our cascade method succeeds in deteriorating the speaker recognition rate by over 24% while simultaneously improving the speech recognition rate by approximately 8% compared with a signal processing-based baseline system of VoicePrivacy 2020 and that (2) our superposition method works comparable to our cascade method in terms of pseudonymization performance.

1. Introduction

Nowadays, we see more opportunities to use spoken dialogue systems with machines and in E-commerce, such as online shopping and online banking. In recent years, the growth of such services has attracted attention on automatic speaker verification (ASV) systems and automatic speech recognition (ASV) systems, increasing the value of speech as a source of big data. Speech includes the information of spoken content as well as the speaker-specific features of the speaker, i.e., the age of the speaker, gender, emotional state, and racial background. With the value of speech as personal information drawing attention, privacy protection and security aspects surrounding the use and sharing of speech data is becoming a concerning element.

* Corresponding author.
E-mail addresses: kai-hiroto@ed.tmu.ac.jp (H. Kai), shinnosuke_takamichi@ipc.i.u-tokyo.ac.jp (S. Takamichi), sayaka@tmu.ac.jp (S. Shiota), kiya@tmu.ac.jp (H. Kiya).

https://doi.org/10.1016/j.csl.2021.101315
Received 29 January 2021; Received in revised form 11 October 2021; Accepted 22 October 2021
Available online 6 November 2021

Generally, there are two well known frameworks for protecting privacy: encryption and speaker de-identification. Encryption uses an algorithm to encrypt data, allowing only users possessing a secret key to decrypt it. Speaker de-identification uses some type of method or technology to modify the speech data in the sense that the de-identified speech is natural and intelligible but the original speaker cannot be identified. Speaker de-identification is considered to be low in computational complexity compared with encryption. It also does not require special knowledge outside of the speech field. Furthermore, speaker de-identification can be classified into two methods: anonymization and pseudonymization. Although anonymization and pseudonymization are interchangeable, we have defined the terms on the basis of the General Data Protection Regulation (Hintze and El Emam, 2018). Anonymization has an advantage in that original speech cannot be recovered from protected speech, but ASV between anonymized pairs of speech is not possible. Pseudonymization has a disadvantage in that a protected voice can be recovered only under certain conditions, but ASV between pseudonymized pairs of speech is possible (Noé et al., 2020). Owing to ASV between protected pairs of speech being possible, this study focuses on the pseudonymization of speech.

Motivated by recognition schemes in other fields, we consider an authentication scenario where a user verification system is constructed in the cloud. Recently, we see more and more systems relying on cloud computing to address the deficiencies of local devices, such as constrained resources and a low computation power. However, storing sensitive data in the cloud without protection provokes privacy concerns in users. Thus, user verification systems are required that use only encrypted data. In the case of ASV, the user would need to send a speech sample to the cloud for speaker verification. Needless to say, the identity of a speaker is pseudonymized when enrolled in ASV, and the speech sample used for verification is also pseudonymized via their local devices before it is sent to the cloud. Thus, we consider speech pseudonymization methods to protect speaker identity. Speech pseudonymization methods can be separated into machine learning-based methods (Bahmaninezhad et al., 2018; Fang et al., 2019) and signal processing-based methods (Sridharan et al., 1991; Lagendijk et al., 2013). Machine learning, i.e., deep learning, has made significant contributions to both image and sound fields. It is well known to achieve very high performance considering that, nowadays, most state-of-the-art methods utilize some type of machine learning. While there are many benefits to using machine learning, one of the drawbacks is the amount of resources needed to utilize it. Typically, machine learning, e.g., deep neural networks (DNNs), requires millions of parameters to be optimized. Although the optimization process can be done automatically using a large amount of training data, the process can be time-consuming and computationally heavy. In comparison, signal processing methods are relatively lightweight compared with machine learning. Only a few parameters need optimization, resulting in less time for computation. However, it is undeniable that the performance achieved using signal processing methods is usually inferior to that of machine learning. Another disadvantage is that parameters are often manually tuned in signal processing-based methods, requiring the user to decide the parameters empirically. Despite the drawbacks of signal processing-based methods, we consider such methods to be more feasible to implement in local devices due to their lightweight nature. Systems heavily dependent on machine learning-based methods require high-performance devices, making them unsuitable under resource-limited conditions. Therefore, we considered a method that would maintain the lightweight characteristic of signal processing-based methods while optimizing the parameters in a data-driven manner.

So far, we have proposed a black-box optimization based on signal processing (Kai et al., 2021), where our proposed method is a speech pseudonymization framework that utilizes the advantages of machine learning and signal processing-based approaches. Due to using signal processing-based methods to modify input speech, our framework is lightweight compared with DNN-based methods. Furthermore, by utilizing training data to optimize the hyperparameters, we mitigated the troublesome manual tuning of hyperparameters. Our framework modifies input speech on the basis of an objective function value that is defined on the basis of the ASV score and the ASR score of pseudonymized speech. By minimizing the objective function value, we aim to suppress speaker information without compromising the intelligibility of the speech. One of the difficulties of using signal processing-based methods for modification is the limited performance. However, by combining multiple methods to modify the input speech, we expanded the search space of the parameter set, resulting in more optimal optimization of the hyperparameters. While the irreversibility of protected information is an important issue in the privacy-protection research field, there were no discussions on the irreversibility of pseudonymized speech in VoicePrivacy 2020. Thus, we also focus on irreversible pseudonymization approaches and propose a superposition approach, yet another pseudonymization method. In terms of independent component analysis, it is more difficult to estimate the parameters of superposed speech compared with speech anonymized by using the cascaded method, increasing its irreversibility. Speech superposition uses the same optimization framework as the cascade method does to find the optimal parameters. Experimental results show an improvement in all three evaluation tasks we have prepared for both of our pseudonymization methods: a cascade method for pseudonymization and a method of superposing speech for better irreversible pseudonymization.

First, Section 2 introduces works regarding the privacy protection of speech. In Section 3 and Section 4, we will present our proposed cascaded voice modification and irreversible pseudonymization methods, respectively. Section 5 will explain the experimental setup and show the obtained results, and finally, Section 6 concludes the paper.

2. Privacy protection for speech

2.1. VoicePrivacy 2020

Recently, a competition called VoicePrivacy 2020 (Tomashenko et al., 2020a) was held. The competition focused on speech pseudonymization, facilitating the development of pseudonymization techniques under common conditions along with the proposal of effective evaluation metrics for voice privacy. For the first competition, a common dataset, two baseline systems, and evaluation

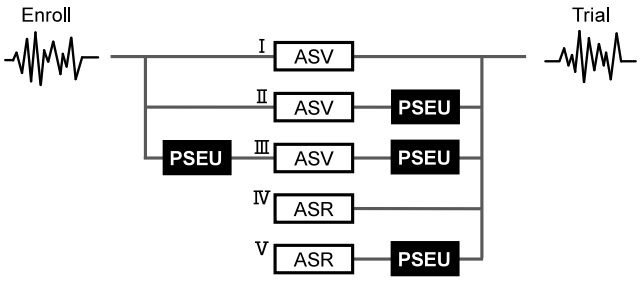


Fig. 1. Evaluation tasks for VoicePrivacy 2020. Tasks I-III are evaluation tasks for ASV, and tasks IV and V are those for ASR. PSEU represents pseudonymization system.

protocols, i.e., objective evaluation tasks (shown in Fig. 1) and subjective ones, were provided. Upon participating in the competition, the participant had to satisfy three requirements given by the organizers. The first requirement was that the output of the constructed system must be a waveform. For the second requirement, pseudonymized speech must maintain intelligibility and simultaneously pseudonymize speaker information. For the third requirement, enrollment and trial utterances from the same speaker must be pseudonymized to different speakers. The satisfaction of the three tasks was part of the evaluation standard of the competition. The objective evaluation was done by the participants, while the subjective evaluation was conducted by the organizers of the competition. In this study, we will refer only to the objective evaluation. Tasks I-III calculate the equal error rate (EER), while tasks IV and V calculate the word error rate (WER) using the provided ASV and ASR systems. The ASV system is based on the x-vector, one of the state-of-the-art methods for ASV (Garcia-Romero et al., 2019; Srivastava et al., 2020a). For ASR, the system uses a TDNN-F acoustic model and a trigram language model with i-vector and MFCC as its input features (Peddinti et al., 2015). Both systems are constructed using the LibriSpeech (Panayotov et al., 2015) dataset and provided with trained model parameters. Throughout the whole evaluation, the model parameters are fixed and cannot be changed. Tasks I to III are conducted to evaluate the pseudonymization performance of a constructed system. Following the protocols of VoicePrivacy 2020, it is desired that ASV be possible, i.e., the EER be low in task I due to no modification being done between the enrollment and trial utterances. As for task II, the EER is expected to be high since only the trial utterance is pseudonymized, resulting in a difference in speaker identity between the enrollment and trial utterances. The results between task II and III are assumed to be the same since task III requires the system to pseudonymize enrollment and trial utterances into different pseudo-speakers, respectively. Tasks IV and V are conducted to evaluate the intelligibility between clean and pseudonymized speech. In order for a system to meet the second requirement, it is ideal that task V show values close to those of task IV.

2.2. Related works

VoicePrivacy 2020 has inspired the community to propose various methods and different evaluation measures for voice privacy. Most of the methods are based on machine learning approaches that focus on x-vectors, which are widely used in the field of ASV. Examples include a method that changes x-vectors, resulting in different speaker identities in pseudonymized speech (Mawalim et al., 2020), a method using an end-to-end ASR (Champion et al., 2020), and a method for pseudonymizing x-vectors using autoencoders (Espinoza-Cuadros et al., 2020). For signal processing-based methods, pseudonymization done on the basis of the difference in vocal tract length, speech rate, and the high frequency range of speech was proposed (Dubagunta et al., 2020). The metrics proposed regarding evaluation measures include a visual representation and evaluation of speaker similarity using a confusion matrix (Noé et al., 2020) and a method for evaluating pseudonymized speech from a forensic perspective (Nautsch et al., 2020).

3. Data-driven optimization for cascaded voice modification modules

3.1. Overview of proposed methods

We consider lightweight pseudonymization that satisfies the VoicePrivacy 2020 conditions stated in Section 2.1. Voice modification methods for pseudonymization can be classified into two types: machine learning-based (Bahmaninezhad et al., 2018; Fang et al., 2019) and signal processing-based methods (Sridharan et al., 1991; Lagendijk et al., 2013). The former (e.g., deep learning) has millions of model parameters but can optimize the model parameters in a data-driven manner (e.g., maximum likelihood estimation). The latter has only a small number of parameters, but they are difficult to optimize and are often hand-tuned. Here, we propose a method that utilizes advantages from both machine learning-based and signal processing-based methods and use a framework that optimizes their parameters in a data-driven manner. Fig. 2 shows the flow of our method. Our voice modification method is done using the modification methods introduced in Section 3.3. Input speech is modified by using voice modification methods, generating pseudonymized speech. The pseudonymized speech is then evaluated with ASV and ASR systems by using the speaker labels and text of the original speech, respectively. At this time, the scores obtained are the EER and WER from ASV and ASR, respectively. Since

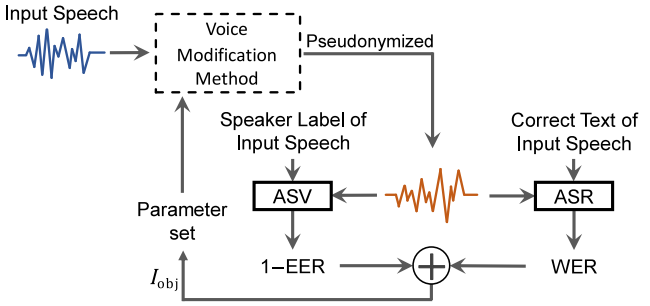


Fig. 2. Flow of proposed methods. I_{obj} denotes objective function.

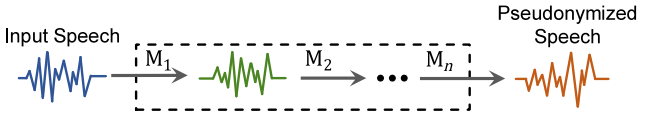


Fig. 3. Modification procedure of cascaded voice modification method. Modification methods are denoted as M_* (M_n indicates nth method used for modification).

the VoicePrivacy 2020 conditions can be seen as maximizing EER and minimizing WER, the objective is to minimize the weighted sum of a negative EER and positive WER. Therefore, we choose a new parameter set that makes the objective smaller and update the parameters of the modification modules. We iteratively search for a parameter set that minimizes the objective.

3.2. Cascaded voice modification

Cascaded voice modification modules are represented as modification modules connected in series as shown in Fig. 3. The output of the nth module is fed to the $(n+1)$th module. Let x and x' be the input speech and output speech of the nth module, respectively. After inputting x into the cascaded voice modification modules consisting of n modules, the output is represented as

$$x' = \mathrm{M}_n\left(\mathrm{M}_{n-1}\left(\ldots \mathrm{M}_1\left(x\right)\right)\right), \tag{1}$$

where M_n denotes the nth modification method.

3.3. Voice modification methods

We here describe the signal processing-based methods M_* that are part of our cascaded voice modification modules. Each method has an individual scalar parameter α_*.

3.3.1. Vocal tract length normalization

Vocal tract length normalization (VTLN) is mainly used in ASR to remove distortion caused by differences in the length of vocal tracts (Lee and Rose, 1998). This method modifies an amplitude spectrum of original speech according to a warping function. Let $\omega \in [0,1]$ and $\omega' \in [0,1]$ be an original normalized frequency and warped one, respectively. $\omega = 1$ corresponds to the Nyquist frequency. ω is warped into ω' as

$$\omega' = \omega + 2\arctan\frac{\alpha_{\mathrm{vtln}}\sin\left(\omega\right)}{1-\alpha_{\mathrm{vtln}}\cos\left(\omega\right)}, \tag{2}$$

where $\alpha_{\mathrm{vtln}} \in [-1,1]$ denotes the warping factor. Fig. 4 shows this warping. When $\alpha_{\mathrm{vtln}} < 0$ and $\alpha_{\mathrm{vtln}} > 0$, the warping curves (left in the figure) become convex and concave, respectively. They contract and expand an amplitude spectrum (right in the figure), respectively. When $\alpha_{\mathrm{vtln}} = 0$, $\omega' = \omega$, which means no warping. In this paper, we warp frequencies of the log amplitude spectra obtained by applying a short-time Fourier transform (STFT) to original speech. The pseudonymized speech is obtained with an inverse STFT of the modified amplitude spectrogram and original phase spectrogram. A spectrogram of speech modified by using VTLN ($\alpha_{\mathrm{vtln}} = 0.2$) is shown in Fig. 5(b). Compared with the original [Fig. 5(a)], we can see that the formants shift towards the lower frequency band.

3.3.2. McAdams transformation

The McAdams transformation modifies the resonance frequencies (i.e., formant frequencies) of original speech (Patino et al., 2020). We obtain N poles $[p_1,\ldots,p_n,\ldots,p_N]$ by applying linear predictive coding (LPC) (Itakura, 1968) to original speech. Pole

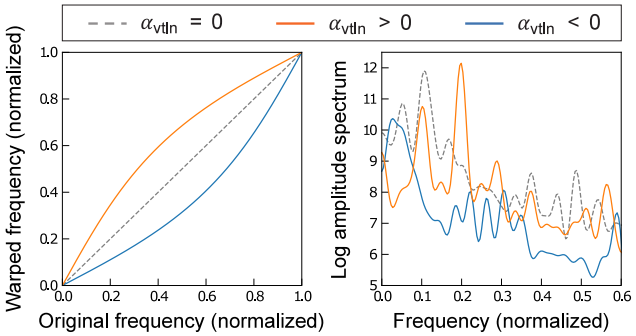


Fig. 4. Example of VTLN: frequency warping function (left) and warped amplitude spectra (right). For clear visualization, we show smoothed amplitude spectra, instead of raw amplitude spectra.

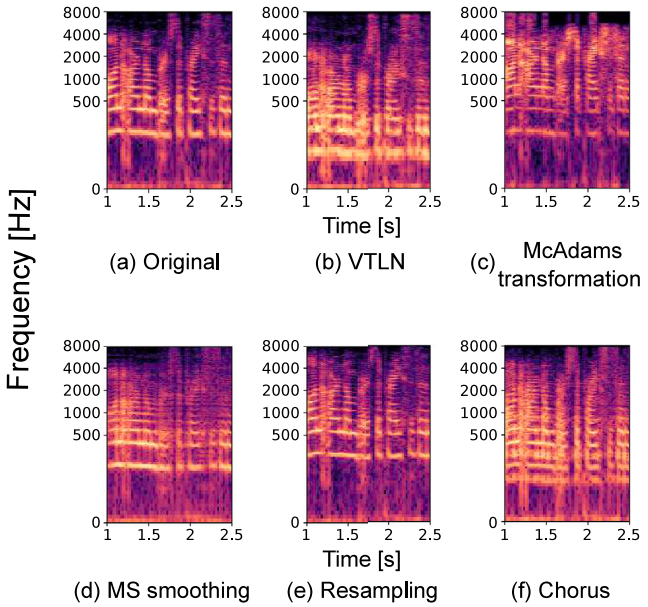


Fig. 5. Spectrograms of (a) original speech, speech modified using (b) VTLN, (c) McAdams transformation, (d) MS smoothing, (e) resampling, and (f) chorus.

$p_n \in \mathbb{C}$ is written as $A_n \exp\left(j\theta_n\right)$ in the polar coordinate, where $A_n \in [0, 1]$ and $\theta_n \in [0, \pi]$ are the absolute value[1] and phase in the upper half of the z-plane, respectively (poles in the lower half of the z-plane are the complex conjugate). They correspond to the formant strength and frequency, respectively. The imaginary unit is represented as j. The McAdams transformation obtains the modified formant frequency $\theta'_n \in [0, \pi]$ by $\theta'_n = \theta_n^{\alpha_{\text{mcadams}}}$, where $\alpha_{\text{mcadams}} \in \mathbb{R}_+$ is the McAdams coefficient. The nth modified pole p'_n consists of the original formant strength and modified formant frequency, i.e., $p'_n = A_n \exp\left(j\theta'_n\right)$. Fig. 6 shows an example. The transformation changes the angles of poles in the z-plane (left in the figure) and contracts or expands the amplitude spectrum (right in the figure).[2] A spectrogram of speech modified by using McAdams transformation ($\alpha_{\text{mcadams}} = 0.7$) is shown in Fig. 5(c). Since $\alpha_{\text{mcadams}} < 1$, we can see that the formant strength became stronger in the higher frequency band compared with the original [Fig. 5(a)].

3.3.3. Modulation spectrum smoothing

Modulation spectrum smoothing removes the temporal fluctuation of speech features (Takamichi et al., 2015). Let $\boldsymbol{X} \in \mathbb{C}^{F \times T}$ be a complex spectrogram obtained by the STFT of original speech, where F and T are the numbers of frequency bins and frames,

[1] Note that, since p_n is calculated from LPC, A_n is always less than 1.

[2] VTLN and the McAdams transformation have a similar role: contracting or expanding the amplitude spectrum of original speech. However, the resulting effects are different from each other. Since VTLN operates in a nonparametric manner, it changes not only the spectral envelope but also the pitch of original speech. In comparison, since the McAdams transformation operates in a parametric manner (i.e., source-filter model), it changes only the spectral envelope parameterized as formants.

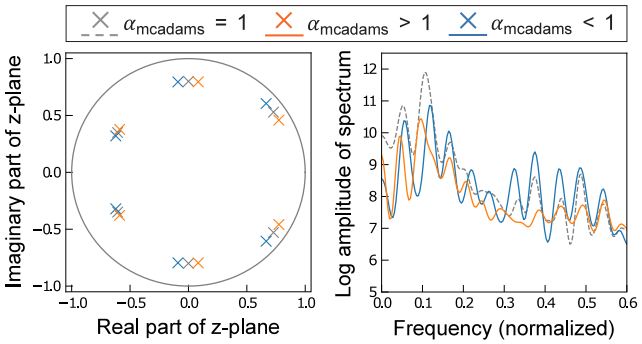


Fig. 6. Example of McAdams transformation: pole reposition in z-plane (left) and modified amplitude spectra (right). Circle in left figure is unit circle. For clear visualization, we show smoothed amplitude spectra, instead of raw amplitude spectra.

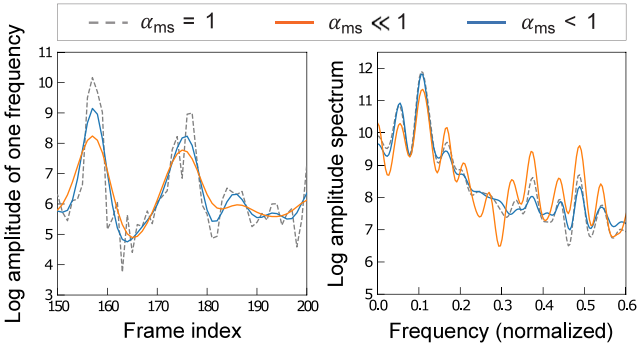


Fig. 7. Example of modulation spectrum smoothing: temporal smoothing of amplitudes (left) and modified amplitude spectra (right). For clear visualization, we show smoothed amplitude spectra, instead of raw amplitude spectra (no outliers were seen outside plots).

respectively. A temporal sequence of a log amplitude spectrogram at frequency f, $\left[\log|X_{f,1}|, \ldots, \log|X_{f,T}|\right]$, is filtered by a zero-phase low pass filter, where $X_{f,t}$ is the $\{f,t\}$th component of $\boldsymbol{X}$. A cutoff modulation frequency[3] of the low pass filter is denoted as $\alpha_{\text{ms}} \in [0,1]$, where 1 corresponds to the Nyquist modulation frequency.[4] Fig. 7 shows an example. As α_{ms} becomes smaller, this method strongly smoothens a temporal sequence (left in the figure) and strongly removes the detailed structure of an amplitude spectrum. The pseudonymized speech is synthesized by an inverse STFT of the smoothed amplitude and original phase spectrograms. A spectrogram of speech modified by using MS smoothing ($\alpha_{\text{ms}} = 0.2$) is shown in Fig. 5(d).

3.3.4. Resampling

Waveform resampling changes the sampling frequency of original speech. Though a typical resampling method changes the duration of original speech, we use a method that does not change the duration. Before resampling with the resampling rate $\alpha_{\text{resample}} \in \mathbb{R}_+$, we stretch original speech with a stretching rate of $1/\alpha_{\text{resample}}$. Namely, pseudonymized speech is obtained by stretching T-sample original speech to $\alpha_{\text{resample}}T$-sample speech and resampling it with a α_{resample}-times faster sampling frequency. Fig. 8 shows an example. With $\alpha_{\text{resample}} = 1$ as the boundary, the speech is stretched or compressed. A spectrogram of speech modified by using resampling ($\alpha_{\text{resample}} = 0.7$) is shown in Fig. 5(e). Since $\alpha_{\text{resample}} < 1$, the sampling frequency is higher in the modified speech, resulting in the formant frequency shifting towards the higher frequency band.

3.3.5. Clipping

Waveform clipping is a distortion method that limits a speech waveform once it exceeds a desired threshold. We use a method based on histogram-based thresholding and hard clipping. Let $x_t \in \mathbb{R}$ be an original speech waveform at time t. We first calculate a cumulative histogram of $|x_t|$. Then, we set a value that exceeds the $\alpha_{\text{clip}} \in [0,1]$ of the cumulative histogram as the threshold value x_{th}. The pseudonymized speech signal at time t, x'_t, is obtained by

$$x'_t = \begin{cases} x_t & (|x_t| < x_{\text{th}}) \\ \text{sign}\left(x_t\right) x_{\text{th}} & (\text{otherwise}), \end{cases} \tag{3}$$

[3] This is not a cutoff frequency because we deal with a modulation spectrum, not a typical spectrum.

[4] The Nyquist modulation frequency is calculated from a frame shift length of STFT. When the shift length is 5 ms, the sampling modulation frequency is the inverse, i.e., 200 Hz, and the Nyquist modulation frequency is half, i.e., 100 Hz.

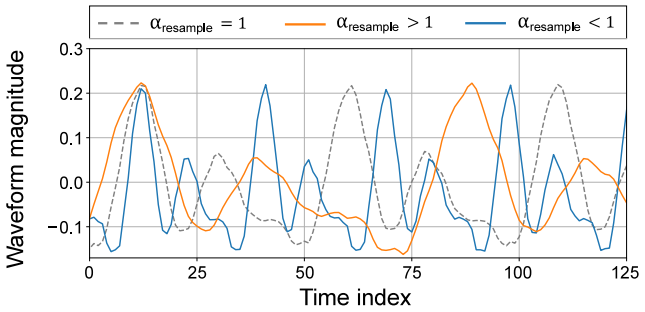


Fig. 8. Example of resampling: waveform magnitude of resampled speech. For clear comparison, we shifted time of pseudonymized speech in order to align first waveform peak of pseudonymized speech with original.

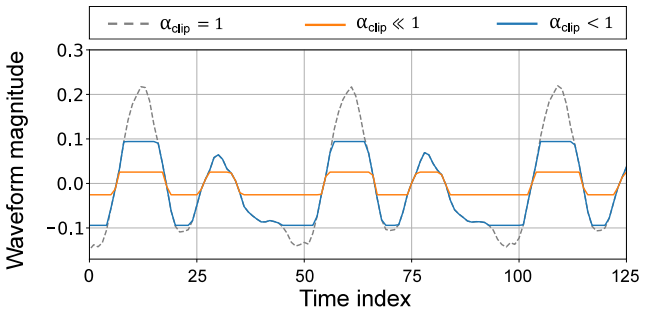


Fig. 9. Example of clipping: waveform magnitude of clipped speech. For clear visualization, we did not preserve power of pseudonymized speech with original (actual clipping process preserves power of original speech).

where $\text{sign}(\cdot)$ is the sign function. Fig. 9 shows an example. As α_{clip} becomes smaller, the threshold value for clipping the waveform magnitude becomes smaller.

3.3.6. Chorus

Chorus is a method that superposes modified speech onto the original speech. Modified speech is obtained by slightly changing the pitch of the original speech. In this paper, given a chorus parameter $\alpha_{\text{chorus}} \in [0, 1]$, we drive VTLN twice with $\alpha_{\text{vtln}} = \pm\alpha_{\text{chorus}}$ and add the results to the waveform domain. A spectrogram of the speech modified by using chorus is shown in Fig. 5(f). Compared with the original [Fig. 5(a)], we can see more formants appear in the spectrogram due to the superposition of multiple pieces of waveform.

3.4. Data-driven optimization

In this paper, data-driven manner means tuning parameters on the basis of data to satisfy a specific task, i.e., minimizing the objective. Let λ be a set (or subset) of the parameters α_* denoted in Section 3.3. We optimize λ by minimizing an objective function given as

$$I_{\text{obj}} = \text{WER} + \omega\,(1 - \text{EER})\,, \tag{4}$$

where ω is the weight of a negative EER.[5] The optimized set $\hat{\lambda}$ is estimated as

$$\hat{\lambda} = \underset{\lambda}{\text{argmin}}\, I_{\text{obj}}. \tag{5}$$

The optimized set allows us to pseudonymize speech with a higher EER and lower WER. We assume that the objective function takes account of task III due to the voice modification systems being based on signal processing. Once the parameters are fixed, the modification is constant, making it possible to guarantee reliable ASV performance even when using both the pseudonymized enrollment utterance and pseudonymized trial utterance.

In the machine learning context, such optimization often utilizes a gradient-based algorithm that uses $\partial I_{\text{obj}}/\partial\lambda$. However, WER and EER are typically not differentiable by λ. To overcome such optimization problems where gradients are not accessible, we focused on using Bayesian optimization (Snoek et al., 2012). This optimization method is beneficial in situations where the objective

[5] We can ignore "1" in the second term during optimization but notate it for intuitively understanding the experimental results shown in the next section.

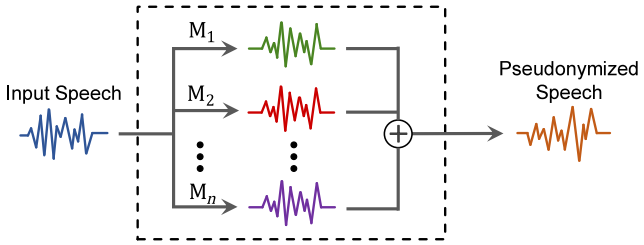


Fig. 10. Modification procedure of speech superposition method. Modification methods are denoted as M_* (M_n indicates nth method used for modification).

function is difficult to evaluate e.g., black-box systems, and is much more efficient than grid searching for optimal parameters. Since we consider the parameter optimization of our framework to be black-box due to it not using gradients nor loss but only scores calculated from ASV and ASR, we used an optimization framework using the Bayesian optimization algorithm in our framework.

3.5. Evaluation protocols

We prepared three tasks for the evaluation phase of our proposed method: tasks II and V from the VoicePrivacy 2020 evaluation plans and a newly defined task III*. In task III*, the enrollment and trial utterances are to be pseudonymized to the same pseudo-speaker if both sets of utterances are from the same speaker. The intention behind evaluating task III* instead of task III from the original VoicePrivacy 2020 protocols is that recognition schemes using encrypted data are a rather common concept in other fields, e.g., the image (23) and medical (24) fields. While databases stored in the cloud hold personal information like passwords and patient medical data, to access that information, the user would need to verify themselves via their local devices. Since storing raw data in the cloud leads to privacy risks and concerns, sending personal information to the cloud to verify a user raises security issues. Thus, user verification done using encrypted enrollment data and encrypted trial data is necessary to build high-security systems. Due to this change, we assume that the ASV score of task III* will become higher. The higher the score, the more successful the system is in verifying speakers even in the pseudo-speaker domain.

4. Irreversible voice modification

4.1. Importance of irreversibility for privacy protection

Due to biometric-based authentication using biometric features that are associated with users' privacy, it is important to protect such features from being leaked. Traditional authentication schemes such as passwords and credit card IDs can be revoked or easily changed by the user when their confidential information is leaked, preventing abuse from adversaries. However, biometric features are unchangeable, heavily affecting the user once they have been compromised. Solutions for such situations are the usage of irreversible transformed versions of biometric features (Xu et al., 2008) and revocable representations of biometric features to preserve privacy (Farooq et al., 2007). In this way, irreversible privacy protection methods are effective against situations assuming the exposure of confidential information to adversaries. It is important to consider assuring the privacy of original data regardless of the disclosure of the keys used to generate the protected data. However, the VoicePrivacy 2020 Challenge did not focus on the irreversibility issue for privacy protection.

4.2. Speech superposition

We propose another pseudonymization method that does not completely follow the VoicePrivacy 2020 protocols but rather focuses on irreversibility. Speech superposition is done by superposing multiple pieces of waveform; each piece of waveform is generated by using different modification methods. Fig. 10 shows speech superposition regarded as a pseudonymized voice modification method. Let x and x' be the input utterance and pseudonymized speech generated by superposing n pieces of modified speech, respectively. x' is represented as

$$x' = \sum_{i=1}^{n} \mathrm{M}_i\,(x). \tag{6}$$

This superposition approach can easily substitute for the cascaded voice modification part of the proposed data-driven optimization. Once we generate pseudonymized speech using speech superposition, the parameter optimization follows the same framework as the cascaded one.

In terms of privacy protection, this method is robust due to its irreversibility. This can be explained from the distribution of signals contained in the pseudonymized signal. Our framework superposes two signals into one, resulting in a single-channel signal as the output. Since the two signals are based on the same original signal, and we cannot assume the independency of each speech distribution, we expect our superposition method to be difficult against source separation tasks, thus causing great difficulties in recovering the original signal, making it an ill-posed problem.

Table 1
Parameter search range for each modification method.

Method	Parameter	Search range
VTLN	α_{vtln}	[−0.2, 0.2]
Resampling	α_{resample}	[0.7, 1.3]
McAdams trans.	α_{mcadams}	[0.7, 1.2]
MS Smoothing	α_{ms}	[0.05, 0.3]
Clipping	α_{clip}	[0.6, 1.0]
Chorus	α_{chorus}	[0.0, 0.2]

Table 2
Experimental results of tasks II, III*, and V obtained using cascaded voice modification for pseudonymization of Dev. set and Eval. set of VCTK male and female datasets. Regarding tasks II and V, the lower the better, whereas in task III*, the higher the better. R, M, MS, CH, and CL denote resampling, McAdams transformation, MS smoothing, chorus, and clipping, respectively.

	Development set			Evaluation set		
	1-EER (%)		WER (%)	1-EER (%)		WER (%)
Method	Task II	Task III*	Task V	Task II	Task III*	Task V
Original	97.9/97.1	–	12.6/13.8	98.0/95.0	–	15.6/17.8
Baseline 1 (x-vector)	53.4/45.5	68.8/74.5	17.5/18.6	46.4/51.8	68.4/69.0	18.5/19.6
Baseline 2 (McAdams trans. (α_{mcadams}=0.8))	71.2/63.7	87.9/82.3	23.3/35.0	71.9/70.1	87.8/83.0	29.4/38.4
R	53.6/56.0	78.1/**89.7**	21.0/**19.9**	61.0/68.5	79.8/**88.9**	26.7/**22.2**
R+CL	**46.7**/51.7	91.2/78.3	17.9/28.8	**51.6**/54.3	89.4/78.3	19.8/31.8
CH+M+R	47.9/52.8	**92.8**/86.3	**15.4**/25.9	53.6/61.4	**91.7**/82.4	**18.7**/29.2
R+CH+M+MS	47.8/**49.7**	89.5/83.1	18.9/21.4	52.5/**52.1**	89.8/78.5	24.3/36.1

5. Experiments

5.1. Experimental setup

The datasets used in our experiments were from VCTK (Veaux et al., 2019). For further details regarding the datasets, refer to the VoicePrivacy 2020 evaluation plans (Tomashenko et al., 2020a). To confirm the effectiveness of our proposed methods, we conducted tasks II, III, and V under the VoicePrivacy 2020 evaluation plans. Regarding task III, we replaced it with task III* as described in Section 3.5. Moreover, we changed the baseline pseudonymization design to instead follow our protocol; Baseline 1 involves choosing the same pseudo-speakers to pseudonymize enrollment and trial utterances within the same speaker. The setup for the proposed methods is as follows. We used the development set of the VCTK male dataset as the training data to search for the optimal parameter set. The optimized parameter set was then used to pseudonymize the evaluation set. On the basis of the results of our preliminary experiment, which was an investigation of speech pseudonymization under several settings using a single modification method, we set the weight ω of the objective function to 1.0 due to the little effect it has on the resulting pseudonymization performance. We used Optuna (Akiba et al., 2019), a software framework utilizing the Bayesian optimization algorithm, for parameter optimization, where the number of iterations for optimization was 50. For voice modification methods except the McAdams transformation, we used STFT with a 2048-sample window length, 2048-sample frame length, and 512-sample shift length. For the McAdams transformation, we followed an official implementation and its default settings (Tomashenko et al., 2020b). For zero-phase low pass filtering in MS smoothing, we first applied a second-order Butterworth filter in a forward time axis and then applied it in a backward time axis. For resampling, we used the waveform-similarity-based synchronized overlap-add algorithm implemented in audio time-scale modification (Verhelst and Roelands, 1993). Table 1 lists the parameter search range for each voice modification method. In our preliminary experiments, we changed the range of the parameters as well as the size of the training dataset. Regarding the range, since too narrow or too wide a range causes poor or computationally heavy optimization, we restricted the ranges as shown in Table 1 on the basis of our preliminary experiments. As for the size of the dataset, we randomly dropped approximately 80% of the trial data of each speaker and found only about 1 point differences in the resulting pseudonymization performance between the full and dropped data. Thus we used the reduced data to shorten the computation time for optimization. One iteration in optimization finished in less than 10 min in our environment with 16 CPUs.

For our first experiment, we used the cascaded voice modification method for speech pseudonymization. We applied pseudonymization to the input data using from a single method up to a combination of four methods. We evaluated a total of 179 combinations where the numbers of combinations of one, two, three, and four methods were 6, 23, 60, and 90, respectively. Since VTLN and the McAdams transformation both modify the spectrum of speech, we considered these two methods to be incompatible with each other. We also found that the results showed little improvement when we used VTLN and McAdams transformation for the same method in our preliminary experiments, and we decided to avoid including both methods in a single combination. In addition, since it was difficult to process the speech data after clipping, clipping was decided to be last in any combination that included clipping.

For our second experiment, we pseudonymized the input speech by using speech superposition. Apart from our first experiment evaluating combinations of up to four methods, we evaluated only two-method combinations for the purpose of assessing the

irreversible voice modification method. Since the modification procedure consists of superposing modified speech using single methods as explained in Section 4, the total number of combinations was 15 (two methods chosen from the six modification methods). The open-source implementation for both methods is available on our project page.[6]

We compared the results of our proposed methods with the two baseline systems introduced in VoicePrivacy 2020. Baseline 1 uses various machine learning techniques such as deep learning and speech synthesis for the pseudonymization procedure. First, linguistic and bottleneck features are extracted from the input speech followed by the extraction of the x-vector by using a pretrained DNN. Next, the x-vector is pseudonymized by using a pool of x-vectors that were extracted from the training data beforehand. Using the pseudonymized x-vector and the extracted linguistic and bottleneck features, the pseudonymized speech is synthesized. Baseline 2 is based on the McAdams transformation introduced in Section 3.3.2 where $\alpha_{\text{mcadams}} = 0.8$. The conditions of the baseline systems were the same in both experiments.

5.2. Experimental results

5.2.1. Pseudonymization using cascaded voice modification

Table 2 shows the best results of the cascaded voice modification method when using from a single method to up to a four-method combination. For comparison, we also show the results of the original and baseline systems introduced in VoicePrivacy 2020 in the same table. Tasks II and V should ideally be low, whereas task III* should be high as explained in Section 3.5. As shown in Table 2, the combination methods outperformed Baseline 2, which is the signal processing-based approach, in every task. We think this is due to the adequate parameter optimization of the cascaded voice modification method. Fig. 11 and Fig. 12 illustrates an example of the transition in WER and 1-EER obtained while searching for parameters using the cascaded voice modification on the male and female datasets, respectively. From both figures, we can see that as the number of methods increased, the search space expanded enabling more flexible and optimal selection of parameters. Regarding both the development set and evaluation set of the male dataset, some performances of the proposed methods were close to the machine learning-based Baseline 1, and, in some evaluation tasks, outperformed Baseline 1. Results for the female dataset show that some methods showed similar tendencies to those for the male one, while some did not. The results of the development set show a good degree of anonymization, but WER degradation was apparent in the case of three and four methods. The same can be said for task V of the evaluation set, with inferior results in anonymizing the speaker identity. From Fig. 12, we can see that the results obtained with the female data tended to scatter in the direction of a lower WER performance compared with the male results. Moreover, we cannot definitely claim that the search space expanded from the figure, illustrating the difficulties in optimizing loss in a pre-defined range. We think more consideration is needed to determine the range of parameters specifically for female data rather than using the same range for male data. In addition, more studies on Baseline 2 need to be done. Due to Baseline 2 modifying the waveform itself, distortion produced in the modification process will likely affect the verification results. Thus, when anonymized utterances are verified against original utterances in task II, the mismatch can cause a higher EER than expected.

Next, we compared the parameters of Baseline 1 and our proposed methods as well as measured the time it took to pseudonymize utterances. We used the same machine described in Section 5.1 for evaluating both methods. While Baseline 1 has over four million model parameters, our proposed method has only four parameters at most. Furthermore, our framework was able to pseudonymize the utterances much faster than Baseline 1. We compared the time it took to anonymize all the utterances in the male evaluation dataset, containing a total of 5420 utterances, for each method. Baseline 1 took approximately 189 minutes (2.09 s per utterance) whereas our framework took approximately 21 minutes (0.24 s per utterance) to pseudonymize the evaluation set, reducing the time of the process by 89%. From these results, we can consider our method to be lightweight.

Fig. 13 shows a box diagram of the objective function values (the loss, WER, and 1-EER) with respect to number of methods used for pseudonymization. From the distribution plot of the loss shown in the left diagram, as the number of methods used in the pseudonymization increased, the average of the objective function value lowered, indicating better minimization results for the objective. We can also see that the WER had a tendency to degrade, while the 1-EER tended to improve as the number of methods increased. Although we can attribute the degradation in the quality of the speech to the speech being modified multiple times with the combination methods, there may be a suggestion of a trade-off between the WER and 1-EER. Furthermore, to investigate the possibility of the cascade method overfitting the development set, we computed the difference in the objective function values between the development set and evaluation set. The distribution of the difference is shown in Fig. 14. Here, the difference in objective function values means that we have subtracted the objective function value of the evaluation set from that of the development set. In the case where less or no overfitting is seen, it is desirable that the difference be in the vicinity of or below zero. From Fig. 14, the 1-EERs are mostly distributed near zero, while the WERs are plotted across a wide range. However, the mean approached zero as the number of methods increased, mitigating the difference in the objective function value between the development and evaluation sets. We conducted further analysis on the importance of each method to the combination by focusing on the Gini impurity. Further descriptions regarding the Gini impurity as well as our insights on this subject are located in Appendix.

[6] https://github.com/sarulab-speech/lightweight_spkr_anon.

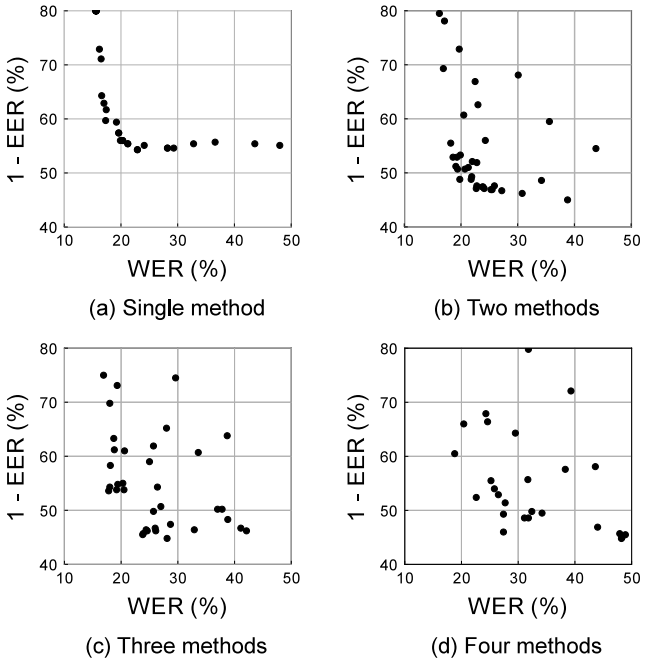


Fig. 11. Transition in WER and 1-EER obtained while searching for parameters by using cascaded voice modification for male dataset. Graph (a) is one method, resampling, (b) is two methods, resampling and MS smoothing, (c) is three methods, resampling, MS smoothing, and McAdams transformation, and (d) is four methods, resampling, MS smoothing, McAdams transformation, and chorus.

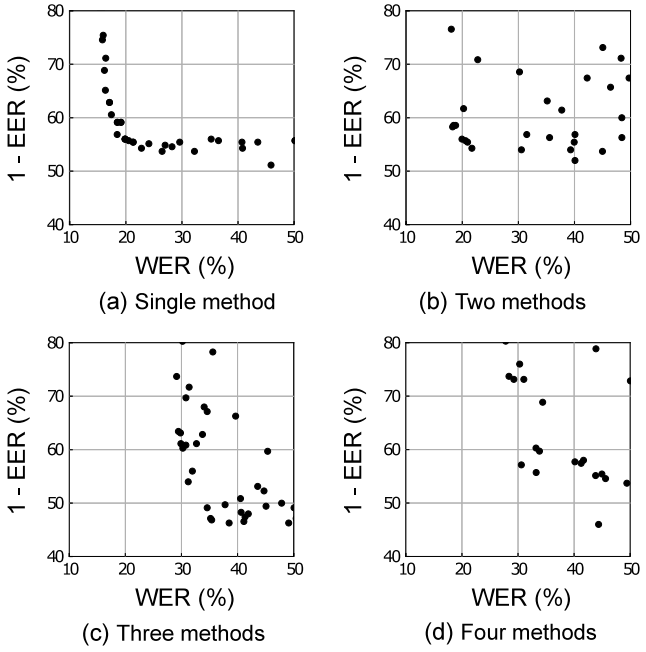


Fig. 12. Transition in WER and 1-EER obtained while searching for parameters by using cascaded voice modification for female dataset. Graph (a) is one method, resampling, (b) is two methods, resampling and clipping, (c) is three methods, chorus, McAdams transformation, and resampling, and (d) is four methods, resampling, chorus, McAdams transformation, and MS smoothing.

5.2.2. Pseudonymization using irreversible voice modification

Table 3 shows an excerpt from the results obtained using the speech superposition method for voice modification. Similar to the cascaded voice modification method results, methods achieving superior results using the development set tended to achieve

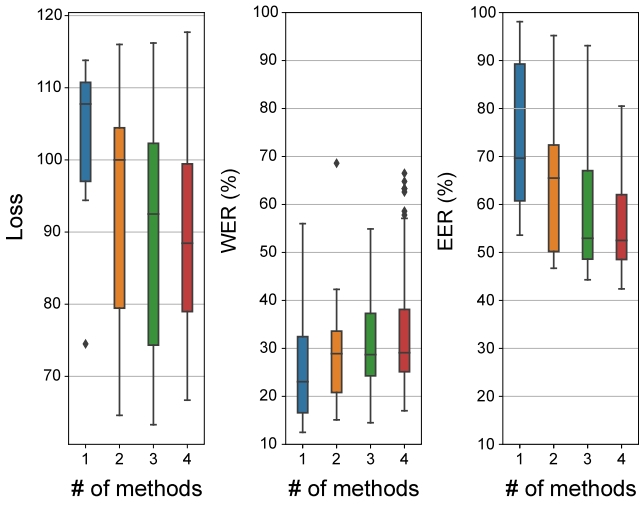


Fig. 13. Distribution of objective function values with respect to number of methods. Left diagram shows distribution of loss, middle diagram shows WER, and right diagram shows 1-EER.

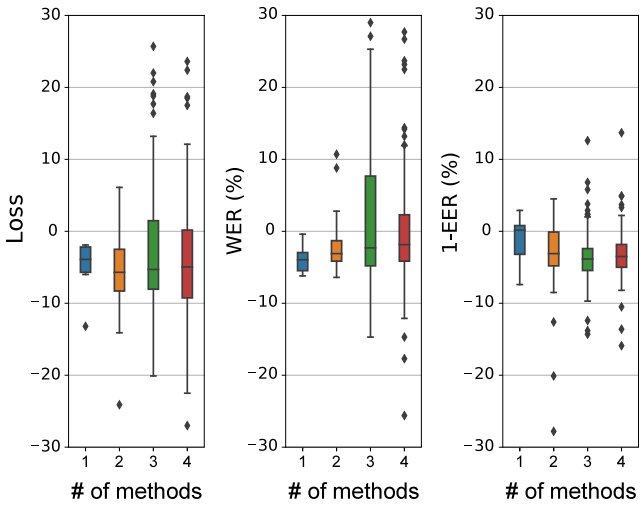


Fig. 14. Distribution of differences in objective function values with respect to number of methods. Left diagram shows distribution of loss, middle diagram shows WER, and right diagram shows 1-EER.

superior results using the evaluation set. We also found that the best methods mostly match between the male and female. However, the best performances of the speech superposition were worse in every task compared to the cascade method. This shows that the superposition method are more likely to distort modified speech than the cascaded one. Nevertheless, it achieved better performance than Baseline 2. To investigate the different tendencies between the cascaded voice modification and the irreversible modification, each search space is shown in Fig. 15. We can see a difference in the search space despite the same number and type of methods being used in pseudonymization. From these figures, the search space of the superposition method tended to be narrow, implying more difficulty in optimizing the parameters than in the cascaded methods.

6. Conclusion

In this paper, we proposed a speech pseudonymization framework based on two voice modification methods for protecting the privacy of speech. We evaluated our proposed methods following the VoicePrivacy 2020 objective evaluation tasks. Despite our methods being simple with very few parameters to optimize, it is possible to achieve better performance in certain tasks than with machine learning-based methods by parameter optimization. From the results of the experiments, we found that combining more than two methods is more effective in pseudonymizing speech without compromising its intelligibility, greatly surpassing Baseline 2 and achieving performance close to Baseline 1.

Table 3
Experimental results of tasks II, III*, and V obtained using irreversible voice modification for pseudonymization of Dev. set and Eval. set of VCTK male and female datasets. Regarding tasks II and V, the lower the better, whereas in task III*, the higher the better.

	Development set			Evaluation set		
	1-EER (%)		WER (%)	1-EER (%)		WER (%)
Method	Task II	Task III*	Task V	Task II	Task III*	Task V
Original	97.9/97.1	–	12.6/13.8	98.0/95.0	–	15.6/17.8
Baseline 1 (x-vector)	53.4/45.5	68.8/74.5	17.5/18.6	46.4/51.8	68.4/69.0	18.5/19.6
Baseline 2 (McAdams trans. (α_{mcadams}=0.8))	71.2/63.7	87.9/82.3	23.3/35.0	71.9/70.1	87.8/83.0	29.4/38.4
Resampling + McAdams trans.	**50.7**/**56.3**	87.1/85.4	**18.4**/**21.1**	**53.4**/**64.8**	86.3/82.9	**22.0**/**23.1**
Resampling + VTLN	54.0/57.7	**89.3**/85.7	23.8/35.0	54.3/69.5	**89.0**/**87.1**	34.2/36.4
Resampling + Clipping	65.9/70.9	87.6/72.3	31.4/27.5	71.9/73.6	88.6/86.3	33.8/33.4
VTLN + Clipping	65.0/71.7	84.5/**86.9**	25.9/25.8	73.2/74.5	80.9/81.4	27.0/27.0

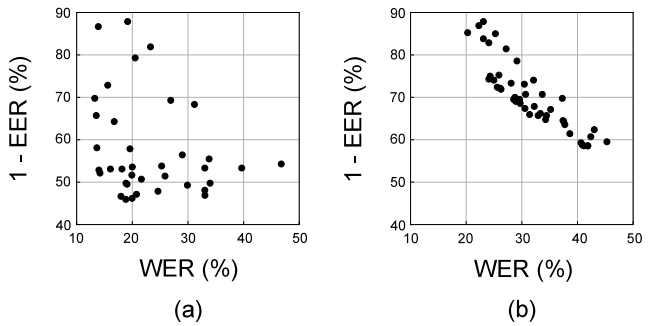


Fig. 15. Transition in WER and 1-EER obtained while searching for parameters by using resampling and clipping as modification methods. Graph (a) uses cascaded voice modification, and (b) uses speech superposition.

Future work involves assuming more practical situations by (1) evaluating our proposed methods using different models and data, (2) carrying out further evaluations, i.e., original task III protocol and subjective evaluations, and (3) evaluating our framework under different attack scenarios. Considerations for (1) include comparing our proposed method with different systems under the same training conditions. Such experiments could support our claim about the lightweight character of our proposed method. Regarding (3), the attacker's knowledge has not been considered in most speaker verification research (Srivastava et al., 2020b). Therefore, we would need to evaluate the robustness of our framework when partial information of the system has been accessed by the adversary. In terms of evaluating the degree of privacy, we would also need to consider effective metrics for evaluating the irreversibility of pseudonymization methods. Moreover, further analysis of the obtained results could shed more light on how combining methods affect the result. In particular, it would be interesting to see how the order of methods would contribute to decreasing the Gini impurity.

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.

Acknowledgments

This work was supported in part by JSPS KAKENHI Early Career Scientists Grant number JP19K20271, ROISDS-JOINT (023RP2020) to S. Shiota, and SECOM Science and Technology Foundation, Japan.

Appendix. Measuring importance of modification methods using Gini impurity

We analyzed each modification method's contribution in terms of the likeliness of improving the loss by calculating the Gini impurity (GI). We labeled each result, or sample, as Better or Worse. Better indicates that the loss of the sample was improved by adding the method in question to the combination, whereas Worse means the loss degraded. By calculating GI, we can quantitatively measure the trends in loss improvement when certain modification methods are added to a combination method. We illustrate a

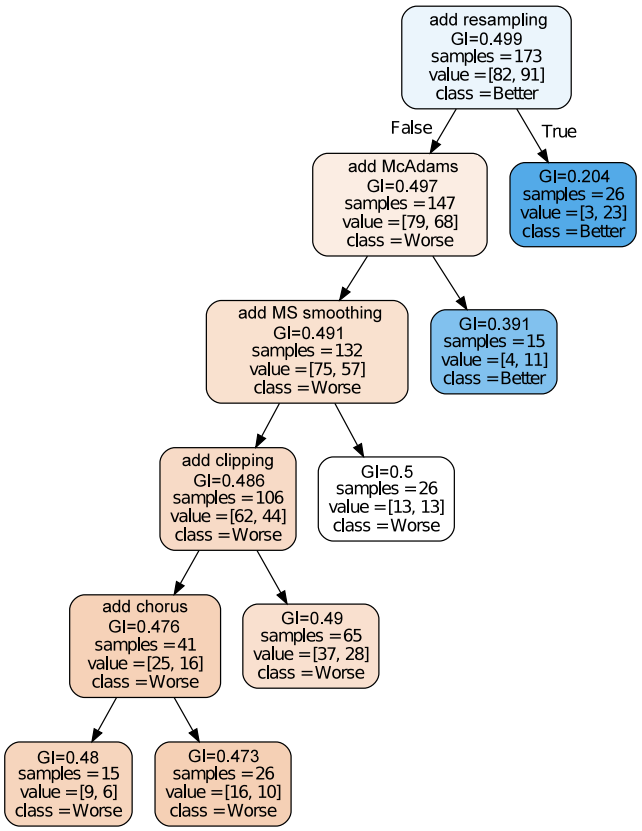


Fig. 16. Decision tree for measuring likelihood of loss improvement on basis of Gini impurity (GI) when certain modification methods are added. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

decision tree, computed using the 173 combination methods as samples, in Fig. 16. Each node contains, from top to bottom, the condition, GI, the overall samples in the node, the number of samples in each class, and the class label with the most samples. Note that, in this case, class refers to Better and Worse. To calculate GI, we used the first factor, condition, to decide whether or not to add a method. The second factor, GI, was computed between 0 and 0.5 on the basis of the number of samples classified in each class, where 0 represents that all of the samples in the node were classified into one class. From the fourth factor, we can see how each sample was classified, with the left and right numbers indicating the classes Worse and Better, respectively. For example, the blue node that branches from the first node of Fig. 16 shows the results when resampling was added last to the combination. Since the number of samples was 26, there were 26 methods out of 173 that included resampling as the last method. From the value, we can see that the overall methods, specifically 23 methods, were classified as Better, resulting in the GI being low at a value of 0.204. Ultimately, we wanted to find a method that would decrease the GI with the overall samples classified as Better. From Fig. 16, we consequently show that adding resampling and McAdams transformation to a combination lowers GI to certain degree, greatly contributing to loss improvement with a higher tendency to improve the result. The other methods made only a mild contribution to loss improvement, but they were likely to degrade the result. As for the VTLN, no contribution was seen in terms of the likelihood of improving the loss.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M., 2019. Optuna: A next-generation hyperparameter optimization framework. In: Proc. ACM SIGKDD. Alaska, U.S.A. 2019. pp. 2623–2631.

Bahmaninezhad, F., Zhang, C., Hansen, J., 2018. Convolutional neural network based speaker de-identification. In: Proc. Odyssey. Les Sables d'Olonne, France, pp. 255–260.

Champion, P., Jouvet, D., Larcher, A., 2020. Speaker Information Modification in the VoicePrivacy 2020 Toolchain. Research Report INRIA Nancy, équipe Multispeech, LIUM-Laboratoire d'Informatique de l'Université du Mans.

Dubagunta, S.P., Van Son, R.J., Doss, M.M., 2020. Adjustable deterministic pseudonymisation of speech: Idiap-NKI's submission to VoicePrivacy 2020 challenge. In: VoicePrivacy 2020 Challenge.

Espinoza-Cuadros, F.M., Perero-Codosero, J.M., Antón-Martín, J., Hernández-Gómez, L.A., 2020. Speaker de-identification system using autoencoders and adversarial training. In: VoicePrivacy 2020 Challenge.

Fang, F., Wang, X., Yamagishi, J., Echizen, I., Todisco, M., Evans, N., Bonastre, J.-F., 2019. Speaker anonymization using x-vector and neural waveform models. In: Proc. 10th ISCA Speech Synthesis Workshop. Vienna, Austria, pp. 155–160.

Farooq, F., Bolle, R.M., Jea, T.-Y., Ratha, N., 2007. Anonymous and revocable fingerprint recognition. In: Proc. IEEE Conference on CVPR. Minnesota, U.S.A. pp. 1–7.

Garcia-Romero, D., Snyder, D., Sell, G., McCree, A., Povey, D., Khudanpur, S., 2019. x-Vector DNN refinement with full-length recordings for speaker recognition. In: Proc. Interspeech. Graz, Austria, pp. 1493–1496.

Hintze, M., El Emam, K., 2018. Comparing the benefits of pseudonymisation and anonymisation under the GDPR. J. Data Prot. Priv. 2 (2), 145–158.

Itakura, F., 1968. Analysis synthesis telephony based on the maximum likelihood method. In: The 6th International Congress on Acoustics. pp. 280–292.

Kai, H., Takamichi, S., Shiota, S., Kiya, H., 2021. Lightweight voice anonymization based on data-driven optimization of cascaded voice modification modules. In: Proc. IEEE SLT Workshop. pp. 560–566, Virtual.

Lagendijk, R.L., Erkin, Z., Barni, M., 2013. Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation. IEEE Signal Process. Mag. 30 (1), 82–105.

Lee, L., Rose, R., 1998. A frequency warping approach to speaker normalization. IEEE Trans. Speech Audio Process. 6 (1), 49–60.

Mawalim, C.O., Galajit, K., Karnjana, J., Unoki, M., 2020. X-vector singular value modification and statistical-based decomposition with ensemble regression modeling for speaer anonymization system. In: Proc. Interspeech. pp. 1703–1707, Virtual.

Nautsch, A., Patino, J., Tomashenko, N., Yamagishi, J., Noe, P.-G., Bonastre, J.-F., Todisco, M., Evans, N., 2020. The privacy ZEBRA: Zero evidence biometric recognition assessment, a speaker recognition perspective. In: Proc. Interspeech. pp. 1698–1702, Virtual.

Noé, P.-G., Bonastre, J.-F., Matrouf, D., Tomashenko, N., Nautsch, A., Evans, N., 2020. Speech pseudonymisation assessment using voice similarity matrices. In: Proc. Interspeech. pp. 1718–1722, Virtual.

Panayotov, V., Chen, G., Povey, D., Khudanpur, S., 2015. Librispeech: an ASR corpus based on public domain audio books. In: Proc. ICASSP. Brisbane, Australia, pp. 5206–5210.

Patino, J., Tomashenko, N., Todisco, M., Nautsch, A., Evans, N., 2020. Speaker anonymisation using the McAdams coefficient.

Peddinti, V., Povey, D., Khudanpur, S., 2015. A time delay neural network architecture for efficient modeling of long temporal contexts. In: Proc. Interspeech. Dresden, Germany, pp. 214–3218.

Snoek, J., Larochelle, H., Adams, R.P., 2012. Practical bayesian optimization of machine learning algorithms. In: Proc. NIPS. Nevada, U.S.A. pp. 2951–2959.

Sridharan, S., Dawson, E., Goldburg, B., 1991. Fast Fourier transform based speech encryption system. IEE Proc. I 138 (3), 215–223.

Srivastava, B.M.L., Tomashenko, N., Wang, X., Vincent, E., Yamagishi, J., Maouche, M., Bellet, A., Tommasi, M., 2020. Design choices for x-vector based speaker anonymization. In: Proc. Interspeech. Shanghai, China, pp. 1713–1717.

Srivastava, B.M.L., Vauquier, N., Sahidullah, M., Bellet, A., Tommasi, M., Vincent, E., 2020b. Evaluating voice conversion-based privacy protection against informed attackers. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, IEEE, pp. 2802–2806.

Takamichi, S., Kobayashi, K., Tanaka, K., Toda, T., Nakamura, S., 2015. The NAIST text-to-speech system for the blizzard challenge 2015. In: Proc. Blizzard Challenge workshop. Berlin, Germany.

Tomashenko, N., Srivastava, B.M.L., Wang, X., Vincent, E., Nautsch, A., Yamagishi, J., Evans, N., Patino, J., Bonastre, J.-F., Noé, P.-G., Todisco, M., 2020. Introducing the VoicePrivacy initiative. In: Proc. Interspeech. pp. 1693–1697, Virtual.

Tomashenko, N., Srivastava, B.M.L., Wang, X., Vincent, E., Nautsch, A., Yamagishi, J., Evans, N., Patino, J., Bonastre, J.-F., Noé, P.-G., et al., 2020b. The VoicePrivacy 2020 challenge evaluation plan. https://www.voiceprivacychallenge.org/docs/VoicePrivacy_2020_Eval_Plan_v1_3.pdf.

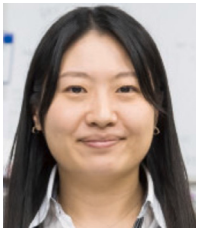
Veaux, C., Yamagishi, J., MacDonald, K., 2019. CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit (version 0.92). University of Edinburgh. The Centre for Speech Technology Research (CSTR).

Verhelst, W., Roelands, M., 1993. An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech. In: Proc. IEEE ICASSP. Minnesota, U.S.A. pp. 554–557.

Xu, W., He, Q., Li, Y., Li, T., 2008. Cancelable voiceprint templates based on knowledge signatures. In: Proc. International Symposium on Electronic Commerce and Security. Guangzhou City, China, pp. 412–415.

Hiroto Kai was born in 1997. He received his B.E. degree from Tokyo Metropolitan University, Tokyo, Japan in 2020. His research interests include speaker verification and voice privacy protection.

Shinnosuke Takamichi received his B.E. degree from the Nagaoka University of Technology, Nagaoka, Japan, in 2011, and M.E. and Ph.D. degrees from the Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Japan, in 2013 and 2016, respectively. He is currently an Assistant Professor at The University of Tokyo. He has received more than 20 paper/achievement awards including the 2020 IEEE Signal Processing Society Young Author Best Paper Award.



Sayaka Shiota received her B.E., M.E., and Ph.D. degrees in intelligence and computer science, engineering, and engineering simulation from the Nagoya Institute of Technology, Nagoya, Japan, in 2007, 2009, and 2012, respectively. From February 2013 to March 2014, she worked at the Institute of Statistical Mathematics as a Project Assistant Professor. In April of 2014, she joined Tokyo Metropolitan University as an Assistant Professor. Her research interests include statistical speech recognition and speaker verification. She is a member of ASJ, IPSJ, IEICE, APSIPA, ISCA, and IEEE.

Hitoshi Kiya received his B.E and M.E. degrees from Nagaoka University of Technology in 1980 and 1982, respectively, and his Dr. Eng. degree from Tokyo Metropolitan University in 1987. In 1982, he joined Tokyo Metropolitan University, where he became a Full Professor in 2000. From 1995 to 1996, he attended the University of Sydney, Australia, as a Visiting Fellow. He is a Fellow of IEEE, IEICE, and ITE. He currently serves as President-Elect of APSIPA, and he served as Inaugural Vice President (Technical Activities) of APSIPA from 2009 to 2013 and as Regional Director-at-Large for Region 10 of the IEEE Signal Processing Society from 2016 to 2017. He was also President of the IEICE Engineering Sciences Society from 2011 to 2012, and he served there as a Vice President and Editor-in-Chief for IEICE Society Magazine and Society Publications. He was Editorial Board Member of eight journals, including IEEE Trans. on Signal Processing, Image Processing, and Information Forensics and Security, Chair of two technical committees, and Member of nine technical committees including the APSIPA Image, Video, and Multimedia Technical Committee (TC) and IEEE Information Forensics and Security TC. He has organized a lot of international conferences in such roles as TPC Chair of IEEE ICASSP 2012 and as General Co-Chair of IEEE ISCAS 2019. He has received numerous awards, including six best paper awards.