electronics

MDPI

Article

Black-Box Watermarking and Blockchain for IP Protection of Voiceprint Recognition Model

Jing Zhang [1], Long Dai [1], Liaoran Xu [1], Jixin Ma [2] and Xiaoyi Zhou [1,*]

[1] School of Cyberspace Security, Hainan University, Haikou 570228, China; zj093327@163.com (J.Z.); 21210839000005@hainanu.edu.cn (L.D.); xuliaoran_312@163.com (L.X.)

[2] School of Computing and Mathematical Sciences, Faculty of Engineering and Science, University of Greenwich, London SE10 9LS, UK; j.ma@gre.ac.uk

* Correspondence: xy.zhou@hainanu.edu.cn

Abstract: Deep neural networks are widely used for voiceprint recognition, whilst voiceprint recognition models are vulnerable to attacks. Existing protection schemes for voiceprint recognition models are insufficient to withstand various robustness attacks and cannot prevent model theft. This paper proposes a black-box voiceprint recognition model protection framework that combines active and passive protection. It embeds key information into the Mel spectrogram to generate trigger samples that are difficult to detect and remove and injects them into the host model as watermark W, thereby enhancing the copyright protection performance of the voiceprint recognition model. To restrict the use of the model by unauthorized users, the index number corresponding to the model and the encrypted model information are stored on the blockchain, and then, an exclusive smart contract is designed to restrict access to the model. Experimental results show that this framework effectively protects voiceprint recognition model copyrights and restricts unauthorized access.

Keywords: copyright protection; voiceprint recognition model; watermarking; blockchain; black-box; Mel spectrogram

check for updates

Citation: Zhang, J.; Dai, L.; Xu, L.; Ma, J.; Zhou, X. Black-Box Watermarking and Blockchain for IP Protection of Voiceprint Recognition Model. *Electronics* **2023**, *12*, 3697. https://doi.org/10.3390/electronics12173697

Academic Editor: Byung Cheol Song

Received: 31 July 2023
Revised: 20 August 2023
Accepted: 22 August 2023
Published: 1 September 2023

1. Introduction

Deep learning technology has enabled neural networks to provide powerful support and new development opportunities in speech recognition [1], image recognition [2], natural language processing [3], etc. However, deep neural network (DNN) models are expensive compared to traditional multimedia data. Training a DNN model for a specific task requires not only massive amounts of training data but also a large amount of hardware resources and professional knowledge [4]. Therefore, well-trained neural network models have significant commercial value and intellectual property (IP) attributes. It is a pressing focus area in the field of deep learning to effectively protect the copyright of a neural network model.

Digital watermarking [5–7] is a technology used to hide copyright information in digital media and extract it for ownership proof. Based on this characteristic, scholars introduced watermarking technology into the copyright protection of deep neural network models [8–11]. An important requirement for neural network watermarking is that it should not influence the original task of the host model. Therefore, the proposed watermarking framework should be imperceptible, effective, secure, and robust to attacks, such as fine-tuning and pruning [12]. Currently, white-box watermarking and black-box watermarking are the most widely used DNN watermarking schemes [13–17]. The model owner embeds a watermark into the host model, and the internal structure and weight parameters of the host model can be known only when the watermark is extracted. Thus, in white-box watermarking [18–21], it is assumed that the model owner verifies the ownership of the suspicious model merely by accessing its internal structure and weight parameters and extracting the embedded watermark. In black-box watermarking [22,23], the model owner applies neural network backdoors to train the model and verifies the model's copyright by constructing a set of triggers with specific input-output pairs.

Although there are already many model watermarking schemes available for copyright protection of neural network models, most of them are designed for neural network models used in image classification and image processing tasks [24–27]. In terms of copyright protection for audio models, the relevant literature investigated databases such as Web of Science (Web of Science is an internationally renowned literature search platform, which widely includes high-quality academic journals, conference papers, and other literature resources worldwide, and is considered one of the world's most authoritative scientific citation databases) and CNKI (CNKI is the China Knowledge Network, a comprehensive academic literature information service platform with massive Chinese literature resources covering a wide range of fields with important academic value and practicality), and only two were found [28,29]. Chen et al. [28] proposed a white-box watermarking scheme to protect speech recognition models by extending the watermark to multiple random subsets of important frequency components, identifying the significant frequency components of the model parameters for ownership proof. This scheme enables the model owner to extract the watermark information inside the suspicious model and verify their ownership of the marked speech recognition model. Experimental results show that the watermarking framework has minimal overhead and retains the recognition accuracy of the original speech model. However, speech recognition only recognizes "what is said" by sound and not "who is speaking"; this scheme does not apply to voiceprint recognition. To explore copyright protection schemes applicable to voiceprint recognition models, Wang et al. [29] proposed a black-box frequency domain watermarking framework for voiceprint recognition. It adds trigger signals to the frequency domain of the original audio samples to construct trigger audio samples. To enable the model to successfully learn the mapping relationships of the trigger samples, a watermark sequence is embedded in each selected segment of the frequency domain of the audio signal to generate a trigger set. Experimental results demonstrated that it could effectively protect the copyright of voiceprint recognition models. However, it neglects robust attacks and security attacks, which may allow adversaries to remove the owner's watermark through pruning, fine-tuning, and ambiguity attacks [30].

From the above analysis, it can be seen that copyright protection for audio processing models is still in its infancy. Although white-box and black-box watermarking schemes are proposed in the literature to protect such models, white-box watermarking is only suited for speech recognition models and does not apply to voiceprint recognition models. The black-box watermarking scheme lacks both robustness and security. Furthermore, they only verify ownership after the model is stolen, without taking into consideration the model's security issues. To address these concerns, this work introduces a voiceprint recognition model protection framework that integrates active and passive protection mechanisms to safeguard the model's copyright. The proposed scheme uses the Mel spectrogram [31] to generate hidden trigger samples that are similar to the original audio sample's distribution, thus avoiding abnormal detection by attackers and enhancing the watermarking scheme's security and robustness. Additionally, to limit the use of illegal users, blockchain technology [32,33] is introduced to store the index number and encrypted information of the protected model, and then, smart contracts [34] are used to record, store, manage, and verify user identity. This paper primarily contributes the following:

(1) A black-box watermarking framework is proposed based on the Mel spectrogram to achieve copyright protection of speaker recognition models. It embeds the watermark information into the Mel spectrogram instead of directly embedding it into the original speech signal. This approach effectively improves the robustness and security of the watermark information;

(2) A proactive defense framework based on blockchain access control is designed. This framework centers around the copyright owner, and visitors must obtain permission

from the owner to share the model. By restricting the use of the model by unauthorized users, it achieves proactive protection of the speaker recognition model;

(3) Experimental results demonstrate the effectiveness of the proposed watermarking framework.

The remaining parts of this paper are as follows: Section 2 briefly introduces related background knowledge; Section 3 describes the implementation of the proposed framework in detail; Section 4 presents experimental results and analysis; and finally, Section 5 summarises the findings of the paper.

2. Background

This section introduces relevant background knowledge closely related to this work, such as blockchain technology, neural network backdoor technology, Mel spectrogram, and watermark embedding algorithm.

2.1. Blockchain Technology

Blockchain [35] is used to restrict access to the model, ensuring data security and privacy through the use of digital signatures [36] and smart contracts and enabling access control-based [37] resource authorization management. The specific process is summarized as follows:

Step 1: The model owner encrypts information with their public key and stores it on the blockchain;

Step 2: Once a user passes the smart contract verification, the blockchain notifies the model owner, then re-encrypts the resource information with the public key of the authorized user;

Step 3: The authorized user utilizes their private key to decrypt and use the resource.

By utilizing blockchain technology to implement access control, a reliable solution is provided for the security and privacy of model resources.

2.2. Neural Network Backdoor

Neural network backdoor [38] is a type of model attack technology that adds perturbation to dataset X of the model, causing the model to output a specific label T. The combination $B = (X_\delta, T)$ of input X is with added perturbation, and specific output label T is called a backdoor. Neural network models embedded in backdoors can have serious consequences [39], such as autonomous driving systems classifying stop signs as speed limit signs by putting sticky notes on them, which can lead to accidents. However, researchers have found that neural network backdoors can also effectively protect the IP [40] of neural network models. Model owners can embed their backdoor into the model and verify their ownership by querying the returned labels of suspicious models. This is shown in Figure 1. Currently, there are two relatively mature methods [41,42] for constructing trigger sets that can lead to problems such as misclassification and blurred model ownership boundaries. To address these problems, researchers combined the two methods and proposed assigning additional labels to triggers [43], thereby converting the original n-classification problem into an n + 1 classification one. This method does not result in misclassification or blurred boundaries and does not affect the model's original accuracy.

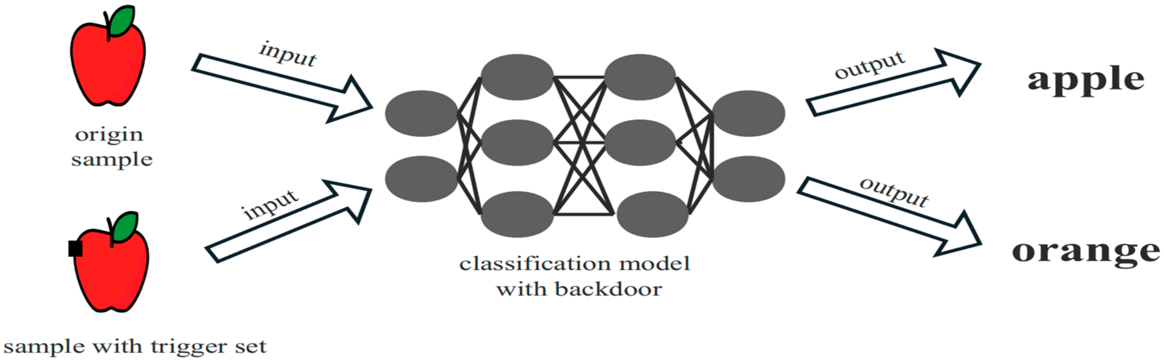


Figure 1. Neural network backdoor example: the original sample was correctly classified as "apple" by the backdoor network, and the sample with trigger set was incorrectly classified as "orange".

2.3. Spectrogram and Mel Spectrogram

Mel spectrograms are a time-frequency domain analysis method that can better describe the characteristics of audio signals. Due to their nonlinear frequency axis and logarithmic scale, Mel spectrograms have broad prospects in the field of digital watermarking. The Mel scale is defined as converting frequency f to $2595 \times ln(1 + f/700)$, where f is the frequency, and 700 is the lowest frequency that the human ear can perceive. In the Mel spectrogram, the low-frequency part is denser than the high-frequency part, which is more consistent with human perception of tones since the human ear is more sensitive to low-frequency parts. By embedding watermark information into the Mel spectrogram, it is possible to avoid the human auditory system and create trigger samples that cannot be distinguished by the human ear while also resisting some channel noise and distortion. Therefore, Mel spectrograms have become a commonly used tool in audio processing tasks such as speech recognition, voiceprint recognition, and emotion recognition.

2.4. Watermark Embedding Algorithm

To construct a hidden trigger set, this paper proposes embedding watermarks in Mel spectrograms to create trigger samples. Common watermarking algorithms [44] include transform-domain algorithms and spatial-domain algorithms. Spatial-domain algorithms directly modify the pixel values of the original image for watermark embedding. The Least Significant Bit (LSB) algorithm [45] is a typical spatial domain algorithm that is simple but less transparent. The transform domain algorithm embeds the watermark by transforming the pixel coefficients to the frequency domain and modifying the frequency domain coefficients [46], which is more robust and transparent. For the reason that the created trigger sample resembles the original sample whilst remaining distinguishable by a voiceprint recognition model, this paper combines these two algorithms. Firstly, transforming the pixel values of the Mel spectrogram into the discrete cosine transform domain and then using the LSB algorithm for embedding.

3. Proposed Method

This scheme comprises two components, the watermark network, and the blockchain network, as shown in Figure 2.

The watermark network consists of three stages: watermark generation; watermark embedding; and watermark verification. Watermark generation involves extracting the Mel spectrogram of the original audio signal, embedding the owner's identity information, and converting it into a speech signal to be used as a trigger sample for subsequent watermark embedding and verification. Watermark embedding is accomplished by training a model with a set of original audio samples and a set of trigger samples simultaneously. During training, the original audio samples output their original labels, while the trigger samples output predefined labels. After training, the host model will be marked and will obtain a watermarked model. During watermark verification, a new set of trigger samples is used as input to query suspicious models. If the output belongs to the label class defined by the owner, the ownership is verified.

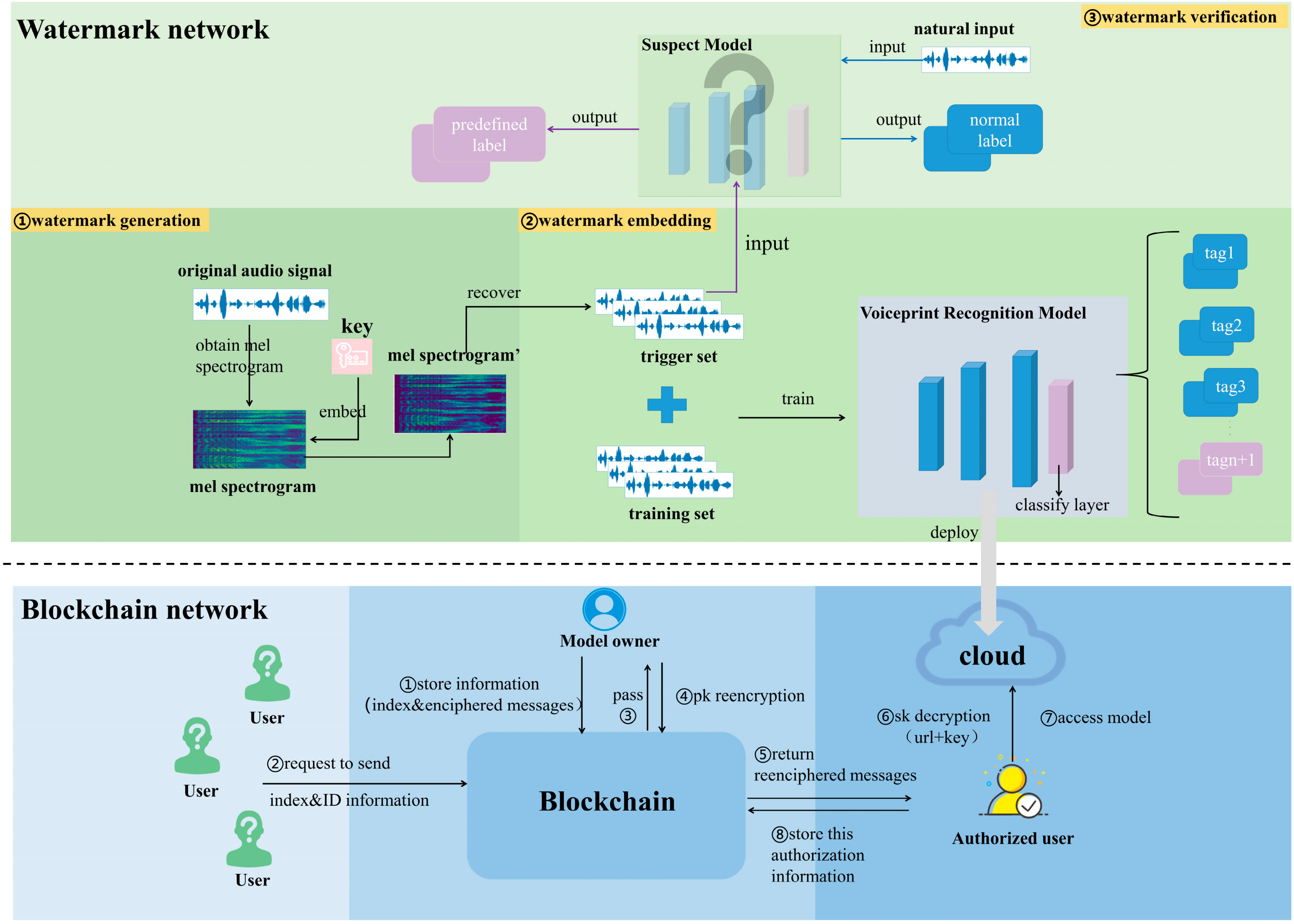


Figure 2. The specific framework of the proposed watermarking scheme.

In the blockchain network, the model owner stores the encrypted model address and password on the blockchain and designs an exclusive smart contract to restrict the usage rights of the model. When a user needs to use the model, they can obtain usage permission through blockchain verification with their identity information. The authorization information is also recorded on the blockchain for traceability.

3.1. Blockchain Network

This paper proposes a proactive protection framework for models based on blockchain technology, as illustrated in Figure 2. First, the model owner encrypts the trained voiceprint recognition model and stores it in the cloud. The model's address (URL) and usage password (key) are encrypted and stored on the blockchain, along with a unique index number assigned to the protected model. Second, this framework provides exclusive access control permissions uniquely designed for model owners. When a user needs to use the model, they send a request to the blockchain, including the index number of the model they want to access and their identity information. After receiving the user's request, the blockchain node calls the predefined smart contract to verify whether the user's identity information matches.

If the verification is successful, the blockchain will notify the model owner, who then re-encrypts the model password using the user's public key (pk) and returns it to the blockchain. Authorized users can decrypt it using their private key (sk) and use the model with the returned URL and decrypted key. At the same time, this authorization information will be recorded on the blockchain. Every time a user uses the model, the authorization information will be updated to record the usage situation. If someone is found to have illegally stolen or used the model without authorization, they can be traced and held responsible through the authorization information recorded on the blockchain.

3.2. Watermark Generation

The process of watermark generation is shown in Figure 3. The model owner randomly selects m audio samples from n categories in a clean dataset $D_{train} = \{(X_s, Y_s)\}_{s=1}^{n}$, and embeds key information Key into the Mel spectrogram of each selected audio sample. The Mel spectrogram is then restored to an audio signal to construct a trigger set $D_{trigger} = \{(X_i, Y)\}_i^m$, where X_i is the watermarked audio signals and Y is the contain pre-defined labels. Assuming M is the Mel spectrogram obtained from the original speech signal, it can be viewed as a simple two-dimensional matrix. The Key is embedded in this matrix, where the Key is a binary image of size $m \times n$. After the dimensionality reduction in Key to obtain a binary sequence of length m, denoted as w, w is used as the watermark for embedding. For the selection of embedding positions, the researchers propose that embedding in frequency bands where the pixel values in the spectrogram are lower than 6.3778 can achieve a better balance between robustness and imperceptibility.

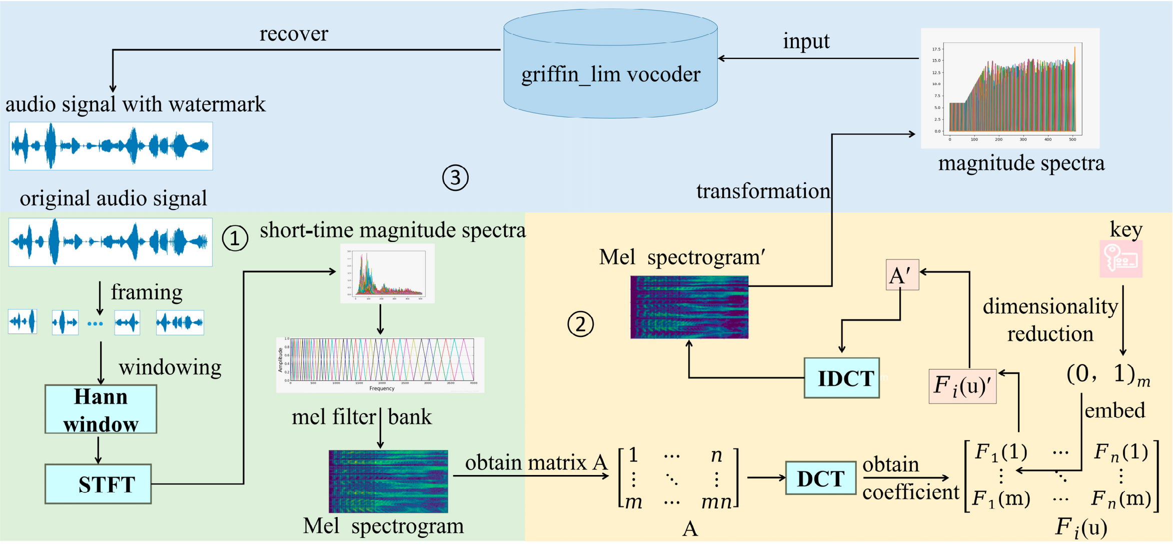


Figure 3. The process of watermark generation.

Therefore, in this paper, a matrix $A(X, Y) \in \{0, 255\}^{m \times n}$ of the same size as the embedded Key is randomly selected from the mid-frequency band of the Mel spectrogram for Key embedding. During embedding, the matrix A is transformed row by row using the following formula via DCT transformation:

$$F(u) = c(u) \sum_{i=0}^{m-1} A(Y) cos\left[\frac{(i + 0.5)\pi}{m} u\right] \tag{1}$$

where

$$c(u) = \begin{cases} \sqrt{\frac{1}{m}} \Big(& , \quad u = 0 \\ \sqrt{\frac{2}{m}} & , \quad u \neq 0 \Big) \end{cases} \tag{2}$$

The coefficients after each row transformation are denoted as $F(u) = \{F_i(u) | i = 1, 2, \ldots, n;\ u = 1, 2, \ldots, m\}$, where u represents the row number, and i represents each element in the row. The first element of each row is the DC coefficient, and the rest are AC coefficients.

We extract the DC coefficients from each row and embed watermarks into them using the following formula:

$$F_1(k)' = F_1(k) + \alpha \times w(k) \qquad (k = 1, 2, 3, \ldots, m) \tag{3}$$

Here, α is the embedding coefficient, and $F_1(k)'$ can be obtained to find the watermarked matrix $A'(m \times n)$. The IDCT transformation is performed on A', and the watermarked Mel spectrogram M_{WM} is obtained. The griffin_lim [47] vocoder algorithm is used to restore the audio signal, which now contains watermark information.

3.3. Watermark Embedding

The goal of watermark embedding is to obtain a model with a watermark. To avoid blurred decision boundaries and misclassifications, this paper assigns additional labels to triggers, changing the n-classification problem to an n + 1 class problem by altering the output class of the classification layer. As the added label is only associated with trigger samples, the false positive rate is low. A new dataset $D = D_{train} \cup D_{trigger}$ is used during training. Specifically, in the training phase, the original audio samples and trigger audio samples are placed together into the model for training. Each original audio sample corresponds to the original label, and all trigger audio samples correspond to the added label. The speaker recognition model will automatically learn this mapping relationship and eventually obtain the marked model. Algorithm 1 shows the generation and embedding scheme of the watermark for the speaker recognition model in this paper. In the algorithm, Y_s represents the label corresponding to the original training set text; M_s represents the extracted Mel spectrogram; A represents the data matrix at the embedding position; $\mathcal{G}()$ function represents the process of embedding the Key into the Mel spectrogram; M_{WM} represents the Mel spectrogram containing the Key, and $Y_{trigger}$ represents the added trigger set label.

Algorithm 1: the generation and embedding scheme of the watermark

Input: training set of original audio $\mathrm{D_{train}} = \{(\mathrm{X_s}, \mathrm{Y_s})\}_{\mathrm{s}=1}^{\mathrm{n}}$, number of trigger audio samples m, key information $Key, Optimizer$,
Output: watermarked model $\mathcal{M}'$

```
    $for\ i \le\ m\ do:$
       $X_s \leftarrow Sample(D_{train}(Y_s))$;
       $M_s \leftarrow get\_Mel(X_s)$; /*extraction of Mel spectrogram from original audio*/
      $A \leftarrow M_s$;/*selection of suitable matrix A in Mel spectrogram*/
      $A' \leftarrow \mathcal{G}(Key, DCT(A))$; /*DCT transformation of matrix A row by row
and embedding of key information,*/
      $M_{WM} \leftarrow IDCT(A')$; /*inverse DCT transformation of watermarked matrix A*/
      $X_i \leftarrow griffin\_lim(M_{WM})$;
      $Y \leftarrow Y_{trigger}$;
       $D_{trigger}[i] = \{X_i, Y\}$; /*watermark generation*/
    $end\ for$
    $while\ Loss\ not\ converge\ do:$
      $model.train\left(D_{trigger}, D_{train}\right)$/*embedding of watermark*/
      $Optimizer.step()$
    $end\ while$
```

3.4. Watermark Verification

If the model is stolen, given that the model owner may suspect an infringement of their copyright by a remotely deployed model, it is necessary to confirm the ownership of the remote neural network model. In this process, the owner first generates a new trigger set and then sends a remote query to the suspicious model to obtain the prediction results.

The accuracy of the result prediction is evaluated on pre-defined labels. The process is represented by the following formula:

$$acc_{\mathcal{M}'} = V(\mathcal{M}', X_{trigger}, Y_{trigger}) \tag{4}$$

$$acc_{\mathcal{M}} = V(\mathcal{M}, X_{trigger}, Y_{trigger}) \tag{5}$$

Here, $X_{trigger}$ is the input trigger sample; $Y_{trigger}$ represents the pre-defined label, and V represents the watermark verification process. For the watermarked model $\mathcal{M}'$, if $acc_{\mathcal{M}'}$ is a value close to 1 or $acc_{\mathcal{M}'} \geq \varepsilon$, where ε is a threshold parameter close to 80%, the owner can verify the IP of the suspicious model and claim ownership of the remote model. In addition, for the non-watermarked model $\mathcal{M}$, since the model classification layer has not been changed, it can be queried normally. When the input trigger sample is given, $acc_{\mathcal{M}}$ should be a value close to 0 or $acc_{\mathcal{M}} \leq 1 - \varepsilon$.

4. Experiment

4.1. Model and Datasets

The experiment used SincNet [48]-based and standard CNN-based speaker recognition models, with the CNN model having the same architecture as SincNet but using standard convolution instead of SincNet-based convolution. These models were trained on two classic datasets, TIMIT [49] and Librispeech [50]:

(1) TIMIT: 3696 audio samples from 462 speakers were used as original audio samples, with 2310 samples used for training, 1386 samples used for testing, and the trigger set built on the basis of 150 randomly selected audio samples;

(2) Librispeech: 2484 audio samples from 2484 speakers were used as original audio samples, with each speaker randomly selecting 12–15 s of audio for training and 2–6 s for testing. The trigger set was built on the basis of 1500 randomly selected audio samples.

4.2. Fidelity and Efficiency

The objective of fidelity is to ensure that embedding the watermark does not affect the performance of the original model. Ideally, the performance of the non-watermarked model should be only slightly different from that of the watermarked model. Using the SincNet and CNN models as the original models, we trained them for 360 epochs in the TIMIT dataset and 2900 epochs in the LibriSpeech dataset. In order to compare with the reference literature, the parameters are set to be consistent with it. We use the RMSprop optimizer with a learning rate of 0.001 and a batch size of 128. The hyperparameter α is set to 0.95, and ε is set to 10^{-7}. When training SincNet on the LibriSpeech dataset, we evaluate the model every 50 rounds. When training SincNet on the TIMIT dataset, we evaluate the model every eight rounds. At the same time, when we train on two datasets, the regularization factor is 10,000, and the adjustment factor is 0.2. Next, the paper successfully reproduced the watermarking scheme proposed by Wang et al. and investigated the performance differences between our proposed watermark model, Wang et al.'s watermark model, and the original model on different datasets.

Tables 1 and 2, respectively, show the comparative results of the original SincNet model and watermarked model, and the CNN model and watermarked model in terms of frame-level error rate (FER) and sentence-level error rate (SER). It can be seen that our watermarking scheme does not significantly affect the performance of the original model, while its error rate is consistently lower than other schemes.

Table 1. Performance of the SincNet model and watermarking model.

Datasets	**TIMIT**			**LibriSpeech**		
model	SincNet	Wang et al.'s	ours	SincNet	Wang et al.'s	ours
FER(%)	0.4873	0.5078	0.4880	0.4956	0.5139	0.4969
SER(%)	0.0072	0.0192	0.0101	0.0158	0.0230	0.0173
running time	4 h 57 min	5 h 10 min	5 h 26 min	5 h 34 min	5 h 40 min	6 h 20 min

Table 2. Performance of the CNN model and watermarking model.

Datasets	**TIMIT**			**LibriSpeech**		
model	CNN	Wang et al.'s	ours	CNN	Wang et al.'s	ours
FER(%)	0.4853	0.5063	0.4900	0.4892	0.5077	0.4898
SER(%)	0.0093	0.0173	0.0123	0.0108	0.0166	0.0144
running time	17 h 44 min	18 h 39 min	18 h 56 min	18 h 40 min	19 h 08 min	19 h 58 min

4.3. Imperceptibility

The goal of imperceptibility is to ensure that the trigger samples created for watermark embedding are invisible in both visual and auditory aspects. Therefore, any changes made to the audio samples during watermark embedding should not be noticeable to the observers' visual and auditory systems. Ideally, the difference between the original audio signal and the audio signal with a watermark should be discernible by the speaker recognition model but imperceptible to the attackers. To evaluate the differences between the two, 50 samples were randomly selected. The average similarity between Mel spectrograms and waveform was calculated using Formulas (6) and (7), respectively.

$$SSIM(M, M_{WM}) = \frac{\left(2\mu_M \mu_{M_{WM}} + c_1\right)\left(2\sigma_{MM_{WM}} + c_2\right)}{\left(\mu_M^2 + \mu_{M_{WM}}^2 + c_1\right)\left(\sigma_M^2 + \sigma_{M_{WM}}^2 + c_2\right)} \tag{6}$$

Here, M represents the extracted original Mel spectrogram; M_{WM} represents the Mel spectrogram with embedded Key; μ represents their mean value; σ represents the variance between them, and c is a constant.

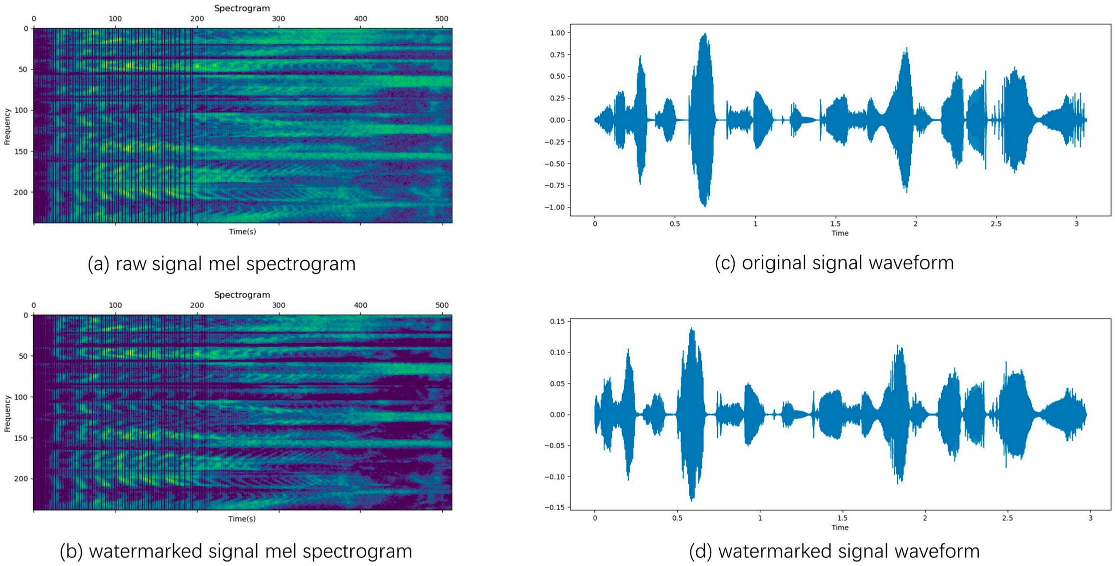
$$Similarity(X_s, X_i) = \frac{\vec{X_s} \cdot \vec{X_i}}{\left\|\vec{X_s}\right\|\left\|\vec{X_i}\right\|} \tag{7}$$

Here, X_s represents the original audio signal, and X_i represents the recovered audio signal.

Table 3 presents a quantitative comparison of the similarity between the two, and Figure 4 illustrates the degree of similarity between them. The results show that the feature distributions of the trigger samples and the original samples are similar, indicating good imperceptibility.

Table 3. Quantitative comparison of the similarity performance.

50 Samples	The SSIM Value	The Cosine Similarity
Sample1	0.9999	0.9803
Sample2	0.9999	0.9967
Sample3	0.9999	0.9875
Sample4	0.9999	0.9808
Sample5	0.9999	0.9974
Sample6	0.9999	0.9982
Sample47	0.9999	0.9866
Sample48	0.9999	0.9972
Sample49	0.9999	0.9977
Sample50	0.9999	0.9849



(a) raw signal mel spectrogram

(c) original signal waveform

(b) watermarked signal mel spectrogram

(d) watermarked signal waveform

Figure 4. Visualization of Mel spectrogram and waveform.

4.4. Effectiveness

The objective of effectiveness is to successfully verify the watermark model. To achieve this goal, two sets of trigger sets are used to query the model. One set is the trained trigger set D, which aims to verify whether the trigger set can be successfully identified by the watermark model. The other set is a newly generated trigger set D$'$, where a portion of the testing samples is selected to generate a new trigger set that has not participated in the training process. The purpose is to verify whether the watermark model remembers the copyright information of the model owner.

Table 4 shows the performance of the two sets of trigger sets on the watermark model. It can be seen that the trained trigger set can be identified by the model with 100% accuracy, and more than 95% of the newly generated trigger set can also be successfully identified by the model. This is due to the inherent generalization and memory capabilities of deep neural networks. The newly generated trigger samples can still be recognized and responded to with predefined labels. In other words, the proposed watermark scheme allows the model to generalize and remember the patterns of the owner's copyright information.

Table 4. The success rate of watermark verification.

Datasets	Train/Test	SincNet	CNN
TIMIT	D (train) D′ (test)	100% 98.67%	100% 96.67%
LibriSpeech	D (train) D′ (test)	100% 97.33%	100% 95.33%

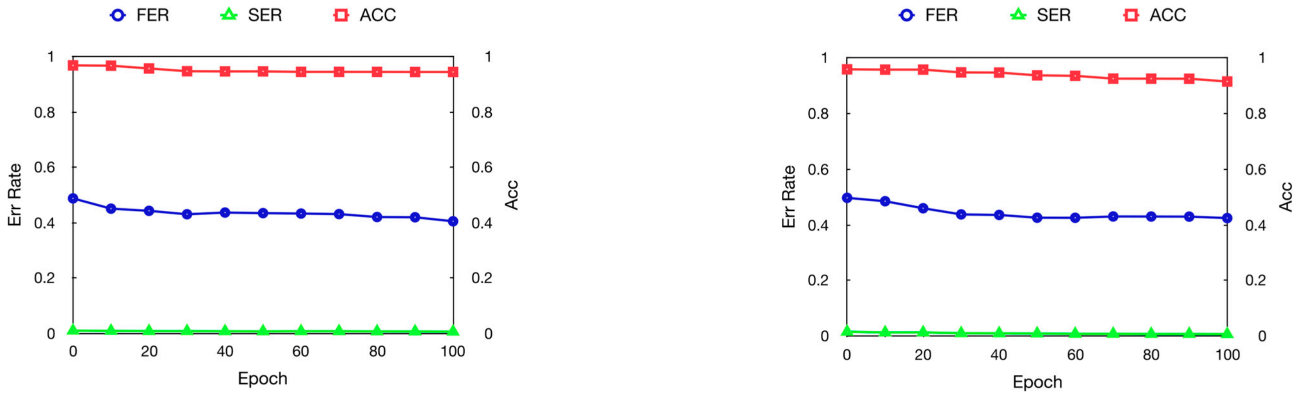
4.5. Robustness

The model watermarking scheme requires effective resistance to common attack methods, and the robustness objective is to successfully resist common watermark removal attacks, which involves model fine-tuning and model pruning.

4.5.1. Fine-Tuning

Usually, training a model from scratch takes a long time. Fine-tuning only requires updating the parameters of the later few layers based on the pre-trained model, which can save a lot of time compared to training from scratch and even improve performance. Usually, during the training process of the model, the parameters of the pre-trained model are also updated so that the model performance is better. For opponents without enough training sets, fine-tuning may be the best way to remove the owner's watermark. Therefore, this paper uses 30% of the testing samples as the fine-tuning dataset and uses the remaining as the testing dataset. The model was fine-tuned for 100 epochs.

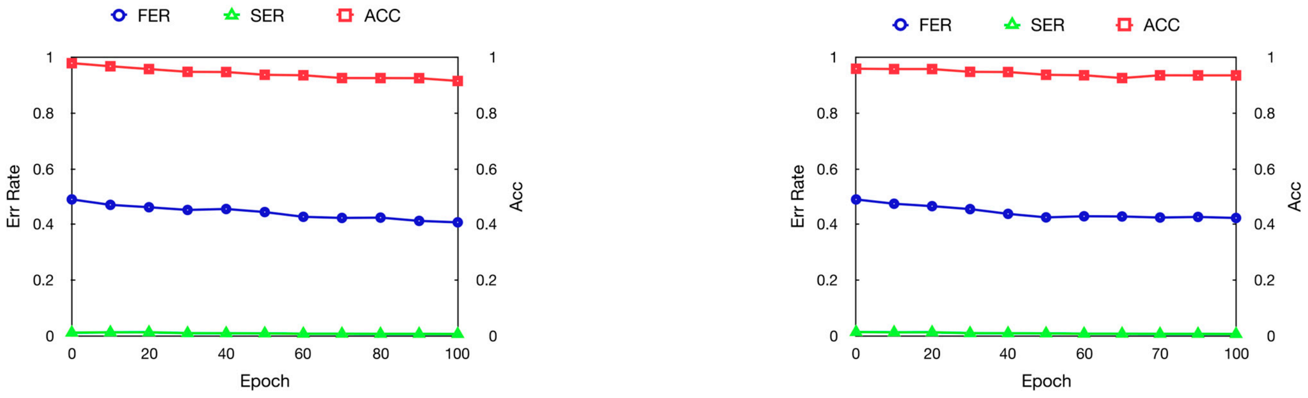
Figures 5 and 6 show the fine-tuning performance of the SincNet and CNN models on the TIMIT dataset and the LibriSpeech dataset, respectively. The left coordinate axis in the graph represents the error rate, and the coordinate axis on the right represents the watermark verification accuracy. As can be seen from the graph, the error rate of the model on the fine-tuning dataset continues to decrease, and at around 100 epochs, the error rate begins to stabilize. Meanwhile, the watermark detection accuracy remains above 90%, indicating that the proposed watermark scheme has strong robustness for model fine-tuning.



(a)TIMIT

(b)LibriSpeech

Figure 5. The success rate of watermark verification after fine-tuning at different epoch stages on the SincNet model.



(a)TIMIT

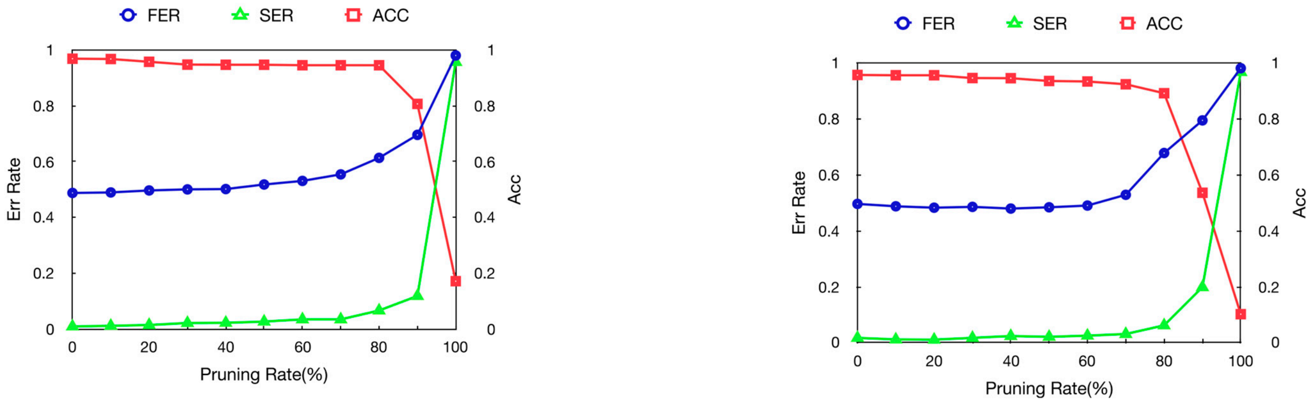
(b)LibriSpeech

Figure 6. The success rate of watermark verification after fine-tuning at different epoch stages on the CNN model.

4.5.2. Pruning

Pruning is generally used to explore the redundancy in the weights of the model. It aims to delete or trim the redundant and unimportant weights to reduce the size of the model and speed up the training and inference of the model while not significantly reducing the performance of the model. The attacker may want to remove the owner-embedded watermark in the model by pruning while maintaining the original performance of the model. In this paper, global random pruning is performed on the SincNet and CNN models.

Figures 7 and 8 show the pruning performance on the TIMIT dataset and the LibriSpeech dataset, respectively. It can be observed that the watermark detection rate can still maintain an accuracy of over 90% when pruning 70% of parameters. After the pruning rate exceeds 80%, although the watermark detection rate decreases, the model accuracy indicators (i.e., FER and SER) are also severely affected. At this point, the watermark model loses usability, and attackers cannot remove the watermark from the model through pruning without reducing the model's performance. Therefore, the proposed scheme can resist model pruning attacks.



(a)TIMIT

(b)LibriSpeech

Figure 7. The success rate of watermark verification after global pruning on the SincNet model.

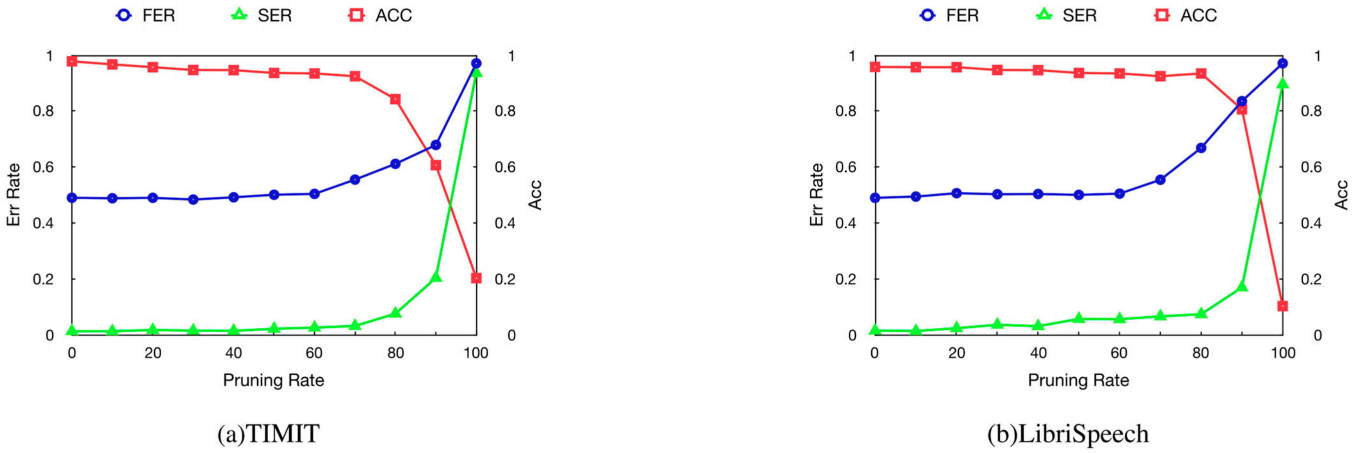


Figure 8. The success rate of watermark verification after global pruning on the CNN model.

4.6. Ambiguity Attacks

In this paper, security refers to the watermarking scheme proposed, which should be difficult for others to replicate or forge. This paper mainly considers resistance against ambiguity attacks. The goal of evading an ambiguity attack is to ensure that attackers cannot destroy the owner's watermark by embedding a fake watermark in the watermark model, thereby claiming ownership. This paper assumes that the attacker knows how to generate the trigger set and may embed their fake watermark in the stolen model to claim ownership, resulting in ambiguity when copyright issues arise. To address this issue, the paper proposes to use the same watermark generation method to embed other copyright information as a watermark in a specific frequency band of the Mel spectrogram and then fine-tune the embedded watermark model.

As shown in Figure 9, attackers can embed their fake watermarks but cannot cover up the original watermark information. Even after embedding the attacker's watermark, the owner's watermark verification success rate can still reach over 90%, which means that the owner can still declare their ownership. During copyright verification, the owner's model only contains the owner's watermark, while the suspicious model contains both the owner's watermark and the attacker's watermark. As a result, attackers cannot prove their ownership, and the proposed scheme can resist ambiguity attacks.

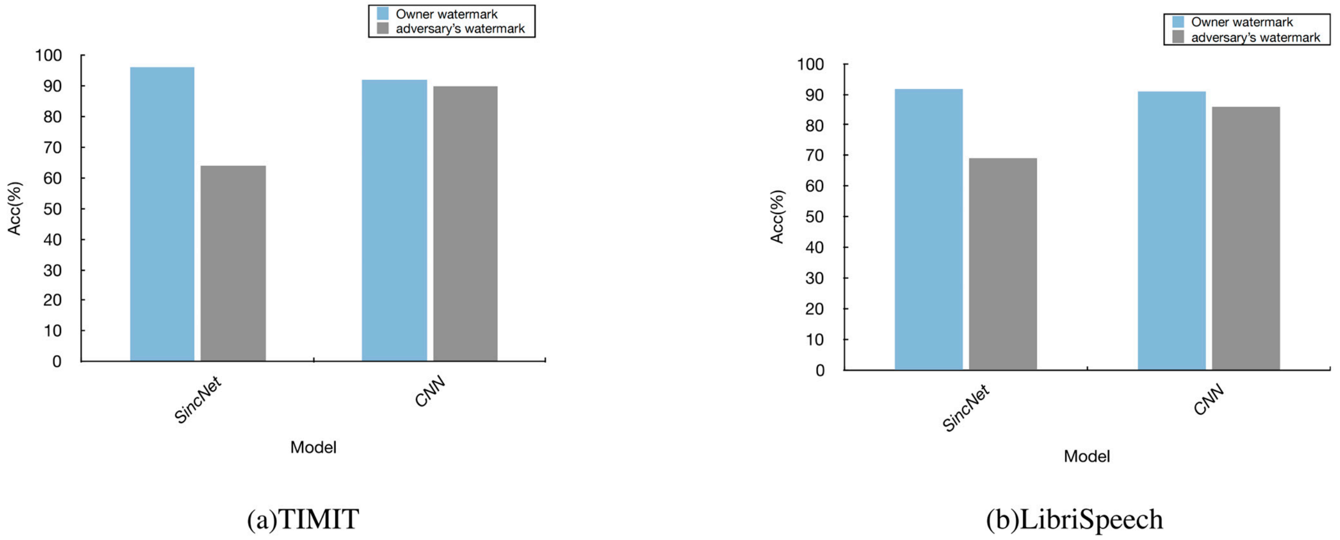


Figure 9. The success rate of watermark verification after embedding the adversary's watermark in the model.

5. Conclusions

The current voiceprint recognition models are vulnerable to attacks and face the risk of model theft. Therefore, our research aims to propose a black-box voiceprint recognition model protection framework to enhance the copyright protection performance of the models and restrict unauthorized access. Through these research results, we will be able to provide more effective protection solutions for the development of voiceprint recognition technology, thereby promoting the application and advancement of voiceprint recognition technology. However, this study also has some limitations. Firstly, the voiceprint recognition dataset used in this study may be small in scale and lack representativeness, which could limit the generalizability of the research findings. Future studies could consider using larger and more diverse datasets to validate the performance of the proposed protection framework. Secondly, the coverage of the attack models is limited. This study primarily focuses on the protection performance of voiceprint recognition models, but further research and validation are needed to assess the robustness of the framework against different types of attacks, such as adversarial attacks or model extraction attacks.

Author Contributions: Conceptualization, J.Z. and L.D.; methodology, J.Z. and L.D.; software, L.X.; validation, J.Z., L.D., and L.X.; formal analysis, J.Z.; investigation, J.Z.; resources, L.D.; data curation, L.X.; writing—original draft preparation, J.Z.; writing—review and editing, J.M. and X.Z.; visualization, J.Z. and L.X.; supervision, X.Z.; project administration, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 62362025) and the Hainan Province Key R&D plan project (No. ZDYF2022GXJS224).

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author, Xiaoyi Zhou, upon reasonable request.

Acknowledgments: No additional support was received for this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Amodei, D.; Ananthanarayanan, S.; Anubhai, R.; Bai, J.; Battenberg, E.; Case, C.; Casper, J.; Catanzaro, B.; Cheng, Q.; Chen, G.; et al. Deep speech 2: End-to-end speech recognition in English and mandarin. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 173–182.
2. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
3. Sennrich, R.; Haddow, B.; Birch, A. Neural machine translation of rare words with subword units. *arXiv* **2015**, arXiv:1508.07909. [CrossRef]
4. You, H.; Li, C.; Xu, P.; Fu, Y.; Wang, Y.; Chen, X.; Baraniuk, R.G.; Wang, Z.; Lin, Y. Drawing early-bird tickets: Towards more efficient training of deep networks. *arXiv* **2019**, arXiv:1909.11957. [CrossRef]
5. Kapse, A.; Belokar, S.; Gorde, Y.; Rane, R.; Yewtkar, S. Digital image security using digital watermarking. *Int. Res. J. Eng. Technol.* **2018**, *5*, 163–166.
6. Prajwalasimha, S.; Sowmyashree, A.; Suraksha, B.; Shashikumar, H.P. Logarithmic Transform based Digital Watermarking Scheme. In Proceedings of the International Conference on ISMAC in Computational Vision and Bio-Engineering, Palladam, India, 16–17 May 2018; Springer International Publishing: Cham, Switzerland, 2019; pp. 9–16.
7. Kumaraswamy, E.; Kumar, G.M.; Mahender, K.; Bukkapatnam, K.; Prasad, C.R. Digital Watermarking: State of The Art and Research Challenges in Health Care & Multimedia Applications. In Proceedings of the IOP Conference Series: Materials Science and Engineering, Warangal, India, 9–10 October 2020; IOP: Bristol, UK, 2020; p. 032031.
8. Fkirin, A.; Attiya, G.; El-Sayed, A.; Shouman, M.A. Copyright protection of deep neural network models using digital watermarking: A comparative study. *Multimed. Tools Appl.* **2022**, *81*, 15961–15975. [CrossRef]
9. Vybornova, Y. Method for copyright protection of deep neural networks using digital watermarking. In Proceedings of the Fourteenth International Conference on Machine Vision (ICMV 2021), Rome, Italy, 8–12 November 2021; Volume 12084, pp. 297–304.
10. Fan, X.; Gui, H.; Zhou, X. PCPT and ACPT: Copyright Protection and Traceability Scheme for DNN Model. *arXiv* **2022**, arXiv:2206.02541.
11. Zhong, H.; Chang, J.; Yang, Z.; Wu, T.; Arachchige, P.C.M.; Pathmabandu, C.; Xue, M. Copyright Protection and Accountability of Generative AI: Attack, Watermarking and Attribution. *arXiv* **2023**, arXiv:2303.09272.

12. Chen, X.; Wang, W.; Ding, Y.; Bender, C.; Jia, R.; Li, B.; Song, D. Leveraging unlabeled data for watermark removal of deep neural networks. In Proceedings of the ICML Workshop on Security and Privacy of Machine Learning, Long Beach, CA, USA, 16–20 June 2019.
13. Fan, X.; Zhou, X.; Zhu, B.; Dong, J.; Niu, J.; Wang, H. Survey of copyright protection schemes based on DNN model. *J. Comput. Res. Dev.* **2022**, *59*, 953–977.
14. Uchida, Y.; Nagai, Y.; Sakazawa, S.; Satoh, S. Embedding watermarks into deep neural networks. In Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, Bucharest, Romania, 6–9 June 2017; ACM: New York, NY, USA, 2017; pp. 269–277.
15. Fan, L.; Ng, W.K.; Chan, C.S. Rethinking deep neural network ownership verification: Embedding passports to defeat ambiguity attacks. In Proceedings of the 33rd Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 1–10.
16. Li, Z.; Hu, C.; Zhang, Y.; Guo, S. How to prove your model belongs to you: A blind-watermark based framework to protect intellectual property of DNN. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, Puerto Rico, 9–13 December 2019; pp. 126–137.
17. Hua, G.; Teoh, A.B.J. Deep fidelity in DNN watermarking: A study of backdoor watermarking for classification models. *Pattern Recognit.* **2023**, *144*, 109844. [CrossRef]
18. Li, F.Q.; Wang, S.L.; Zhu, Y. Fostering the robustness of white-box deep neural network watermarks by neuron alignment. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 3049–3053.
19. Yan, Y.; Pan, X.; Zhang, M.; Yang, M. Rethinking White-Box Watermarks on Deep Learning Models under Neural Structural Obfuscation. In Proceedings of the 32th USENIX Security Symposium (USENIX Security 23), Anaheim Marriott Hotel in Anaheim, CA, USA, 9–11 August 2023.
20. Kuribayashi, M.; Tanaka, T.; Suzuki, S.; Yasui, T.; Funabiki, N. White-box watermarking scheme for fully-connected layers in fine-tuning model. In Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security, Virtual Event, Belgium; 2021; pp. 165–170.
21. Kuribayashi, M.; Yasui, T.; Malik, A. White Box Watermarking for Convolution Layers in Fine-Tuning Model Using the Constant Weight Code. *J. Imaging* **2023**, *9*, 117. [CrossRef]
22. Lv, H.; Shen, S.; Lin, H.; Yuan, Y.; Duan, D. SVD Mark: A Novel Black-Box Watermarking for Protecting Intellectual Property of Deep Neural Network Model. In Proceedings of the 8th International Conference on Artificial Intelligence and Security, Qinghai, China, 15–20 July 2022; pp. 390–410.
23. Liu, Y.; Wu, H.; Zhang, X. Robust and imperceptible black-box DNN watermarking based on Fourier perturbation analysis and frequency sensitivity clustering. *arXiv* **2022**, arXiv:2208.03944.
24. Meng, Y.; Chen, X.; Sun, X.; Liu, Y.; Wei, G. A Dual Model Watermarking Framework for Copyright Protection in Image Processing Networks. *Cmc-Comput. Mater. Contin.* **2023**, *75*, 831–844. [CrossRef]
25. Chen, J.; Wang, J.; Peng, T.; Sun, Y.; Cheng, P.; Ji, S.; Ma, X.; Li, B.; Song, D. Copy, right? a testing framework for copyright protection of deep learning models. In Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 22–26 May 2022; pp. 824–841.
26. Vybornova, Y.; Ulyanov, D. Copyright protection for image classification models using pseudo-holographic watermarks. In Proceedings of the Fifteenth International Conference on Machine Vision (ICMV 2022), Rome, Italy, 18–20 November 2022; SPIE: Bellingham, WA, USA, 2023; Volume 12701, pp. 183–191.
27. Shen, W.; Rong, J.; Liu, Y.; Zhao, Y. IrisMarkNet: Iris feature watermarking embedding and extraction network for image copyright protection. *Appl. Intell.* **2023**, *53*, 9992–10008. [CrossRef]
28. Chen, H.; Rouhani, B.D.; Koushanfar, F. SpecMark: A Spectral Watermarking Framework for IP Protection of Speech Recognition Systems. *Interspeech* **2020**, 2312–2316.
29. Wang, Y.; Wu, H. Protecting the intellectual property of speaker recognition model by black-box watermarking in the frequency domain. *Symmetry* **2022**, *14*, 619. [CrossRef]
30. Zhang, J.; Chen, D.; Liao, J.; Zhang, W.; Hua, G.; Yu, N. Passport-aware Normalization for Deep Model Protection. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 22619–22628.
31. Yang, C.H.H.; Qi, J.; Chen, S.Y.C.; Chen, P.-Y.; Siniscalchi, S.M.; Ma, X.; Lee, C.H. Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 6523–6527.
32. Rajasekaran, A.S.; Azees, M.; Al-Turjman, F. A comprehensive survey on blockchain technology. *Sustain. Energy Technol. Assess.* **2022**, *52*, 102039. [CrossRef]
33. Patil, P.; Sangeetha, M.; Bhaskar, V. Blockchain for IoT Access Control, Security and Privacy: A Review. *Wirel. Pers. Commun.* **2021**, *117*, 1815–1834. [CrossRef]
34. Hewa, T.; Ylianttila, M.; Liyanage, M. Survey on blockchain based smart contracts: Applications, opportunities and challenges. *J. Netw. Comput. Appl.* **2021**, *177*, 102857. [CrossRef]
35. Habib, G.; Sharma, S.; Ibrahim, S.; Ahmad, I.; Qureshi, S.; Ishfaq, M. Blockchain Technology: Benefits, Challenges, Applications, and Integration of Blockchain Technology with Cloud Computing. *Future Internet* **2022**, *14*, 341. [CrossRef]

36. Kumar, P.; Sharma, S.K. An empirical evaluation of various digital signature scheme in wireless sensor network. *IETE Tech. Rev.* **2022**, *39*, 974–984. [CrossRef]
37. Li, D.; Han, D.; Crespi, N.; Minerva, R.; Li, K. A blockchain-based secure storage and access control scheme for supply chain finance. *J. Supercomput.* **2023**, *79*, 109–138. [CrossRef]
38. Yao, Y.; Li, H.; Zheng, H.; Zhao, Y.B. Latent backdoor attacks on deep neural networks. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, UK, 11–15 November 2019; pp. 2041–2055.
39. Gao, Y.; Doan, B.G.; Zhang, Z.; Ma, S.; Zhang, J.; Fu, A.; Nepal, S.; Kim, H. Backdoor attacks and countermeasures on deep learning: A comprehensive review. *arXiv* **2020**, arXiv:2007.10760.
40. Zhang, J.; Gu, Z.; Jang, J.; Wu, H.; Stoecklin, M.; Huang, H.; Molloy, I. Protecting intellectual property of deep neural networks with watermarking. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security, 2018*; ACM: New York, NY, USA, 2018; pp. 159–172.
41. Adi, Y.; Baum, C.; Cisse, M.; Pinkas, B.; Keshet, J. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In Proceedings of the 27th USENIX Security, Baltimore, MD, USA, 15–17 August 2018; USENIX Association: Berkeley, CA, USA, 2018; pp. 1615–1631.
42. Namba, R.; Sakuma, J. Robust watermarking of neural network with exponential weighting. In Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security; ACM: New York, NY, USA, 2019; pp. 228–240.
43. Zhang, Y.; Sun, G. A watermark algorithm based on space-domain and transform-domain. In Proceedings of the 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 12–14 July 2019; pp. 41–44.
44. Zhong, Q.; Zhang, L.Y.; Zhang, J.; Gao, L.; Xiang, Y. Protecting IP of Deep Neural Networks with Watermarking: A New Label Helps. *Adv. Knowl. Discov. DataMin.* **2020**, *12085*, 462–474.
45. Faheem, Z.B.; Ishaq, A.; Rustam, F.; de la Torre Díez, I.; Gavilanes, D.; Vergara, M.M.; Ashraf, I. Image Watermarking Using Least Significant Bit and Canny Edge Detection. *Sensors* **2023**, *23*, 1210. [CrossRef]
46. Mohammed, A.A.; Salih, D.A.; Saeed, A.M.; Kheder, M.Q. An imperceptible semi-blind image watermarking scheme in DWT-SVD domain using a zigzag embedding technique. *Multimed. Tools Appl.* **2020**, *79*, 32095–32118. [CrossRef]
47. Sharma, A.; Kumar, P.; Maddukuri, V.; Madamshetti, N.; Kishore, K.G.; Kavuru, S.S.S.; Raman, B. Fast Griffin Lim based waveform generation strategy for text-to-speech synthesis. *Multimed. Tools Appl.* **2020**, *79*, 30205–30233. [CrossRef]
48. Ravanelli, M.; Bengio, Y. Speaker recognition from raw waveform with sincnet. In Proceedings of the 2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece, 18–21 December 2018; pp. 1021–1028.
49. Garofolo, J.S.; Lamel, L.F.; Fisher, W.M.; Fiscus, J.G.; Pallett, D.S.; Dahlgren, N.L.; Zue, V. *TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1*; Linguistic Data Consortium: Philadelphia, PA, USA, 1993.
50. Panayotov, V.; Chen, G.; Povey, D.; Khudanpur, S. Librispeech: An asr corpus based on public domain audio books. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 19–24 April 2015; pp. 5206–5210.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.