# A Robust Audio Watermarking Algorithm Based on SVD-DWT

Huan Zhao[1],Fei Wang[1], Zuo Chen[1], Jun Liu[1]
[1]*School of Information Science and Engineering, Hunan University,*
*Lushan South Rood, Changsha 410082, P. R. China*
*hunanwangfei@126.com*

***Abstract*—In this paper, we proposed a novel blind audio watermarking algorithm, which combined Singular Value Decomposition(SVD) with Discrete Wavelet Transform(DWT). In our algorithm, We first partition the rearranged audio signal into blocks, then generate the vector by selecting the biggest singular values after performing SVD on these blocks. Finally we embed the watermark into the approximate components obtained from the DWT decomposition of the vector by means of quantization process. Experimental results showed that our algorithm has good robustness against the common audio signals processing operations. Compared with earlier schemes based on SVD, the proposed scheme has satisfying imperceptibility and improved payload.**

***Index Terms*—Audio watermarking, discrete wavelet transform (DWT), singular value decomposition (SVD), payload.**

## I. INTRODUCTION

With the rapid development of network technology and digital media audio technique, illegal users can easily obtain the audio resource and spread them. Therefore, that how to solve the issues about the copyright of audio has attracted a great deal of attention [1].

The problem above-mentioned can be efficiently solved by embedding the copyright information in a host audio, namely audio watermarking approach. As stipulated in the International Federation of the Phonographic Industry (IFPI) [2], determining whether an audio watermarking algorithm is effective can be based on whether the algorithm meet the following four basic requirements or not:

1) Robustness: Unless the audio suffers from serious damage, the embedded information can be accurately extracted from a watermarked audio even under the condition that the host audio undergoes common audio signal processing operations;

2) Imperceptibility: The distinction between the original audio and embedded audio can hardly be distinguished by the human ears. Besides, Signal to Noise Ration (SNR) which is used to appraise the quality of audio should be more then 20 dB;

3) Payload: we usually evaluate the payload of an audio watermarking scheme with bits per second, and the payload of an effective watermarking algorithm should be higher than 20 bps without affecting the imperceptibility of the audio;

4) Security: the security of a scheme should not relay on its algorithm. People without authorization can not extract embedded information from a covered audio.

In previous works [3], many audio watermarking techniques have been proposed, several good schemes have made contribution to significant progress. We can classify most of the audio watermarking algorithms into two categories according to the position where watermarking bit is embedded. One is time-domain algorithms, and the other one is transform-domain algorithms. On the one hand, the time-domain algorithms usually embed watermark information by directly modifying audio signal, such as Least Significant Bit (LSB) algorithm [4] and Echo algorithm [5]. Although these algorithms may have an outstanding performance at imperceptibility, embedded regions are still affected by many common signal and geometric processing. In other words, they can not meet the requirement of robustness. On the other hand, the transform-domain algorithms [6] usually use Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) or Discrete Fourier Transform (DFT) to transform the audio signal to locate appropriate embedded location. Similarly, pure transform-domain algorithms still show unsatisfactory robustness in signals. To cope with thus problem, Singular Value Decomposition (SVD) has been applied to robust watermarking as an effective transformation technique [7]. Recently, several audio watermarking techniques, which combined DWT or Short-Time Fourier Transform (STFT) with SVD have been proposed in [8]–[10]. For instance, Bhat *et al*. [11] proposed an adaptive audio watermarking algorithm; the watermarking information is embedded in the DWT domain based on SVD. The scheme is found to be imperceptible and robust to many attacks. However, its capacity is not very high. Bai Ying [12] presented an audio watermarking algorithm based on SVD-DCT. This method is different from the schemes, which use SVD to modify the transformed coefficients. The watermarking information is embedded by modifying the DCT coefficients of the blocks which is composed of SVs obtained by performing SVD transform on audio blocks. This method also has a good performance at imperceptibility and robustness. However, its capacity has the same drawback with [11].

Manuscript received March 29, 2013; accepted October 24, 2013
This work was sponsored by National Nature Science Foundation of China (61173106), Specialized Research Fund for the Doctoral Program of Higher Education, China (20100161120021) and the Young Teacher's Growth Program of Hunan University.

In this paper, different from traditional SVD-based schemes, which embed watermark bits information by modifying SVs directly. The proposed scheme obtains the biggest coefficient of SVs after SVD transformed audio sub blocks at first. Second it concatenates the selected coefficients to obtain a vector. Then the DWT is performed on the vector. Finally, we embed watermark bits by modifying the obtained approximate component. To improve the security of the scheme, a pseudo-random location sequence, which has been produced based on a secret key, is used to modify the approximate component.

The paper is structured as below: Section II introduces the related knowledge of SVD and DWT which used in our scheme; Section III describes the embedding algorithm in detail. Watermark extraction is addressed in Section IV. The experimental results are presented in Section V. The algorithm performance comparison is discussed in Section VI. Finally, we conclude the paper in Section VII.

## II. RELATED KNOWLEDGE

### A. SVD

The SVD of a matrix is a factorization of the matrix into three matrices, the SVD of a m × n matrix M can be described as bellow

$$M = USV^T = U\begin{vmatrix} \sigma_1 & .. & .. & 0 \\ 0 & \sigma_2 & .. & 0 \\ .. & .. & .. & .. \\ 0 & .. & .. & \sigma_r \end{vmatrix} V^T, \tag{1}$$

where $U \in R^{m\times m}$ and $V \in R^{n\times n}$ are both unitary matrixes. Matrix $S = diag(\sigma_1, \sigma_2 .... \sigma_r)$ has characteristic [13] as follows

$$\sigma_1 \geq \sigma_2 \geq .... \geq \sigma_r \geq \sigma_{r+1} ... = 0. \tag{2}$$

Given matrix $M \in R^{n\times n}$ and defined $\widehat{M} = M + \Delta M$ , where $\Delta M$ is a litter disturbance with the matrix M, their $i$th corresponding SVs are $s_i(M)$ and $s_i(\widehat{M})$ . The relation between them two can describe as follows

$$s_i(\widehat{M}) - s_i(M) \leq \|\Delta M\|_2, \tag{3}$$

where $i = 1, 2, ..., n$ , $\|\Delta M\|_2$ means 2-norm of matrix $\Delta M$ . The above mentioned SVD properties means slight modification of some components in matrix S rarely reduces the perception features of cover matrix. This property can be used to satisfy the robustness requirements of watermarking scheme.

### B. DWT

DWT is a useful tool for processing digital signal and has been widely used in computer science and engineering [14]. The DWT of signal $x(k)$ can be described as follows:

$$y_{low}[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n-k], \tag{4}$$

$$y_{high}[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n-k], \tag{5}$$

where $g$ and $h$ mean low-pass fitter and high-pass fitter respectively. An original signal will resolve into two parts: the approximate part and detail part. Further, the detail part can also be decomposed into two parts: high frequencies part and low frequencies part. A 3-lever DWT decomposition of a signal is shown in Fig. 1.

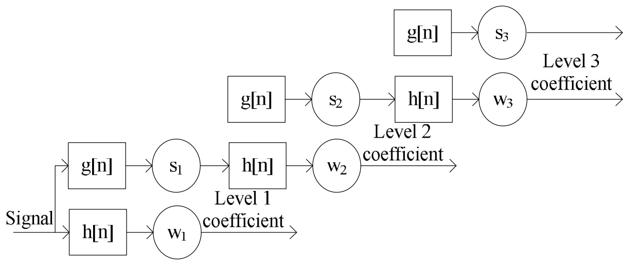



Fig. 1. A 3-level DWT of a signal.

In this paper, we choose the approximate part $w_3$ used for watermark embedding because [15] proved that it has the excellent spatio-frequency location properties.

## III. WATERMARKING EMBEDDING

### A. Synchronization Code

Embedding synchronization code in host audio can effectively resist the cropping and compressing attacks. The strategy has been widely used for watermarking techniques [16]. In order to reduce the cost of computation and time when searching the synchronization code, we choose the time-domain to embed the synchronization code. In our scheme, 16-bit barker code which has low autocorrelation properties $C_o = [1111100110101110]$ is treated as synchronization code. We divide the host audio into two parts $A_1^0$ and $A_2^0$, one is for synchronization code embedding, and the other is for watermark embedding. We can embed the synchronization code as bellow:

Step 1. First we segment the audio signal $A_1^0$ and suppose that each audio segment has n samples. The synchronization code is embedded into $Lsyn$ audio segments

$$SP(k) = A_1^0(i + k\times n), \tag{6}$$

where SP(k) means the $k$th audio segment, $1 \leq i \leq n$ , $1 \leq k \leq Lsyn$ ;

Step 2. Calculate the mean value of the audio segment

$$\overline{SP(k)} = \frac{1}{n}\sum_{i=1}^{n} A_1^0(i + k\times n), \tag{7}$$

where $1 \leq i \leq n$ , $1 \leq k \leq Lsyn$ ;

Step 3. Given synchronization code denoted as

$sw(k) \in \{0,1\}$, then each of them is embedded as follows

$$A_1^{0\prime}(i+k\times n) = A_1^{0}(i+k\times n) + (\overline{SP'(k)} - SP(k)), \quad (8)$$

where $\overline{SP'(k)}$ can be obtained from formula (9), and $Q(\overline{SP(k)}) = \mathrm{mod}(floor(SP(k)/¢_1),2)$, $floor(\cdot)$ means rounding to the minus integer. $mod(\cdot)$ denotes modulus after division, and $¢_1$ is the quantization step

$$\overline{SP'(k)} = \begin{cases} floor(\frac{\overline{SP(k)}}{¢_1})\times ¢_1 + \frac{¢_1}{2} & if \quad Q(\overline{SP(k)}) = sw(m), \\ floor(\frac{\overline{SP(k)}}{¢_1})\times ¢_1 - \frac{¢_1}{2} & else. \end{cases} \quad (9)$$

*B. Embedding Method*

The whole steps of the watermark embedding procedure are displayed in Fig. 2. The details of watermark bits embedding procedures are described as bellow.

Step 1. Partitioning. In the watermark embedding process, we first rearrange the audio signals $A_2^0$ into a matrix $B(m,m)$, and then the matrix is divided into non-overlapping $8\times 8$ blocks b;

Step 2. SVD transformation. For each block $b_i$ which can be denoted as $b_i = U_i \times S_i \times V_i^T$, we apply SVD on it;

Step 3. DWT decomposition. First we generate the vector $L = [s_1, s_2 .... s_n]$, $n = m/8$, where $s_i$ stands for the maximum value of matrix $S_i$, and then perform a 3-level DWT on L, finally we can obtain approximate part $L^3$ and detail parts $H^k$, where $1 \le k \le 3$;

Step 4. Watermark bits embedding. Given two embedded positions which are generated based on a secret key denote as $(p_1, p_2), 1 \le p_1, p_2 \le L_s$, where $L_s$ means the length of vector $L^3$, $¢_2$ is the quantization step. Supposing that the watermark bit w is going to be embedded in position $p_1$, we can describe the embedding process as (10)

$$L^{3'}(p_1) = \begin{cases} IQ \times ¢_2 + ¢_2/2 \; if \;\; w=1, \\ IQ \times ¢_2 - ¢_2/2 \; else, \end{cases} \quad (10)$$

where $IQ = floor(L^3(p_1)/¢_2)$ and $Q = \mathrm{mod}(IQ,2)$.

Step 5. Inverse DWT and SVD. We first perform inverse DWT on modified vector L then reconstruct $S'$ and calculate $b_i' = U_i \times S_i' \times V_i^T$. Finally, reduce the dimensions of the matrix so that the watermark can be embedded in the audio.

## IV. WATERMARK EXTRACTION ALGORITHM

*A. Search for the Synchronization Code*

Set the start index of the audio to *l*, the synchronization code searching process can be described as follows:

Step 1. Extract the synchronization code from the sample *l* to sample $l + Lsyn \times n$, where $n$ is the length of the synchronization code. For each synchronization code bit $V(k)$, the formulas are given as follows

$$SP''(k) = floor(\frac{1}{n}\sum A_1'^{0}(i + k\times n + l)), \quad (11)$$

where $1 \le i \le n, 1 \le k \le Lsyn$.

$$V(k) = \mathrm{mod}(SP''(k), 2), \quad (12)$$

where $1 \le k \le Lsyn$;

Step 2. Calculate the distance between the $C_0$ and the extracted synchronization code $V$

$$\mathrm{T} = \mathrm{V} \otimes \mathrm{C}_0, \quad (13)$$

where $\otimes$ means XOR operation.

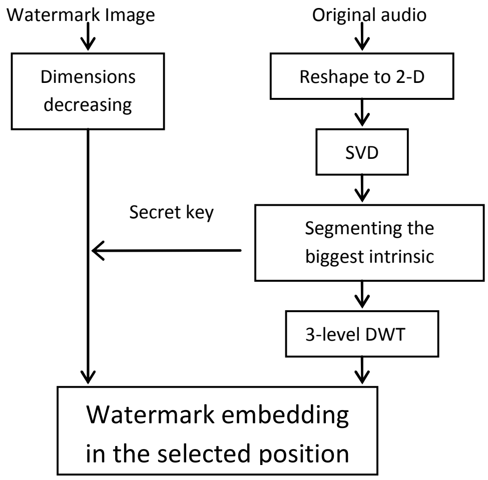



Fig. 2. Diagram of watermark embedding process.

Step 3. Check the value of T. If T is less than the given threshold, which means that V is the synchronization, move to water extraction. Otherwise, set $l = l + 1$, then repeat Step 1 and Step 2.

*B. Watermark Eextraction*

The watermark extraction without original audio signals procedures can be described as bellow:

Step 1. From the position where synchronization code search decided retrieve SVD-DWT block. Then obtain the vector $L' = [L^{3\prime}, H^{3\prime}, H^{2\prime}, H^{1\prime}]$;

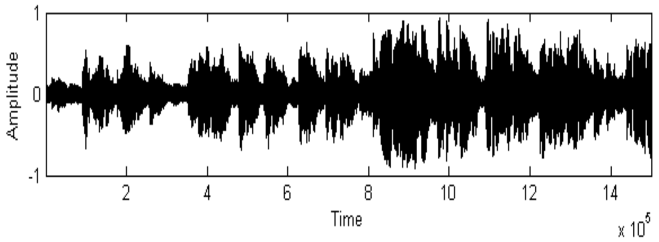
Step 2. Given extraction position $(p_1, p_2)$ $1 \le p_1, p_2 \le Ls$ based on the secret key. Where $Ls$ is the length of vector $L^{3\prime}$. Finally, the watermark is extracted as (15):

$$IQ' = floor(L^{3\prime}(p_1)/¢_2), \quad (14)$$

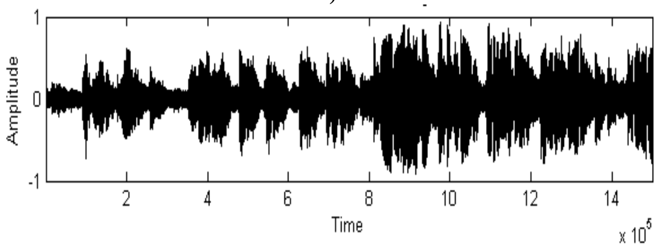
$$w' = \mathrm{mod}(IQ', 2). \quad (15)$$

## V. EXPERIMENTAL RESULTS

In this section, some experimental results are presented to illustrate the performance of our scheme. We choose five common music types to build the test database. The detailed description is listed in Table I. Each of the audio signals mentioned in this table is a mono wave file whose sampling rate is 44.1 kHz. A $32\times32$ binary image is used as watermark signal. The original watermark is shown in Fig. 4. In the experiment, we set 5 bit audio signal samples to embed one bit synchronization code. The haar wavelet basis has been used. The quantization parameter $¢_1$ is 0.035 and the quantization parameter $¢_2$ is 0.1. The threshold defined in Section IV A is set to be 3. The selection of all the parameter values aims at achieving a good trade-off between the requirements of imperceptibility, robustness and capacity.




a)




b)

Fig. 3. Audio signal a) riginal Pop audio and b) the watermarked host audio signal.

## A. Imperceptibility

In this paper, we use the following two common performance indicators to objectively evaluate the imperceptibility of the proposed algorithm.

SNR (Signal to Noise Ratio) is a statistical difference criterion that aimed to analyse the perceptual similarity between audio embedded watermark and original audio. SNR is calculated by

$$SNR(S,S^{'}) = 10\log_{10}\left(\frac{\sum_{i=1}^{L} S^2(i)}{\sum_{i=1}^{L}[S(i)-S'(i)]^2}\right). \tag{16}$$

SegSNR (Segmental SNR) is defined as the mean of evenly segmented signal-noise ratios. It is widely used as an estimator for signal quality, which is described as below

$$SegSNR(S,S^{'}) = \frac{10}{K}\sum_{m=0}^{K-1}\log_{10}\left(\frac{\sum_{i=1}^{r} S^2(i)}{\sum_{i=1}^{r}[S(i)-S'(i)]^2}\right), \tag{17}$$

where s'(i) and s(i) mean the watermarked and original signal respectively, K is the number of frames in the watermarked audio signal and r is the sample numbers of each frame. The SNR and SegSNR values of all tested audios are much above 20 dB, which satisfy the IFPI standard, are shown in Table II.

TABLE I. TEST DATABASE 16 BITS.

| Index | Audio description |
|---|---|
| 1 | Blues |
| 2 | Classic |
| 3 | Jazz |
| 4 | Pop |
| 5 | Country |

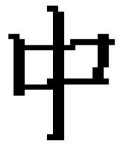


Fig. 4. Original watermarking.

The original and watermarked audio signals in time domain are presented in Fig. 3, the difference between them two is invisible.

## B. Robustness Test

According to IFPI, an effective audio watermarking algorithm should be robust to many common attacks. For the purpose of illustrating the robustness of the proposed watermarking algorithm, some attacks are performed by using MATLAB 2010 and CoolEdit2.0. We present the description of the attacks as follows:

1) Noise-attack: mix the watermarked signal with white Gaussian noise until the SNR of verarbeitet audio is 20 dB;

2) Resampling: the watermarked audio with a sampling rate of 44.1 kHz is down-sampled to 22.05 kHz, then up-sampled back to 44.1 kHz; in other situation, the watermarked audio with a sample of 44.1 kHz is up-sampled to 88.2 kHz, then down-sampled back to 44.1 kHz;

3) Requantization: quantize the 16-bit watermarked signal to 8 bits/sample and then back recovery the verarbeitet audio signal to 16 bits/sample;

4) Low-pass filtering: apply a low-pass filter on the watermarked audio with a cut-off frequency of 22.05 kHz;

5) Cropping: remove 100 samples of the watermarked audio at three random positions;

6) MP3 compression: the MPEG layer III compression and decompression at a bit rate of 196kbps, 96kbps and 64 kbps is applied respectively.

Our estimate of the similarity between the original watermark image and the extracted image rests on the two following common formulas:

NC (Normalized Cross-correlation) can be calculated from (18)

$$NC(I,I') = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} I(i,j)I'(i,j)}{\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{N} I^2(i,j)}\sqrt{\sum_{i=1}^{N}\sum_{j=1}^{N} I'^2(i,j)}}. \tag{18}$$

BER (the Bit Error Rate) can be obtained from (19)

$$BER(I,I') = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} I(i,j)\otimes I'(i,j)}{N\times N}, \tag{19}$$

where $I'$ and $I$ are the extracted watermarks and the original watermarks respectively. M means the number of embedded image bits, j and i are indexes of the image, and $\otimes$ means XOR operation.

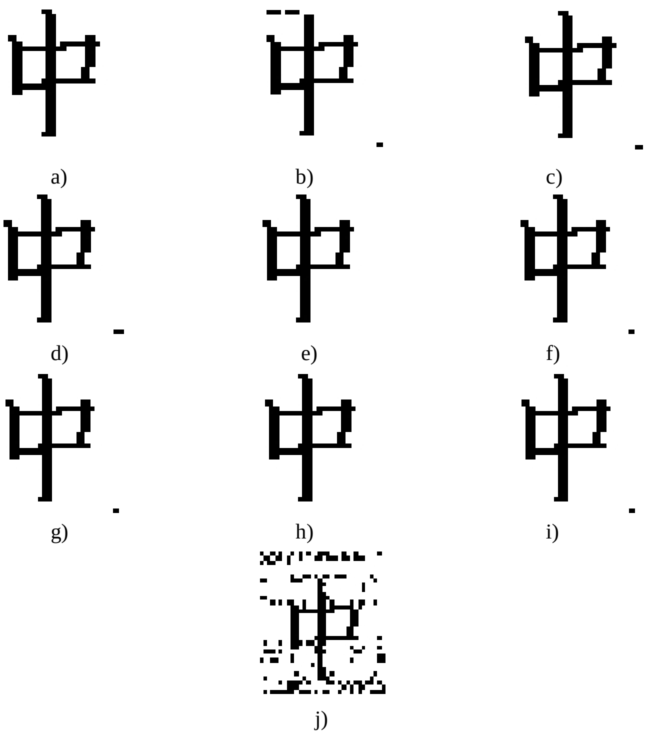



Fig. 5. Extracted watermark from attacks of Pop music. a) No attack, b) down-resample, c) up-resample, d) requantization, e) noise-attack, f) low-pass filtering, g) cropping, h) Mp3 192 kbps, i) Mp3 96kbps, j) Mp3 64 kbps.

From Table II and Fig. 5, we can observe that all extracted watermarks have high NC values and low BER values against common audio signal attacks. These results indicate that our algorithm satisfies the requirement of the robustness. For compression attacks, even the compressive strength is 64 kbps, which means the recovery audio will lose many bits, the recovered watermark image is still identifiable.

TABLE II. NC AND BER EXTRACTED WATERMARK FORM EACH AUDIO FILE.

| Audio file | Attack | NC | BER (100 %) |
|---|---|---|---|
| Blues | No attack | 1 | 0 |
| | Down-resample | 0.9995 | 0 |
| | Up-resample | 0.9995 | 0 |
| | Requantization | 1 | 0 |
| | Noise-attack | 0.9995 | 0 |
| | Low-pass filtering | 0.9995 | 0 |
| | Cropping | 1 | 0 |
| | Mp3 192 kbps | 0.9995 | 0 |
| | Mp3 96 kbps | 0.9978 | 0.39 |
| | Mp3 64 kbps | 0.8902 | 19.04 |
| Country | No attack | 1 | 0 |
| | Down-resample | 0.9995 | 0 |
| | Up-resample | 0.9995 | 0 |
| | Requantization | 1 | 0 |
| | Noise-attack | 0.9995 | 0 |
| | Low-pass filtering | 0.9995 | 0 |
| | Cropping | 0.9989 | 0.2 |
| | Mp3 192 kbps | 0.9995 | 0 |
| | Mp3 96 kbps | 0.9989 | 0.2 |
| | Mp3 64 kbps | 0.9906 | 1.76 |
| Classic | No attack | 1 | 0 |
| | Down-resample | 1 | 0 |
| | Up-resample | 1 | 0 |
| | Requantization | 0.9995 | 0 |
| | Noise-attack | 0.9995 | 0 |
| | Low-pass filtering | 1 | 0 |
| | Cropping | 0.9995 | 0 |
| | Mp3 192 kbps | 0.9989 | 0.2 |
| | Mp3 96 kbps | 0.9989 | 0.2 |
| | Mp3 64 kbps | 0.9016 | 17.29 |
| Jazz | No attack | 1 | 0 |
| | Down-resample | 0.9995 | 0 |
| | Up-resample | 1 | 0 |
| | Requantization | 0.9995 | 0 |
| | Noise-attack | 0.9995 | 0 |
| | Low-pass filtering | 1 | 0 |
| | Cropping | 1 | 0 |
| | Mp3 192 kbps | 0.9989 | 0.2 |
| | Mp3 96 kbps | 0.9995 | 0 |
| | Mp3 64 kbps | 0.9956 | 0.78 |
| Pop | No attack | 1 | 0 |
| | Down-resample | 0.9950 | 0.88 |
| | Up-resample | 0.9995 | 0 |
| | Requantization | 0.9995 | 0 |
| | Noise-attack | 1 | 0 |
| | Low-pass filtering | 0.9989 | 0.2 |
| | Cropping | 0.9995 | 0 |
| | Mp3 192 kbps | 0.9995 | 0 |
| | Mp3 96 kbps | 0.9989 | 0.2 |
| | Mp3 64 kbps | 0.9152 | 14.45 |

### A. *Data payload*

We define the data payload of a watermarking algorithm as the amount of information which is embedded in a host audio signal. By convention, we measure the data payload by bps and describe it as

$$P_c = F_s / N_L, \tag{20}$$

where $F_S$ is the sampling rate of the host audio. Meanwhile we defined the amount of samples for one bit information as $N_L$. Then we can figure out the data payload of the scheme is 86 bit/s.

### B. *Security*

In the proposed watermarking scheme, the selection of the embedding position is based on secret key and the quantization parameters, which greatly influence the effect of extracted watermark is unknown to illegal users. Therefore, the algorithm meets the requirement of security.

## VI. PERFORMANCE CCOMPARATION AND DISCUSSION

In this section, we compared the proposed algorithm with related audio watermarking algorithms. We selected two lately related literatures, which also embedded watermark based on SVD. The detailed comparisons between imperceptibility and payload are shown in Table IV. The robustness of the scheme has been displayed in Table II. All algorithms listed in Table IV have a good performance of the imperceptibility. Their SNRs are much beyond 20 dB. As shown in Table III, our algorithm shows better imperceptibility than the algorithm in [11]. The data payload of the proposed scheme is much higher than other algorithms listed in the Table IV. These data indicate that our scheme can embed much more information with the same audio file. Meanwhile the DCT is computational intensive and needs more execution time. In conclusion, our scheme has achieved a better balance among the requirements of robustness, payload and imperceptibility.

TABLE III. SNR AND SEGSNR BETWEEN ORGINAL AUDIO AND WATERMARKED AUDIO.

| Index | SNR (dB) | SegSNR(dB) |
|---|---|---|
| 1 | 29.5 | 32.0 |
| 2 | 23.2 | 26.1 |
| 3 | 20.3 | 23.8 |
| 4 | 27.7 | 30.04 |
| 5 | 24.3 | 24.9 |
| **Average** | **25** | **27.4** |

TABLE IV. COMPARISON RESULTS OF CAPACITY AND IMPERCEPTIBILITY.

| Algorithms | Method | Payload (bps) | SNR (dB) | Synchronization |
|---|---|---|---|---|
| [11] | Adaptive DWT-SVD | 45.9 | 24.4 | Yes |
| [12] | SVD-DCT | 43 | 32.5 | Yes |
| **Ours** | **SVD-DWT** | **86** | **25.0** | **Yes** |

## VII. CONCLUSIONS

A novel blind audio watermarking scheme that combined features of DWT and SVD was proposed in this paper. It Performs DWT on maximum singular values that obtained from SVD of host audio rather than on cover audio directly that fully exploited the outstanding characteristics of SVD. Experimental results have indicated that our algorithm has excellent imperceptibility and is robust to common audio attacks including add-noise, low-pass filtering, re-sampling, requantization, cropping and Mp3 compression. The comparison of proposed scheme with other SVD based algorithms in [10], [11] indicates that the proposed scheme is with higher payload and satisfying imperceptibility. The simulation results verify that our scheme fulfils the IFPI performance requirement and is suitable for application in copyright protection.

## ACKNOWLEDGMENT

The deepest gratitude goes to all people who helped me during the creation of this paper. I would like to give my appreciation to my teammates of the laboratory for many usefull writing suggestions.

## REFERENCES

[1] M. Swanson, B. Zhu, A. Tewfik, "Current state of the art, challenges and future directions for audio watermarking", in *Proc. IEEE ICMCS*, 1999, vol. 1, pp.19–24.

[2] S. Katzenbeisser, F. Petitcolas, "Information hiding techniques for steganography and digital watermarking", Artech House, USA, 2000.

[3] M. D. Swanson, B. Zhu, A. H. Tewfik, L. Boney, "Robust audio watermarking using perceptual masking", *Signal Processing*, vol. 66, no. 3, pp. 337–355, May 1998. [Online]. Available: http://dx.doi.org/10.1016/S0165-1684(98)00014-0

[4] N. Cvejic, T. Seppanen, "Increasing robustness of LSB audio steganography using a novel embedding method", in *Proc. 5th IEEE Int. Conf. Info. Tech.: Coding and Computing*, Apr. 2004, vol. 2, pp. 533–537.

[5] P. Bassia, L. Pitas, "Robust audio watermarking in the time domain", *IEEE Trans. Multimedia*, vol. 3, no. 2, pp. 232–241, Jun. 2001. [Online]. Available: http://dx.doi.org/10.1109/6046.923822

[6] X. Wang, H. Zhao, "A novel synchronization invariant audio watermarking scheme based on DWT and DCT", *IEEE Trans. Signal Process*, vol. 54, pp.4835–4840, 2006. [Online]. Available: http://dx.doi.org/10.1109/TSP.2006.881258

[7] C. C. Chang, P. Tsai, C. C. Lin, "SVD-based digital image watermarking scheme", *Pattern*, vol. 26, pp. 1577–1586, Jul. 2005.

[8] R. Liu, T. Tan, "An SVD-based watermarking scheme for protecting rightful ownership", *IEEE Trans. Multimedia*, vol. 4, no. 1, pp. 121–128, Mar. 2002. [Online]. Available: http://dx.doi.org/ 10.1109/6046.985560

[9] H. Ozer, B. Sankur, N. Memon, "An SVD-based audio watermarking technique", in *Proc. Seventh Workshop on Multimedia and Security*, New York, Aug. 2005, pp. 51–56. [Online]. Available: http://dx.doi.org/10.1145/1073170.1073180

[10] Abd El-Samie, "An efficient singular value decomposition algorithm for digital audio watermarking", *Int. Journal of Speech Technology*, vol. 12, no. 1, pp. 27–45, 2009. [Online]. Available: http://dx.doi.org/10.1007/s10772-009-9056-2

[11] B. Vivekananda, S. Indranil, D. Abhijit, "An adaptive audio watermarking based on the singular value decomposition in the wavelet domain", *Digital Signal Process*, vol. 20, no. 6, pp. 1547–1558, Feb. 2010. [Online]. Available: http://dx.doi.org/ 10.1016/j.dsp.2010.02.006

[12] B. Y. Lei, I. Y. Soon, Z. Li, "Blind and robust audio watermarking scheme based on SVD–DCT", *Signal Processing*, vol. 91, pp. 1973–1984, Mar. 2011. [Online]. Available: http://dx.doi.org/ 10.1016/j.sigpro.2011.03.001

[13] R. Z. Liu, T. N. Tan, "SVD based digital watermarking method", *Acta Electronica Sinica*, vol. 29, no. 2, pp. 168–171, Feb. 2001.

[14] A. N. Akansu, P. Duhamel, X. Lin, M. de Courville, "Orthogonal transmultiplexers in communications: A review", IEEE *Trans. Signal Processing*, vol. 46, pp. 979–995, Apr. 1998. [Online]. Available: http://dx.doi.org/10.1109/78.668551

[15] H. Hao, L. Chen, Y. P. Zhang, Y. G. Zhang, B. Zhang, "A high performance audio watermarking algorithm based on substitution in wavelet domain", *Advances in Electronic Engineering, Communication and Management*, vol. 2, pp. 601–607, 2012. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-27296-7_92

[16] S. Wu, J. Huang, D. Huang, Y. Shi, "Efficiently self-synchronized audio watermarking for assured audio data transmission", *IEEE Trans. Broadcast*, vol. 51, pp. 69–76, 2005. [Online]. Available: http://dx.doi.org/10.1109/TBC.2004.838265