

# A Novel Audio Watermarking Algorithm Based On Reduced Singular Value Decomposition

Jian Wang<sup>1</sup>, Ron Healy<sup>2</sup>, Joe Timoney<sup>3</sup>

Computer Science Department

NUI Maynooth,

Co. Kildare, Ireland

jwang@cs.nuim.ie<sup>1</sup>, rhealy@cs.nuim.ie<sup>2</sup>, jtimoney@cs.nuim.ie<sup>3</sup>

**Abstract** — This paper presents a transparent and accurate semi-blind audio watermarking algorithm, based on the Reduced Singular Value Decomposition (RSVD). The proposed scheme manipulates coefficients of one of the resulting unitary matrices for embedding watermark bits. To preserve audio fidelity a threshold-based distortion control technique is given. Test results on real music signals show that this watermarking scheme is transparent, accurate and robust against MP3 compression.

*Keywords*- RSVD, MP3 Compression, Audio Watermark

## I. INTRODUCTION

Audio watermarking is a technique whereby data is hidden in a cover or host audio signal as a form of steganography. Depending on whether it requires access to the original signal and/or the actual watermark, the scheme may be described as ‘informed’ or ‘blind’. ‘Semi-blind’ schemes do not need the original signal or the actual watermark at the decode stage but instead require some other information related to the process [1].

MPEG-1 Layer III (MP3) is the *de facto* standard for transmission and storage of compressed audio for both World Wide Web (WWW) and portable media applications. Many audio watermark schemes have been designed for non-compressed audio data but unfortunately in many cases the watermark cannot survive MP3 compression. This limitation obviously limits the usefulness of any audio watermarking scheme. The difficulty when designing a watermark that can withstand MP3 compression is related to the fact that the MP3 algorithm non-linearly modifies the spectrum based on a psychoacoustic model and the expected bit rate. Thus, any watermark embedded among the audio’s spectral components risk being lost after compression.

The Singular Value Decomposition (SVD) has been used as an effective technique in digital image watermarking [4], as some coefficients in the decomposed matrices are consistent after image compression, such as JPEG. This robustness is impressive and suggests that an SVD-based approach for digital audio could also be effective. Taking inspiration from [4][5], this paper describes a novel algorithm for the application of the SVD to the watermarking of audio which will undergo compression. Given that image and audio data are very different, it will also seek to resolve any specific perceptual issues that might arise.

## II. BACKGROUND

The SVD is a well-known numerical analysis tool used on matrices. The Reduced SVD (RSVD) is a more compact representation, and compared with the standard SVD, this version can be more efficient and unique in its decomposition outcome [3]. If  $A$  is an arbitrary matrix, with the Reduced SVD this matrix can be decomposed into three matrices as  $U, S, V$ , where  $U$  and  $V$  are orthogonal matrices with  $p \times r$  and  $q \times r$  ( $r$  is the rank of matrix  $A$ ), and  $S$  is a  $r \times r$  diagonal matrix with positive elements and no non-zero elements [3]. The inverse of matrix  $A$  can then be constructed as

$$A^{-1} = USV^T = \begin{bmatrix} u_{11} & \dots & u_{1r} \\ \vdots & \ddots & \vdots \\ u_{p1} & \dots & u_{pr} \end{bmatrix} \begin{bmatrix} s_{11}^{-1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & s_{rr}^{-1} \end{bmatrix} \begin{bmatrix} v_{11}^{-1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & v_{q1}^{-1} \end{bmatrix} \quad (1)$$

The SVD has been already been applied to digital image watermarking [4], [5], [6]. One particular digital image watermarking algorithm based on SVD has been proposed in [4]. It is formulated based on the two observations. Firstly, column-wise modification of the elements of the  $U$  matrix will cause less visible distortion than modifying the elements row-wise. Secondly, row-wise modification of the elements of  $V^T$  will cause less visible distortion than modifying the elements column-wise [4]. A further observation in [5] was found also to be useful in that the elements in the first column of  $U$  and  $V$  can be modified without significantly impacting signal integrity, allowing them to survive common attacks including compression. Other ideas for SVD-based watermarking have been also presented in [8] [10]. However, they used an ‘informed-decode’ process, requiring access to the original signal or the watermark, which limits their practical application.

Compared to image watermarking, audio watermarking is a more difficult task because the human auditory system is more sensitive to difference than the human vision system [9]. Therefore, alterations have to be made to the SVD watermarking approach [4] before it can be applied to audio. The next section details how this is achieved.

## III. ALGORITHM

Before applying the SVD to an audio signal, the issue of how to organize the data into matrix form has to be

addressed. The audio signal should first be split into frames whose length is denoted as  $len$ . The magnitude spectrum of the signal in each frame can then be computed. This spectrum will contain  $len/2$  frequency bins below the Nyquist frequency. If we put all of the frequency components of several consecutive frames into the same matrix, there should be redundancy, as, generally, each component's magnitude would be similar to its value in the previous and the next frame. An alternative approach to organizing the RSVD input matrix is to put all the spectral components from one frame only into a single matrix. For example, with a frame length of 1024 we just use the first 512 magnitude values and transform them into a  $64 \times 8$  matrix. The magnitudes of the first 64 frequency bins will be put into the first column, the second 64 bin magnitudes into the second column and so on. The values in the first column will normally but not always be more significant than those in subsequent columns.

As mentioned in Section II, the watermarking algorithms proposed in [4][5] cannot be applied directly to the RSVD of this input matrix because, as seen in (1)  $s_{1,1}$  is the largest value in  $S$  and, when expanding the product it is only multiplied with the column elements  $u_{1,p,1}$ . This results in the most significant peaks within the reconstructed spectrum being determined by  $u_{1,p,1} * s_{1,1} * v_{1,1,q}$ . Even a minor alteration of any element in  $u_{1,p,1}$  may therefore result in significant distortion when the time signal is resynthesized.

To overcome this obstacle the following process was devised. Similarly to the point made about  $s_{1,1}$  in (1),  $s_{2,2}$  is the second largest value in the  $S$  matrix and is multiplied with  $u_{1,p,2}$  which is more significant than the coefficients in all other columns except  $u_{1,p,1}$ . Empirically, we found that the peaks in  $u_{1,p,2}$  are almost always retained after compression. Modifying the elements of  $u_{1,p,2}$  rather than those in  $u_{1,p,1}$  will therefore result in less distortion in the reconstructed signal as  $s_{2,2}$  is generally much smaller than  $s_{1,1}$ .

Thus, we choose to create a local peak in  $u_{1,p,2}$  to represent the watermark data. Assume we have a binary information stream representing the watermark. If the bit to be embedded is a '1', we increase the magnitude of the element  $u_{c,2}$  where  $c$  is a user-defined key value such that it is a peak value. The surrounding elements at  $u_{c-1,2}$  and  $u_{c+1,2}$  are also reduced to a negligible low value  $\phi$ . Similarly, if the bit to be embedded is a '0', we increase the magnitude of the element  $u_{c+1,2}$  such that it is a peak value. The surrounding elements at  $u_{c,2}$  and  $u_{c+2,2}$  are also reduced to the same negligible low value  $\phi$ .

However, the extent of the peak value introduced is an issue that requires careful evaluation. If it is big, it will be robust under compression but may result in perceptual distortion in the reconstructed audio. Likewise, if it is small, it may not survive compression but will not be perceived.

Generally, a trade-off exists between the accuracy of watermark detection, its robustness and its perceptual transparency [2]. Our solution to minimize distortion without significantly affecting detection precision is now explained. Consider the embedding of a '1' bit: according to (2), all values in row  $c$  of the reconstructed signal spectrum matrix  $A'_{c,1..q}$  are affected by the manipulation of  $u_{c,2}$  and thus all of

them will be distorted to a greater or lesser extent. An expression for the row  $c$  of  $A'$  is

$$A'_{c,1..q} = \sum_{p=1}^P \sum_{k=1}^Q u'_{c,p} * s_{p,2} * v_{k,2} \quad (2)$$

where  $u'_{c,p}$  denotes the  $c$  row of the  $u$  matrix after modification of  $u_{c,2}$ .

We can see from (2) that the  $k^{\text{th}}$  element at  $A'_{c,k}$  is mostly determined by  $u'_{c,2} * s_{2,2} * v_{2,k}$ , thus  $A'_{c,k}$  should be strongly correlated with  $u'_{c,2}$ . However, we can eliminate the distortion of each component of  $A'_{c,1..q}$ , except  $A'_{c,k}$ , by returning their magnitudes back to their original values given in  $A_{c,1..q}$ . Doing this, the level of distortion in the reconstructed audio signal can be reduced to a large extent with minimal impact on the values of  $u'_{c,2}$ . This means that we can control the distortion but retain the watermark.

Now we can address the issue of which value should be assigned to  $u'_{c,2}$  if embedding a '1' bit. The modification of any element of  $u_{1,p,2}$  must be tailored so that after the reconstructed audio is compressed this modification will persist. Note that when any signal undergoes the compression process all large magnitude spectral components are not guaranteed to survive it. This is because such a component could be masked by other larger magnitude components that exist in the same critical band. Only those components whose magnitude is strongest in their own critical band can survive compression, even if they are comparatively low in the entire spectrum. Thus, it is best to set a value for the peak of  $u'_{c,2}$  that will create a strong local maximum, within a particular range, in the spectrum of the reconstructed signal.

Denoting  $locMag$  as the local maximum value among  $A_{c,d..c+d,k}$ , where  $d$  denotes a predefined range of items on either side of the element  $A_{c,k}$ , an expression for the new local peak value of  $u'_{c,2}$  is given

$$u'_{c,2} = \frac{locMag}{s_{2,2} * v_{k,2}} \quad (3)$$

As a consequence of applying (3),  $u'_{c-1,2}$  and  $u'_{c+1,2}$  will be set equal to  $\phi$ . Audible distortion appears in the reconstructed audio if  $u'_{c,2}$  is much bigger than  $u'_{c-1,2}$  and  $u'_{c+1,2}$ . Following experimentation, we found that by guaranteeing that  $u'_{c,2} - u'_{c+1,2} \leq 0.08$ , it is sufficient for the local peak  $u'_{c,2}$  to survive compression without introducing noticeable distortion.

By way of an illustration of the output of the watermarking process described, an example log spectrum in dB of a frame of the original signal and its watermarked counterpart are given in Figure 1, where the x-axis denotes frequency bins and the y-axis denotes magnitude in dB. From figure 1, we can see that the watermarked signal is approximately the same as the original signal in their log

spectrum domain, except for one small peak around bin 125 in this case. Thus, the distortion introduced is minimal.

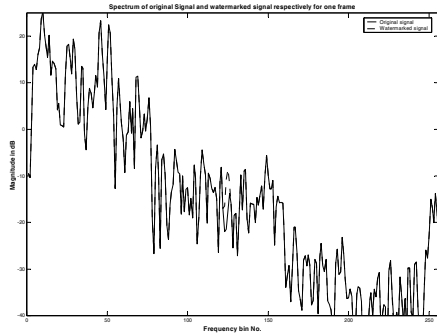


Figure 1 Log Spectrum of original signal and its watermarked counterpart for one frame

A block diagram of our complete watermark embedding scheme is given in Figure 2. The FFT is found of each frame of the original signal and then RSVD is applied to each spectrum after its reorganization into matrix form. The elements in the vicinity of row  $c$  in  $u_{1..p,2}$  are modified to satisfy the embedding conditions detailed above. Note that the procedure below only shows the case of embedding the bit '1'.

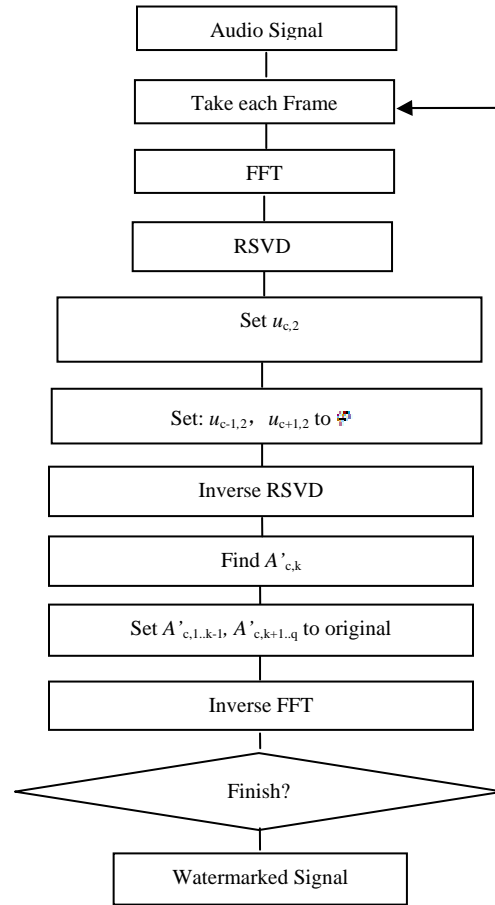


Figure 2 Watermark embedding flowchart

Once the modification has taken place, an inverse RSVD is performed to find the matrix  $A'$ . The elements from  $A'_{c,1..k-1}$ , and  $A'_{c,k+1..q}$  are restored to their original values from  $A'$ . The inverse FFT is found to generate the time domain watermarked audio frame. This process is repeated until all the frames have been watermarked.

The watermark extraction procedure can be briefly described in the following steps:

1. Get watermarked signal and split into frames.
2. Apply the FFT to each frame and then organize its spectrum magnitudes as a matrix  $A$ .
3. Apply the RSVD to  $A$ , check if  $u_{c,2} - u_{c+1,2} > 0$ . If this is true, the bit detected is a '1', else it is a '0'.

#### IV. EXPERIMENTAL RESULTS

We randomly selected 30 candidate music files of different genres from a collection of 350 and performed the full encode-embed-decode cycle on each file. We embedded 2048 bits into 2048 frames. In each case, the accuracy of the recovery of the watermark bit sequence was calculated as

$$\text{CodecPrecision}_x = \frac{L - \sum_{i=1}^L |DCode_i(L) - ECode_i(L)|}{L} \quad (4)$$

where  $L$  is the length of watermark in bits,  $ECode$  denotes the bit value embedded during the embedding process and  $DCode$  denotes the output bit identified from the watermark extraction process.

The *CodecPrecision* of 30 files is shown in upper panel of Figure 3, where the x-axis denotes the file number, and the y-axis denotes the precision for each file with and without MP3 compression. The compression algorithm used was the *LAME MP3* encoder [12]. The mean precision of the watermark detection per audio file before and after applying MP3 compression was 98.79% and 98.72% respectively. This means that almost every watermark was successfully detected. By employing another popular audio tool-Adobe Audition to compress all the 30 music files, approximately the same result was obtained.

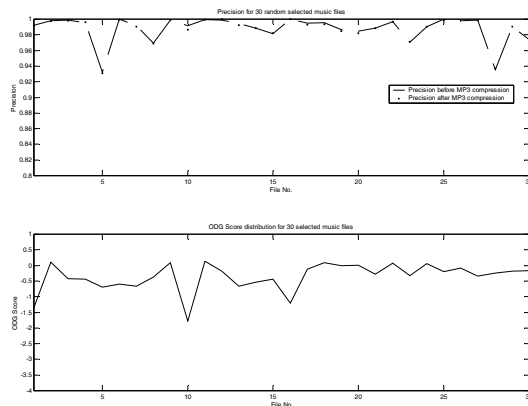


Figure 3 Precision and ODG distribution for 30 music files

In order to evaluate the perceptual transparency of the scheme, we performed a series of experiments to ascertain the ‘Objective Difference Grade’ (ODG) [11] using ‘*PQEvalAudio*’[13], which is an implementation of the ‘Perceptual Evaluation of Audio Quality’ (PEAQ) [11]. The ODG scale varies from 0 to -4, with 0 representing two files being perceptually identical. We compared 30 watermarked files against their original counterparts. The average ODG score was -0.3591 meaning that there was no noticeable perceptual difference between the watermarked files and their original counterparts. The distribution of ODG scores is shown in the lower panel of Figure 3. It can be seen that there are 3 music files with a score around -1.5 which implies that in these cases the differences are ‘perceptible but not annoying’ according to the ODG classification.

A subjective listening test was carried out on these three files and consequently it was found to be very difficult to detect any audible difference between the original and watermarked versions. To investigate further, we compared these three watermarked files’ spectrums with their original

counterparts’ spectrums to find the reason for these lower scores. It was discovered that there were some significant spectral differences. These were attributed to the threshold value of 0.08 as it was found that by lowering this value these differences were made smaller. A conclusion that can be drawn from this result is that it would be better if this threshold was adaptive to each specific audio frame rather than using a single global threshold.

## V. CONCLUSION

An algorithm for audio watermarking using the RSVD was presented in this paper. By modifying elements within the second column of the  $U$  matrix of the RSVD output, it was possible to embed watermark information that could survive the MP3 compression process with a minimum of audible distortion according to PEAQ tests. Experimental results showed a detection precision of around 99%. As for watermark capacity, we can embed around 11431 watermark data bits into a music file that has a length of around 4 minutes, assuming one bit in each frame.

Future work will consider to increase this capacity by developing the algorithm to embed more than one bit within each frame. Additionally, a psychoacoustic model will also be incorporated to create a frame-dependent adaptive threshold that will reduce the potential for audible distortion.

## VI. REFERENCES

- [1] J. Shieh, D. Lou, and M. Chang, “A semi-blind digital watermarking scheme based on singular value decomposition”, *Computer Standards & Interfaces*, April 2006, pp. 428-440
- [2] Dittmann, J et al ‘Theoretical framework for a practical evaluation and comparison of audio watermarking schemes in the triangle of robustness, transparency and capacity’, *Transaction on Data Hiding and Multimedia Security I*; pp. 1-40. Springer, 2006
- [3] L. Trefethen and D. Bau, *Numerical linear algebra*, SIAM : Society for Industrial and Applied Mathematics, PA, USA, 1997.
- [4] K. Chung, W. Yang, Y. Huang, S. Wu, and Y. Hsu, “On SVD-based watermarking algorithm”, *Applied Mathematics and Computation*, 188(1), May 2007, pp. 54-57.
- [5] M. Fan and S. Li, “Restudy on SVD-based watermarking scheme”, *Applied Mathematics and Computation*, 203(2), Sept. 2008, pp. 926-930.
- [6] R. Liu and T. Tan, “An SVD-based watermarking scheme for protecting rightful ownership”, *IEEE Trans. on Multimedia*, 4(1), Mar. 2002, pp. 121-128.
- [7] X. Zhang and K. Li, “Comments on “An SVD-based watermarking scheme for protecting rightful ownership” ”, *IEEE Trans. on Multimedia*, 7(2), Apr. 2005, pp. 593-594.
- [8] A. A. Mohammad, A. Alhaj, and S. Shaltaf, “An improved SVD-based watermarking scheme for protecting rightful ownership”, *Signal Processing*, 88 (9), Sept. 2008, pp. 2158-2180.
- [9] Y. Kim, H. Kang, K. Kim, and S. Han, “A Digital Audio Watermarking Using two masking effects”, *LNCS: Advances in Multimedia Information Processing – PCM 2002*, 2532/2002, Springer, Berlin, Jan. 2002, pp. 105-115.
- [10] V. Bhat K, I. Sengupta, and A Das, “An adaptive audio watermarking based on the singular value decomposition in the wavelet domain”, *Digital Signal Processing*, Article in press, available online Feb. 2010.
- [11] D. Câmpeanu and A. Câmpeanu, ‘PEAQ – An Objective Method To Assess The Perceptual Quality of Audio Compressed Files’,

Proceedings of International Symposium on System Theory, SINTES 12, Craiova, România, October 2005, pp. 487-492.

[12] LAME MP3 encoder, Available <http://lame.sourceforge.net/> [Accessed: 10 May 2010 ]

[13] P. Kabal, 'An implementation of the Perceptual Evaluation of Audio Quality (PEAQ) algorithm in matlab', Available <http://www.mmsp.ece.mcgill.ca/Documents/Downloads/PQevalAudio/> [Accessed: 10 May 2010 ]