INTERNATIONAL JOURNAL OF
RESEARCH IN COMPUTER
APPLICATIONS AND ROBOTICS
ISSN 2320-7345

BLIND AUDIO WATERMARK BASED CONTOURING TECHNIQUE FOR REAL-TIME APPLICATIONS

Asst. Prof. Dr. Hala B. Abdul Wahab[1] Asst. Prof. Dr. Abdul-Mohssen J. Abdul-Hossen[1]
Sana Ahmed Kadhom[2]

[1]Department of Computers, University of Technology, Baghdad, Iraq
[2]Department of Computer Science, Al Mammon University College, Baghdad, Iraq

Abstract: - Audio data has been used in many cyberspace applications such as voice calls, games, songs distributing, controlling, and many other applications. Audio watermarking techniques have been increased and expanded extremely in the last few years with the extreme expansion of world connectivity using the World Wide Web and what is lately called cyberspace. In this paper a proposed audio watermarking model is suggested for creating the watermark from audio file itself using a mathematical model for analyzing the audio signal in selected frames. Contouring technique is used to specify a sub area from the selected frame to extract features and create the watermark. The proposed mathematical model with the contouring technique decreased watermarking time and preserve audio integrity which makes the system applicable for on-line and real-time applications. The proposed system was tested on different types of audio files and compared to other previous real-time systems using SNR, BER, Time and other metrics.

Keywords: Audio, watermark (WM), Contouring technique, Feature extraction, Real-time.

I. Introduction

Steganography is one of the most effective methods used in securing transmitted media in the last few decades. Watermarking techniques have been used with all different types of media for many reasons such as authentication, identification, ownership, and others[1]. A blind watermarking system limits the exchanged data between the sender and the receiver to minimum. One of these limitations in blind watermarking is that, the receiver should have no previous knowledge about the watermark.

In blind watermarking systems, the watermark is created from audio file itself which means that the watermark differs from audio to another and is created depending on the features of the audio file [2]. A new mathematical model is produced here where audio features are calculated mathematically by analyzing each specific wave constructing a frame signal to create a watermark. If more than one frame is selected to extract features then the extracted watermarks will have different sizes and values.

A contour is a process which is used in many applications, especially images, like image recognition and objects separation. The algorithm used in this process is usually called "contour tracing algorithm"[3]. Basically contouring means: finding an object within an image by determining the object's boundaries. In the proposed system, contouring will be used with audio. Contouring audio gives a very different meaning since no object's boundary is tried to be found. Contour is employed here to determine an area for extracting features instead of using the whole frame.

In previous related work, Megias et al [4], suggested a blind watermarking method using Fast Fourier Transform FFT or Discrete Fourier Transform DFT, and a synchronization method in time domain; the watermarking process is applied in positions separated by gaps; the method is considered blind, robust and imperceptible, but many secrete information should be exchanged between sender and receiver such as SYN bits, their length, number of embedding SYN samples, length of WM, number of samples the WM will be embedded in, and the FFT positions used for embedding. Rami et al [5] suggested an audio watermarking by dividing the original audio file into a number of sub audio files; the watermark is embedded in each block, two bits at a time, to increase speed and payload; the embedding process uses Fast Fourier Transform FFT coefficients with the highest amplitudes. In the extraction process, FFT was applied on watermarked audio and the coefficients are sorted (ascending order), then two bits are extracted from each coefficient. The algorithm was tested on (mono) audios. The work is applicable for real time applications by measurements results.

In the next sections the proposed mathematical model for feature extraction (create watermark), the contouring technique, the hiding process, qualification metric measurements, and a comparison with previous work will be explained.

II. Feature extraction model (create watermark)

In the proposed system, chosen frames will be analyzed and specific values will be calculated depending on the shape of the waves constructing each frame. From each wave (positive or negative, figure 1), many features will be found such as the number of points representing the wave, the start point, the last point, the middle point, the maximum peak, and the distance between the middle point and the maximum peak (radius of the half circle the wave represents). The calculated features are unique for each individual wave, and off course, for each individual frame where these features will be used as a watermark.

The proposed mathematical model is used for the first time in extracting features and creating watermarks. It gives unique, self-extracted, and different watermarks with each audio file.

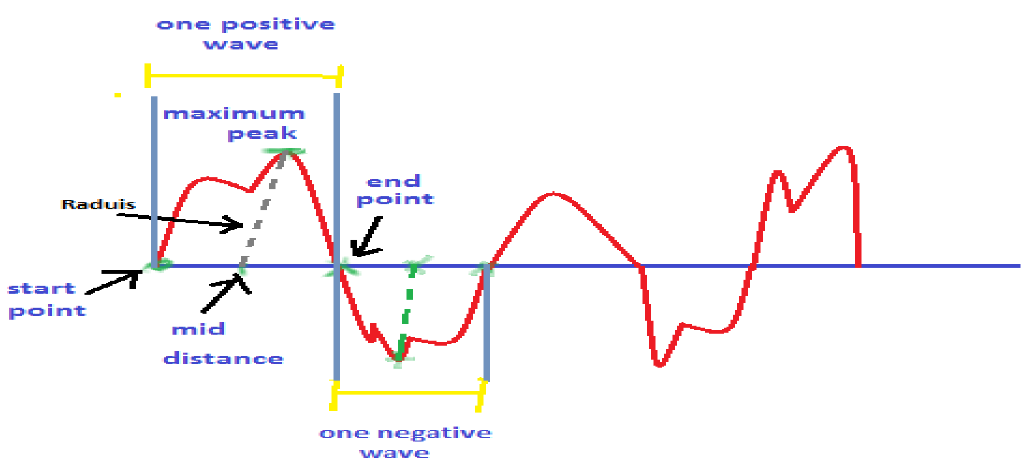


Figure 1: Wave analysis

III. Extracting Features Based on Contouring Technique

In the proposed system, contouring will be used with audio. Contour is employed here to determine an area for extracting features instead of using the whole frame.

Contouring technique is used for two important reasons. Firstly, to increase the complexity (for hackers) of determining the positions of features (which are used as a watermark) since part of the frame will be used not all of it and that part is determined by some control points. Secondly, to decrease time (extraction time, hiding time, and verification time) since the features will be calculated to part of the frame not the whole frame and the watermark created will be smaller than the watermark created for the same frame using the whole frame.

The proposed contouring technique has the following steps:

1. Read a frame of size (256 samples) from selected position.
2. Convert the frame into a two-dimensional matrix (16*16).
3. Find five control points, where each point (x,y) represents a position within the matrix.
4. Create a close curved area (polygon) from those five points.
5. Select all samples included in that polygon (contouring area) where this contouring area will be used for feature extraction.

IV. Hiding watermark in audio file

Audio signals in time domain (spatial domain) is very sensitive to changes therefore, hiding process will be applied on frequency domain in which each value is constructed from many samples in spatial domain. In other words, that any change made to a value in frequency domain is actually distributed between many samples of audio signal in spatial domain such that the change effects are spreading between many samples and no distortion could occur on audio signal at all[6].

The created watermark will be hidden in specific bits of the transformed coefficients. The proposed method of feature hiding is illustrated in the following procedure steps:

1. Read audio frame in the predetermined position with a size suitable to the watermark to be hidden. Each extracted features (created watermark) has its own size (different sizes for different watermarks) therefore, the sizes of frames used for hiding suit the sizes of created watermarks (variant length).
2. For each frame, transform to frequency domain. The result will be a set of coefficients.
3. In each coefficient produced from the transform, hide one bit of watermark in a selected bit of the binary representation of a coefficient.
4. After hiding watermark is completed, apply the inverse transform on the watermarked frame.
5. Return the watermarked frame to its original position in audio.

An important step in hiding process is the selection of the bits where the watermark will be hidden. After transformation of audio new numbers will be produced, normally fraction numbers, a proposed method was introduced for selecting a non constant but yet nonaudible, bit position for watermarking. The steps of the proposed bit selection method are:

1. Read the number (coefficient) from transformed data;
2. Take the first digit from left to be used in determining the position of watermarking bit;
3. Apply (p = digit mod 3 + 1), Modula 3 is useful in constrain result to be less than 3 (+1 is used to ensure (p) is 1,2 or 3 that force the modification to be on the first three digits only);
4. Take the last digit from right of coefficient to be used in hiding watermark;
5. Convert into 4 bit binary;
6. Replace the bit number p of the binary representation of the last digit with the watermark bit;
7. Convert binary into decimal (N);
8. Apply N mod 10;
9. Return the digit to its place;

Example on bit selection steps:

Suppose the coefficient is : 0.62318 , First digit = 6

P = digit mod 3 + 1, P = (6 mod 3) + 1 = 0 +1 = 1
Last digit = 8 , binary(8) = 1000
Suppose watermark bit to be hidden = 1; Replace the bit number P of the binary representation of the last digit with the watermark bit 1000 → 1001, Convert to decimal N; 1001 = 9, 9 mod 10 = 9, the new coefficient = 0.62319

From previous example, it is very clear that sometimes the coefficient may not be changed (Suppose the watermark bit is 0, instead of 1 then the new value will be 0.62318 which is the same as the original one), but even it does where the change will be very small that won't affect the audio since the change is done in the fifth fractional digit of the coefficient and this change is not audible nor effectible to the audio signals.

V. Implementation of the proposed system

In the proposed blind watermarking method, watermarks are created from contouring area of each selected frame by extracting the features. The created watermarks were hidden in other frames after applying transform and select bits positions to hide the watermark's bits. Finally, inverse transform is applied on watermarked frames.

To clarify the implementation of the proposed system, the same audio file was watermarked with three watermarks. These watermarks were created once by using the features extracted from the whole selected frames, and second, by using contouring technique. Figures 2,..,5 show the implementation of the system using first method (extract features without contouring).

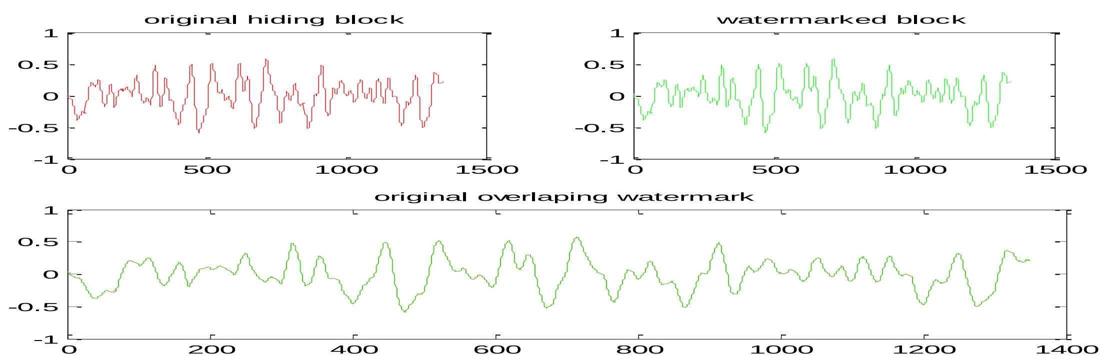


Figure 2: The first watermarking frame

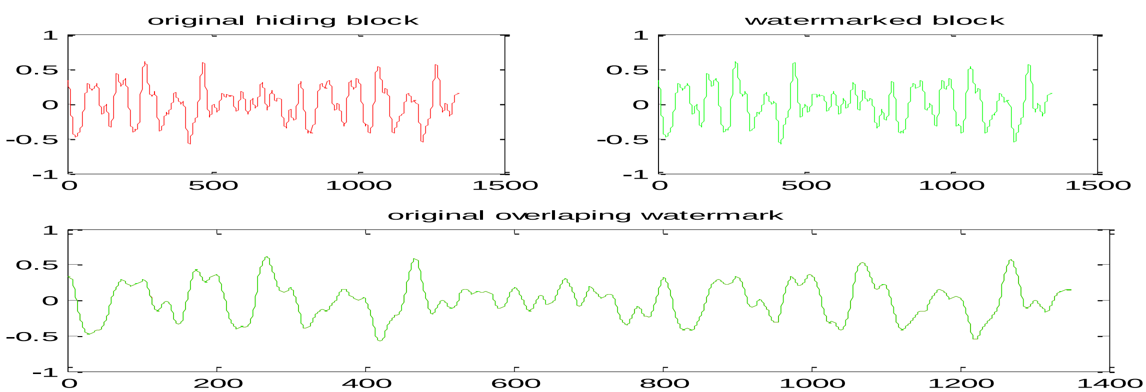


Figure 3: The second watermarking frame

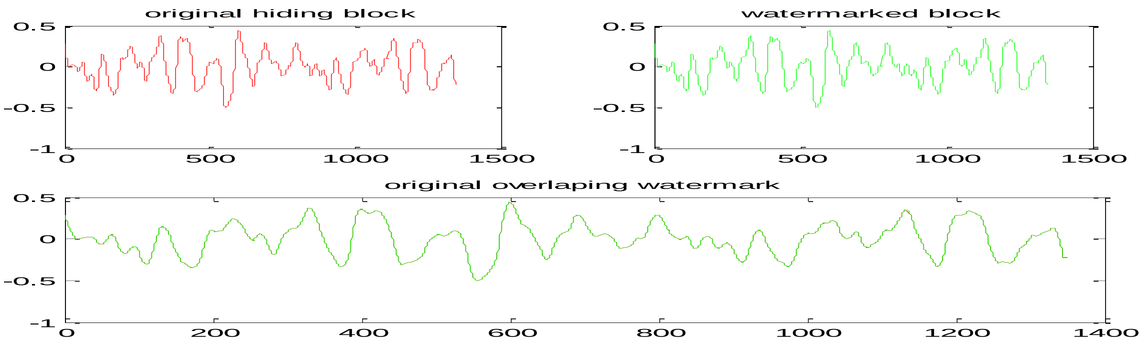
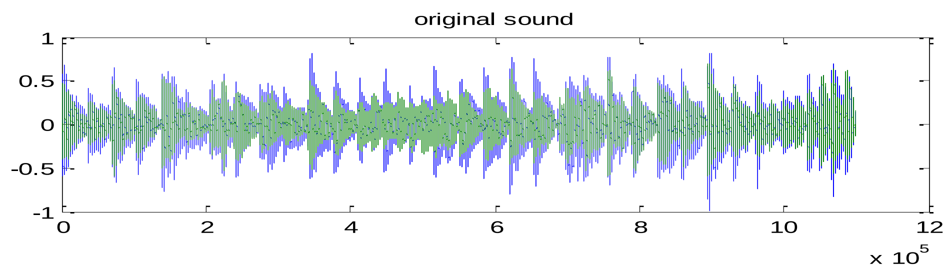
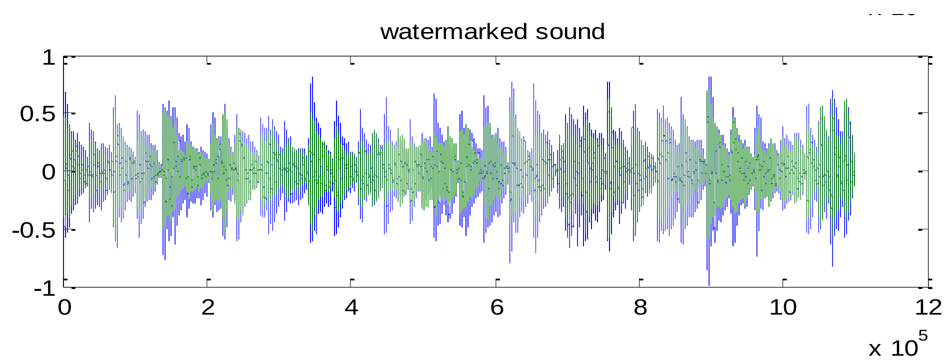


Figure 3: The third watermarking frame

Figure 4 shows the complete input audio file before watermarking (a) and the same audio file after watermarking (b).



(a)



(b)

Figure 4: Original audio file (a), the watermarked audio file (b)

(Extract features using the whole frame)

By applying contouring technique on the same audio file with three watermarks, the results were different since contouring extract features from part of the frame not the whole frame. Figure 6 illustrates the first chosen frame to be used in creating watermark; the frame has 256 samples.

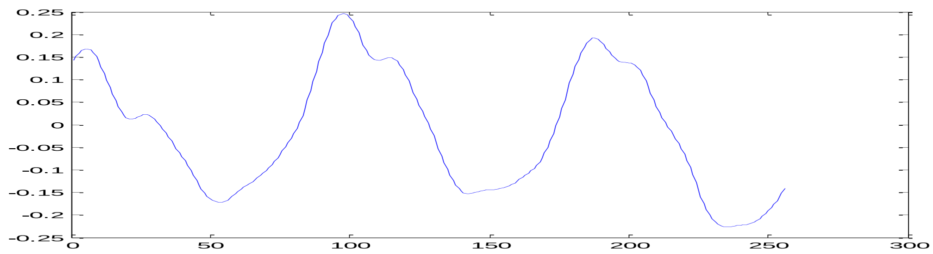


Figure 6: The first frame (for feature extraction… create watermark)

The contouring area created is illustrated in figure 7 below:

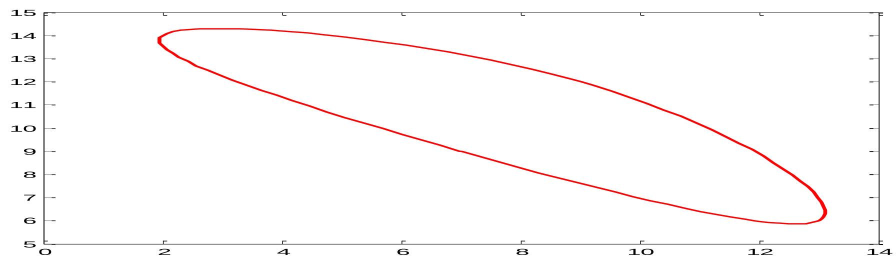


Figure 7: The first contouring area

The selected contouring area was applied on the original frame to extract part of the frame to be used in feature extraction (create watermark) process. The resulted new frame has fewer samples than the original one and off course a different shape. The new frame is illustrated in figure 8:

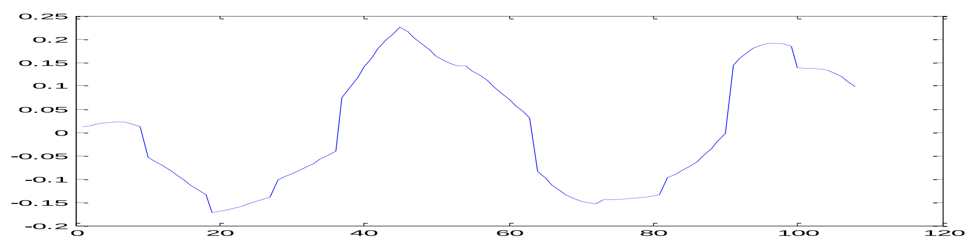


Figure 8: The new frame to create watermark (after contouring)

Watermark was created from the new frame and hidden in another chosen frame where the original and watermarked frames are shown in figure 9:

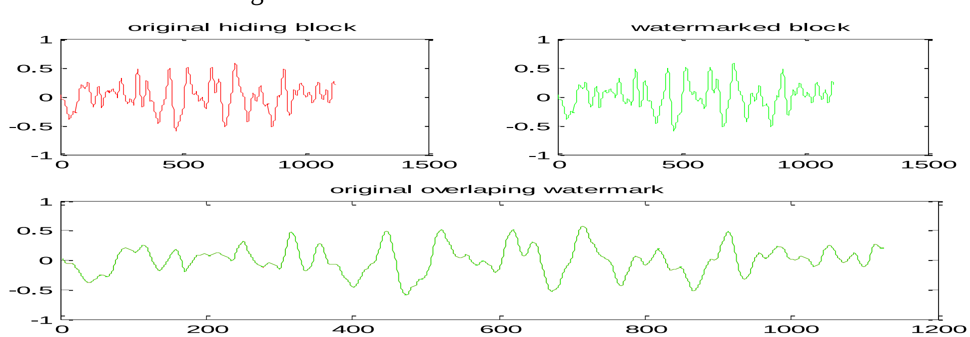


Figure 9: The first watermarked frame

Vol.5 Issue 10, Pg.: 13-25
October 2017

The second frame was chosen to create the second watermark as shown in figure 10:

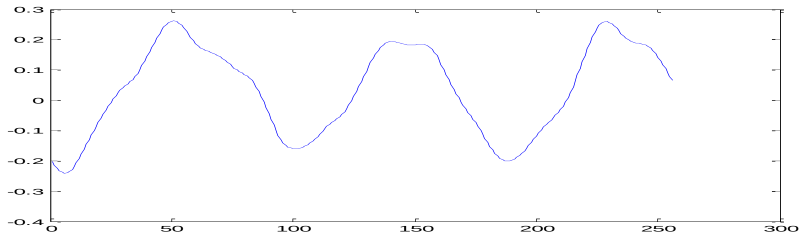


Figure 10: The second frame (for feature extraction… create WM)

The second contouring area created was illustrated in figure 11:

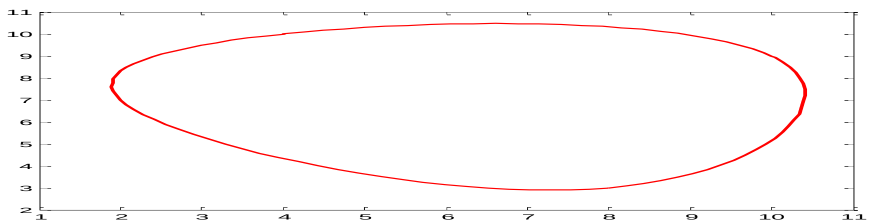


Figure 11: The second contouring area

The selected contouring area was applied on the second original frame to extract part of the frame to be used in feature extraction (create second WM) process. The resulted new frame has fewer samples than the original one and a different shape. The new frame is illustrated in figure 12:

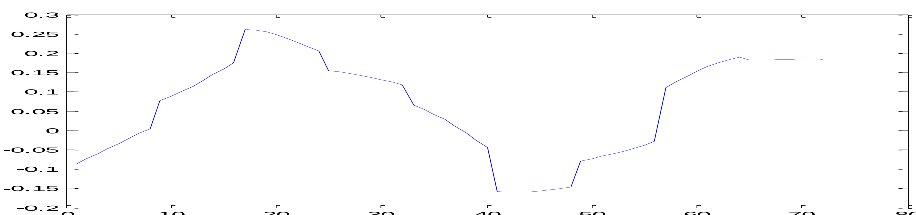


Figure 12: The new frame to create second WM

The second watermark was created from the new frame and hidden in the second chosen frame. The original and watermarked frames are shown in figure 13:

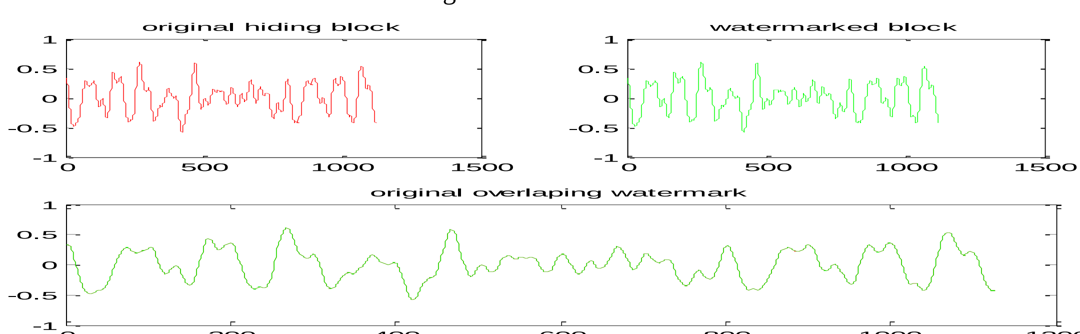


Figure 13: The second watermarked frame

The same steps will be applied on the third WM and the results are shown in figures 14, 15, 16 and 17:

Vol.5 Issue 10, Pg.: 13-25
October 2017

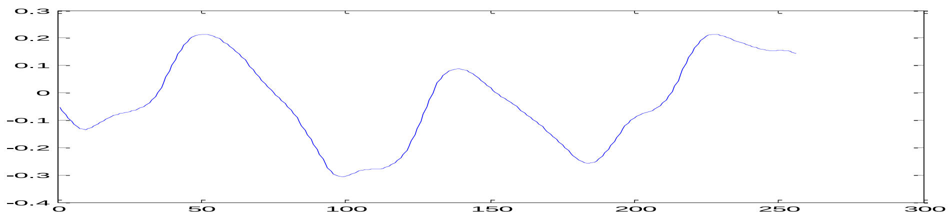


Figure 14: The third frame (for feature extraction… create WM)

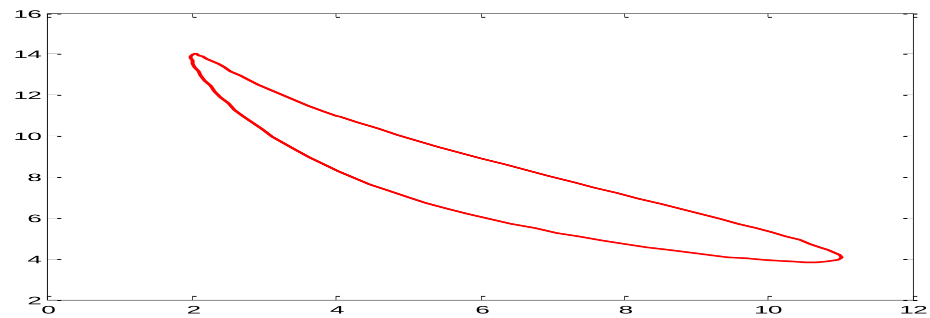


Figure 15: The third contouring area

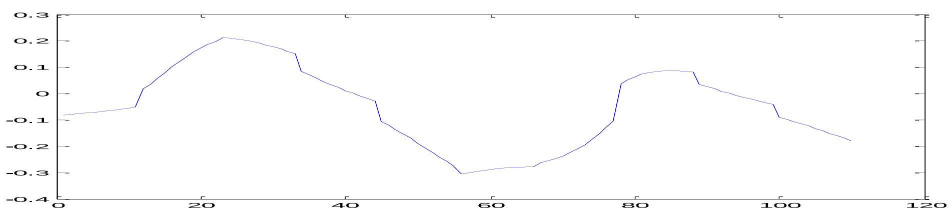


Figure 16: The new frame to create WM

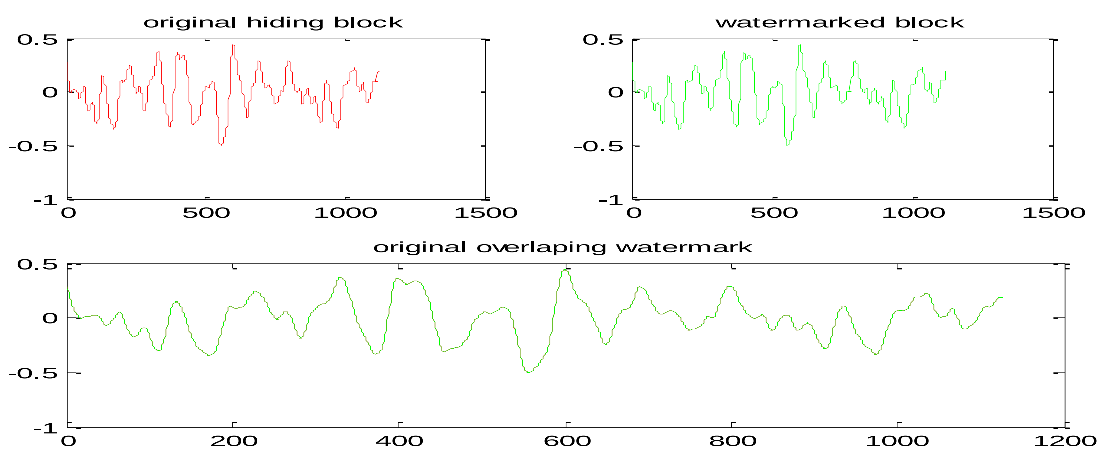
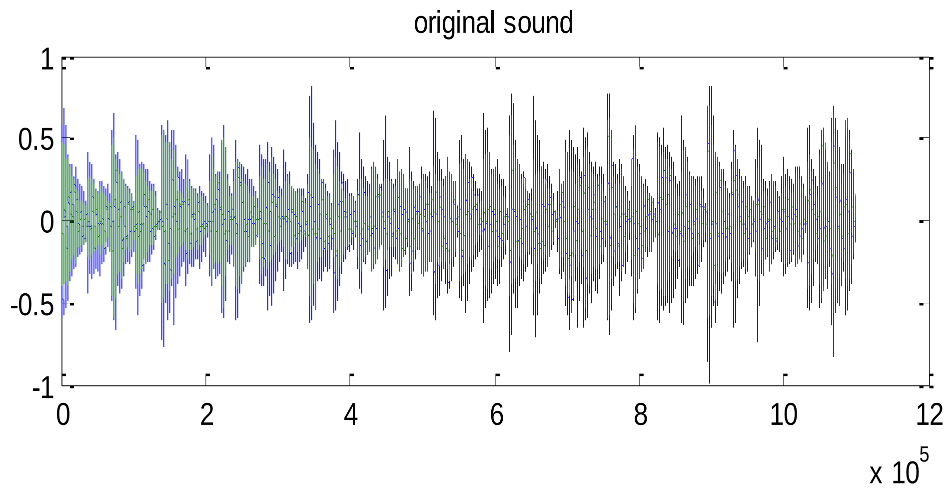
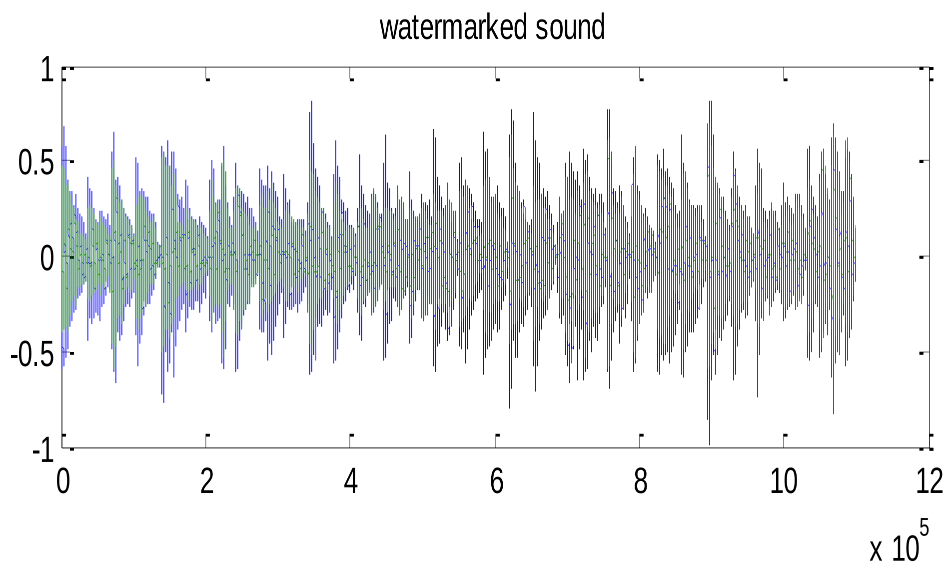


Figure 17: The third watermarked frame

The complete audio file before watermarking (a), and after watermarking (b) is shown in figure 18- a and b:



(a)



(b)

Figure 18: Original audio file (a), Watermarked audio file (b)

VI. Qualification measurements and comparison results

The proposed system was built in a manner that satisfies the integrity of audio file to be watermarked. Integrity may be tested by many metrics, BER: this metric gives the exact ratio of changes made to audio file. It is found by counting the number of the wrong bits (watermark bits not equal to audio bit). MSR: is the square norm for difference between input signal and the approximation. The result is divided by total number of samples. MAXERR: is the absolute maximum square deviation of the approximation of the input signal. $PSNR_{dB}$ represents the maximum noise happened in one sample (each sample represented with 16 bit) its value should be large. SNR_{dB} represents the ratio of signal to the noise where its value should be greater than (20 dB). Table 1 shows the previous metrics for nine audio files watermarked by the proposed system with and without contouring technique.

Vol.5 Issue 10, Pg.: 13-25
October 2017

Table 1: Integrity measurements of the proposed system

Audio file name	Audio length	BER contour	BER without	SNR_{dB} contour	SNR_{dB} without	PSNR contour	PSNR without	MSR contour	MSR without	MAXERR contour	MAXERR without
W10	651535	0.5153	0.5023	40.2950	39.0211	161.7379	160.4250	4.3580 e-012	5.8963 e-012	1.0596 e-004	1.0602 e-004
W14	1099640	0.4833	0.4986	45.7885	43.8925	167.2820	165.3411	1.2159 e-012	1.9010 e-012	1.0577 e-004	1.0577 e-004
W18	962808	0.4832	0.4901	41.4944	37.9496	162.8717	159.4086	3.3567 e-012	7.4511 e-012	1.0595 e-004	1.0602 e-004
Jas2	655010	0.5043	0.4855	39.5218	35.5025	160.8485	156.9019	5.3485 e-012	1.3271 e-011	1.0602 e-004	1.0577 e-004
Jas4	3303936	0.5385	0.5249	46.8018	43.9031	168.1008	165.2014	1.0069 e-012	1.9631 e-012	1.0592 e-004	1.0604 e-004
Jas5	5210285	0.5157	0.4978	48.7410	45.7268	167.1765	164.1023	1.2457 e-012	2.5284 e-012	1.0599 e-004	1.0591 e-004
Jas6	487392	0.5238	0.5176	39.6875	37.0674	158.1776	155.5517	9.8928 e-012	1.8110 e-011	1.0583 e-004	1.0591 e-004
Noise	1165279	0.5132	0.5150	39.0133	37.2623	157.6355	155.8248	1.1208 e-011	1.7006 e-011	1.0602 e-004	1.0605 e-004
Host_n	1310020	0.5250	0.5176	42.6678	37.9713	161.0180	156.3557	5.1437 e-012	1.5049 e-011	1.0600 e-004	1.0600 e-004
Average	1649545	0.511367	0.505489	42.6679	39.81073	162.7609	159.9014	3.63211111	2.89816667	1.0594	1.05943333

The previous table shows that:

- The values of SNR_{dB} were rated between 35 and 45 for the proposed system if no contouring applied while the same metric rated between 38 and 46 when contouring applied. For both types, the value of SNR_{dB} was larger than 20dB which is the minimum acceptable value.
- PSNR has a minimum value of 155 where that value and others were increased by using contouring to reach 168.
- MSR and MAXERR were very small (they have to be very small values), and applying contouring technique has minimized these values.
- BER rated between 48 and 53 this means that in each watermarking only half the watermark bits had changed the audio bits value.

Table 2 represents the time consumed in sender side only for both methods (if contouring is applied or not).

Table 2: Time measurements for the sender stages with and without contouring

Audio file name	Audio duration (MM:SS)	Extract contour	Extract without	Hide WM contour	Hide WM without	CPU time contour	CPU time without
W10	00:14	0.0148	0.0031	0.1098	0.1887	0.1247	0.1918
W14	00:24	0.0134	0.0039	0.0685	0.1227	0.0819	0.1266
W18	00:21	0.0222	0.0033	0.1707	0.3508	0.1929	0.3542

Jas2	00:14	0.0262	0.0039	0.1696	0.3617	0.1957	0.3656
Jas4	01:14	0.0324	0.0031	0.1405	0.2425	0.1730	0.2456
Jas5	02:42	0.0198	0.0047	0.2008	0.3070	0.2206	0.3117
Jas6	00:15	0.0140	0.0030	0.1120	0.1997	0.1259	0.2027
Noise	02:25	0.0327	0.0045	0.2399	0.4768	0.2726	0.4814
Host_n	00:29	0.0144	0.0053	0.1752	0.4140	0.1896	0.4193
Average	**0:55**	**0.0211**	**0.003867**	**0.154111**	**0.295989**	**0.175211**	**0.299878**

From table 2, it is clear that the time needed for extracting features to create watermark with contouring is larger than the time needed for the same stage without contouring and this is logically true since additional procedure was added to find contouring area and isolates the samples not included in that area. The hiding WM stage with contouring has less time than the same stage without contouring. This is again logically true since the size of the created watermark from contouring area is smaller than the watermark created from the whole frame without contouring.

The total time consumed by the sender if using contouring was noticeably reduced sometime to half the original time and this means that contouring has a great effect on the time consumed by sender.

Receiver, on the other hand, has common stages with sender also has stages of finding hidden WM and verify (compare) it. Table 3 shows the time consumed by receiver stages.

Table 3: Time measurements for the receiver stages with and without contouring

Audio file name	Audio duration (MM:SS)	Feature Extract contour	Feature Extract Without	Find Hidden & compare contour	Find Hidden & compare Without	CPU time Contour	CPU time without
W10	00:14	0.0148	0.0031	0.0401	0.0597	0.0592	0.0631
W14	00:24	0.0134	0.0039	0.0210	0.0368	0.0408	0.0425
W18	00:21	0.0222	0.0033	0.0626	0.0921	0.0723	0.0965
Jas2	00:14	0.0262	0.0039	0.0589	0.1111	0.0744	0.115
Jas4	01:14	0.0324	0.0031	0.0358	0.0787	0.0534	0.083
Jas5	02:42	0.0198	0.0047	0.0328	0.0837	0.0352	0.0884
Jas6	00:15	0.0140	0.0030	0.0300	0.0971	0.0909	0.101
Noise	02:25	0.0327	0.0045	0.0785	0.1252	0.0899	0.1298
Host_n	00:29	0.0144	0.0053	0.0593	0.131	0.0739	0.1348
Average	**0:55**	**0.0211**	**0.003867**	**0.04655556**	**0.0906**	**0.06555556**	**0.0949**

Table 4 describes a comparison between the proposed system embedding timing with two other systems [4] and [5] considered to be suitable and applicable for real time applications. The proposed system has the minimum timing.

Table 4: Speed comparison with two real time systems

Watermark scheme	Speed of embedding bps	Time for each bit Sec
[22]	338.03	0.003
[11]	1171.43	0.00085
Proposed	**3291.4**	**0.000030382**

Table 5 shows the SNRdB, for three audio signals watermarked by the proposed system and the other two previous works. The proposed system has the largest SNRdB values for all three files.

Table 5: SNRdB comparision schemes for the same audio files

Audio file name	SNRdB for different schemes for the same audio files		
	[11]	[22]	**Proposed**
Rust	26.55	25.15	**38.2761**
Stop Payment	22.88	22.53	**43.4535**
Flood Plain	24.75	24.63	**49.0868**

Conclusion

The proposed system was implemented and tested on different types of audio. The results showed that the proposed system was fast with a minimum CPU timing comparing with previous works. It was proved that the integrity of audio files was preserved. The proposed method contouring was used to determinate a specific area within a frame to be used in feature extraction (create WM). This new method proved to be useful in decreasing time and BER and increasing SNR and PSNR according to the results described in previous tables.

REFERENCES

[1] Juergen Seitz, “Digital watermarking for digital media”, University of Cooperative Education, Heidenheim, Germany, 2005.

[2] Harleen Kaur, Usvir Kaur, “Blind Audio Watermarking schemes: A Literature Review”, IRACST – Engineering Science and Technology: An International Journal (ESTIJ), ISSN: 2250-3498,Vol.3, No.2, 2013.

[3] Thomas A. Lampert, “Spectrogram Track Detection an Active Contour Algorithm”, pattern recognition letters, DOI:10.1016/j.patrec .2009.09.021, 2010.

[4] D Megias, J Serra-Ruiz, and M Fallahpour, “Efficient self-synchronized blind audio watermarking system based on time domain and fft amplitude modification”, signal processing, 90(12), 2010.

[5] Rami Almasi, Hooman Nikmehr, “A Highly-Payload Audio Watermarking Scheme for Real-Time Application”, Journal of computing and security, Vol. 2, No. 2, 2015.

[6] Syed Ali Khayam, ‘The Discrete Cosine Transform: Theory and Application’, Information Theory and Coding, 2003.

[7] http://www.mathworks.com/help/wavelet/ref/wavedec.html.

[8] William K, Wen-Hsiung Chen, and Lloyd R. Welch, "Slant Transform Image Coding", IEEE, Transaction on communications, VOL. COM-22, NO. 8, 1974.

[9] Baiying Lei, Ing Yann Soon, Feng Zhon, Zhen Li, Haijun Lei, ' A Robust watermarking scheme based on lifting wavelet transform and singular value decomposition', signal processing 92, 2012.

[10] Hala B., Abdual Mohsen J., Sana Ahmed, 'Variant length, Self-extracted audio watermark for verification using LWT and random selections', Diyala Journal for Pure Science DJPS, ISRA Impact Factor: 3.715, P-ISSN: 2222-8373, E-ISSN:2518-9255, Vol. 13, Issue 2, Part 2, 2017.

[11] Mallat, S., "A theory for multi-resolution signal decomposition: the wavelet representation," IEEE Pattern Anal. And Machine Intell., vol. 11, no. 7, pp 674–693, 1989.