Research Article Impact Factor: 4.226 ISSN: 2319-507X
Piyush S. Jain, IJPRET, 2015; Volume 3 (9): 427-437 IJPRET

INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

RECONFIGURABLE ARCHITECTURE FOR AUDIO WATERMARKING USING EMD

PIYUSH S. JAIN, DR. SURESH N. MALI

Sinhgad Institute of Technology and Science Narhe, Pune-41, India

Accepted Date: 05/03/2015; Published Date: 01/05/2015

Abstract: Digital watermarking is the technique of inserting specific information into signal, data, image or video. Copyright protection, data authentication, covert communication and content identification can be achieved by Digital watermarking. This work aims at developing a robust and secure blind watermarking technique for speech signal. The speech is watermarked using Virtex-5 FPGA which implements a MicroBlaze soft core processor. The watermarking is done using EMD algorithm in MATLAB which is converted to C/C++ code to be used in FPGA through MATLAB Coder and Xilinx EDK software. The developed hardware finds use in public safety digital radio communications and such an interoperable emergency communication is integral to initial response, public health, safety of communities, national security and economic stability.

Keywords: Empirical mode decomposition, intrinsic mode function, audio watermarking, FPGA, Microblaze, virtex-5, Xilinx, Matlab

Corresponding Author: MR. PIYUSH S. JAIN

Access Online On:

www.ijpret.com

How to Cite This Article:

Piyush S. Jain, IJPRET, 2015; Volume 3 (9): 427-437

PAPER-QR CODE

INTRODUCTION

Due to rapid progress in wireless communication systems, extreme prevalence mobile systems, and smart card technology, information is more vulnerable to abuse. For these reasons, it is important to make information systems secure to protect data and resources from malicious acts. Digital watermarking has been developed as an effective solution for multimedia data protection. Watermarking usually assumes embedding of secret signal that should be robust and imperceptible within the host data. Also, reliable watermark detection must be provided. Digital watermarking has been proposed for a variety of applications, including content protection, authentication, and digital rights management. The current systems used for public safety radio are employed by police, fire department and other emergency services. These radios have a single encryption and are transmitted over open channels. Hence they need a more robust and secure method for checking the authenticity of the received signal. Moreover, open channel radio is highly vulnerable to attacks. To test whether the signal is genuine or not, a blind watermarking scheme should be very useful. Hence this project aims at developing digital watermarking hardware for security and authorisation purposes. Many works have been reported for audio watermarking based on discrete wavelet transform (DWT) method that claim increased bit rate, increased SNR, minimum audio-cover period, quality of the watermarked audio and watermark robustness against audio attacks as seen in [2] and [3]. A new method of first decomposing the signal by singular value decomposition and then applying watermark has the advantage of high improvement in audio-cover period is demonstrated in [4]. DCT method has been used successfully in Large capacity digital audio watermarking also employing DWT as reported in [5]. Various other attempts of audio watermarking using DCT scheme have been effective with different advantages like in [6] is tested for additive noise and cropping and method in [7] show good robustness to resample and MP3 compression attacks. A blind watermarking method using DCT is presented in [8]. This project deals with signal processing through FPGA hardware. The target hardware is the Genesys circuit board which is a complete, ready-to use digital circuit development platform based on a Xilinx Virtex-5 LX50T. The large on board collection of high-end peripherals, including Gbit Ethernet, HDMI Video, 64-bit DDR2 memory array, and audio and USB ports make the Genesys board an ideal host for complete digital systems, including embedded processor designs based on Xilinxs MicroBlaze. Genesys is compatible with all Xilinx CAD tools, including ChipScope, EDK, and the free WebPack, so designs can be completed at no extra cost. The signal processing and watermarking is done in MATLAB. The MATLAB code is later converted to C/C++ code through MATLABCoder tool. A FPGA consists of reconfigurable logic blocks and reconfigurable interconnects. These reconfigurable logic blocks and interconnects can be made to behave like a processor. Such a processor is called a soft core processor because it is not actually present but the FPGA is acting like one. Xilinx FPGA and software tools support the design and implementation of the Microblaze soft core processor on the FPGA. The C/C++ code generated is used to program the MicroBlaze soft core processor in the Virtex-5 FPGA using Xilinx ISE Design Suite software and its allied utilities. The results of watermarking can be verified using MATLAB. Also the watermarked audio signal can be tested against various attacks in MATLAB. This complete work flow of the project is shown in figure 1.

I. AUDIO WATERMARKING IN MATLAB

The idea of the proposed watermarking method is to hide into the original audio signal a watermark together with a Synchronized Code (SC) in the time domain. The input signal is first segmented into frames and EMD is conducted on every frame to extract the associated IMFs. Then a binary data sequence consisted of SCs and informative watermark bits is embedded in the extrema of a set of consecutive last-IMFs. A bit (0 or 1) is inserted per extrema. Since the number of IMFs and then their number of extrema depend on the amount of data of each frame, the number of bits to be embedded varies

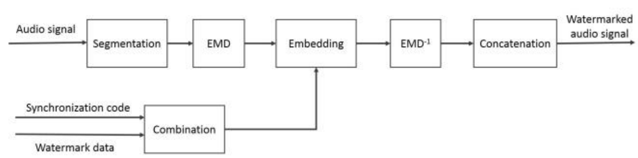
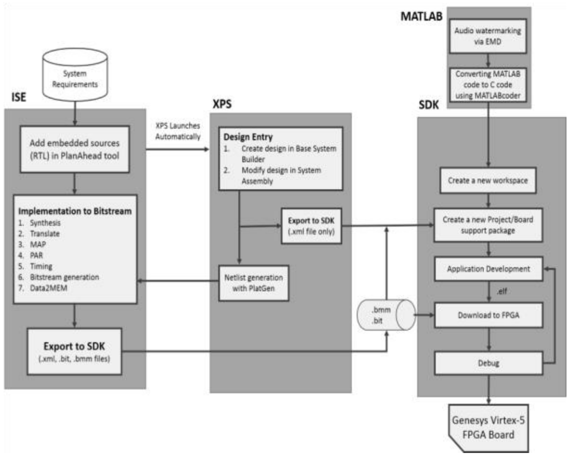


Fig. 2. Watermark Embedding

From last-IMF of one frame to the following. Watermark and SCs are not all embedded in extrema of last IMF of only one frame. In general the number of extrema per last-IMF (one frame) is very small compared to length of the binary sequence to be embedded. This also depends on the length of the frame. The figure 3.2 shows the watermark embedding procedure. If we design by N_1 and N_2 the numbers of bits of SC and watermark respectively, the length of binary sequence to be embedded is equal to $2N_1 + N_2$. Thus, these $2N_1 + N_2$ bits are spread out on several last-IMFs (extrema) of the consecutive frames. Further, this sequence of $2N_1 + N_2$ bits is embedded P times. Finally, inverse transformation (EMD^{-1}) is applied to the modified extrema to recover the watermarked audio signal by superposition of the IMFs of each frame followed by the concatenation of the frames. For data extraction, the watermarked audio signal is split into frames and EMD applied to each frame. Binary data sequences are extracted from each last IMF by searching for SCs.



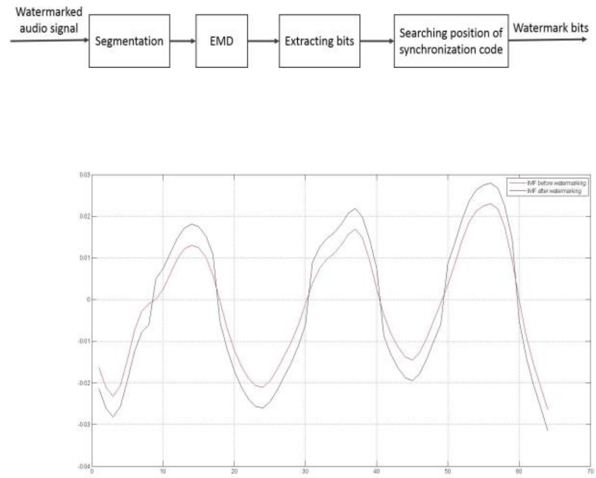


Figure 1. Project work Flow **Figure 3. Water Mal Extraction**

EMD being fully data adaptive, thus it is important to guarantee that the number of IMFs will be same before and after embedding the watermark. In fact, if the numbers of IMFs are different, there is no guarantee that the last IMF always contains the watermark information to be extracted. To overcome this problem, the sifting of the watermarked signal is forced to extract the same number of IMFs as before watermarking. The proposed watermarking scheme is blind, that is, the host signal is not required for watermark extraction (Figure 3). Audio watermarking is done using MATLAB in which the audio file of .wav format is taken as input and a watermark of binary image in .png format is applied to it. The watermarked audio is saved bearing the name ’watermarked audio’ in .wav format. The steps undertaken for watermark embedding and extraction are given below.

A. Synchronisation Code

To locate the embedding position of the hidden watermark bits in the host signal a SC is used. This code is unaffected by cropping and shifting attacks. Let *U* be the original SC and *V* be an unknown sequence of the same length. Sequence *V* is considered as a SC if only the number of different bits between *U* and *V* , when compared bit by bit, is less or equal than to a predefined threshold *τ.*

B. Watermark Embedding

Before embedding, SCs are combined with watermark bits to form a binary sequence denoted by $m_i \in \{0,1\}$, i-th bit of watermark. Basics of our watermark embedding are shown in figure 3.2 and detailed as follows:

Step 1:	**Split original audio signal into frames.**
Step 2:	Decompose each frame into IMFs.
Step 3:	Embed *P* times the binary sequence $\{m_i\}$ into extrema of the last IMF (IMF_C) by QIM :

Step 4: Reconstruct the frame (EMD^{-1}) using modified IMF_C and concatenate the watermarked frames to retrieve the watermarked signal.

$$e_i^* = \lfloor e_1/S \rfloor \bullet S + sgn(3S/4) \ if \ m_i = 1 \quad (1)$$

$$e_i^* = \lfloor e_1/S \rfloor \bullet S + sgn(S/4) \ if \ m_i = 0 \quad (2)$$

Where e_i and e_i^* are the extrema of of the host audio signal and the watermarked signal respectively. sgn function is equal to ”+” if e_i is a maxima, and ”-” if it is a minima. bc denotes the floor function, and S denotes the embedding strength chosen to maintain the inaudibility constraint.

The Figure 4 below shows the last IMF of an audio frame before and after watermarking.

C. Watermark Extraction

For watermark extraction, host signal is split into frames and EMD is performed on each one as in embedding. We extract binary data using rule given by (4.2). We then search for SCs in the extracted data. Watermarking embedding and extraction processes are summarized in Figure 4.2. and the binary watermark is shown in Figure 4.3.

This procedure is repeated by shifting the selected segment (window) one sample at time until a SC is found. With the position of SC determined, we can then extract the hidden information bits, which follows the SC. Let $y = \{m_i\}$ denote the binary data to be extracted and denote the original SC. To locate the embedded watermark we search the SCs in the sequence $\{m_i^*\}$ bit by bit. The extraction is performed without using the original audio signal. Basic steps involved in

The watermarking extraction, shown in Figure 3.3 are given as follows:

Step 1:	**Split original audio signal into frames.**
Step 2:	Decompose each frame into IMFs.
Step 3:	**Extract the extrema** $\{e_i^*\}$ of IMF_C.
Step 4:	**Extract from using the following rule:** $m^*_i = 1$ *if* $e^*_i - be_1/Sc \geq sgn(S/2)$ **(3)** $m^*_i = 0$ *if* $e^*_i - be_1/Sc < sgn(S/2)$ **(4)**
Step 5:	Set the start index of the extracted data, *y*, to I=1 and L=N_1 select samples (sliding window size).
Step 6:	Evaluate the similarity between the extracted segment V = *y*(I:L) and U bit by bit. If the similarity value is ≥ τ ,then V is taken as the SC and go to Step 8. Otherwise proceed to the next step.
Step 7:	Increase I by 1 and slide the window to the next L=N_1 samples and repeat Step 6.
Step 8:	Evaluate the similarity between the second extracted segment, $V^0 = y(I + N1 + N2 : I + 2N1 + N2)$ and U bit by bit.
Step 9:	$I \leftarrow I + N_1 + N_2$, of the new I value is equal to

Sequence length of bits, go to Step 10 else repeat Step 7.

Step 10: Extract the *P* watermarks and make comparison bit by bit between these marks, for correction, and finally extract the desired watermark.

III. RESULTS OF AUDIO WATERMARKING

The performance of this method is evaluated in terms of Bit

Error Rate (BER). According to International Federation of the Photographic Industry (IFPI) recommendations, a watermark audio signal should maintain more than 20 dB SNR. To evaluate the watermark detection accuracy after attacks, we used the BER defined as follows:

$$BER(W, \acute{W}) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} W(i,j) \oplus \acute{W}(i,j)}{M \times N} \qquad (5)$$

where is ⊕ the XOR operator and $M \times N$ are the binary watermark image sizes. W and W' are the original and the recovered watermark respectively. BER is used to evaluate the watermark detection accuracy after signal processing operations.

To assess the robustness of our approach, different attacks are performed:

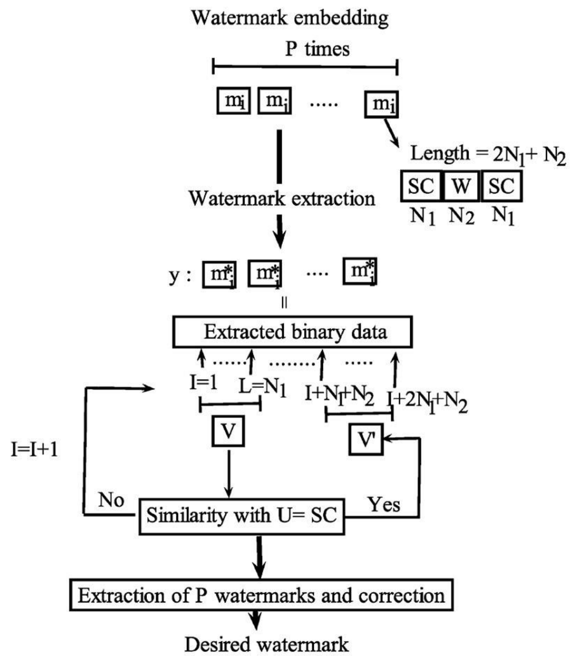


Fig. 5. Embedding and extraction processes

Fig. 6. Binary watermark

- Noise: White Gaussian Noise (WGN) is added to the watermarked signal until the resulting signal has an SNR of 20 dB. Even Pink noise was added and BER was calculated.

- Filtering: Filter the watermarked audio signal using Wiener filter.

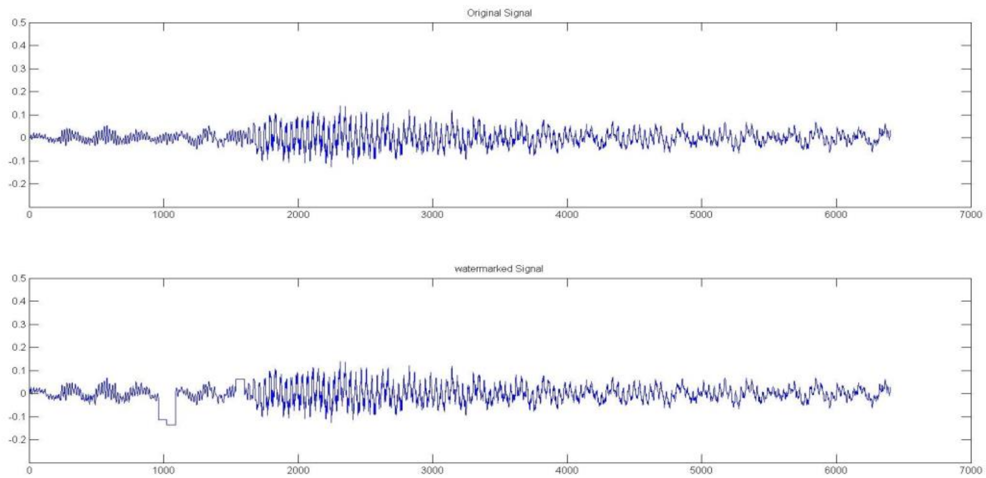


Fig. 7. Audio signal and its watermarked version

- Cropping: Segments of 512 samples are removed from the watermarked signal at thirteen positions and subsequently replaced by segments of the watermarked signal contaminated with WGN.

TABLE I. BER OF EXTRACTED WATERMARK FOR AUDIO SIGNAL BY PROPOSED APPROACH

Attack Type	% BER
No Attack	2.2461
Gaussian Noise	10.0586
Filtering	14.1602
Cropping	4.9805
Resampling	2.2461
Pink Noise	11.2305

TABLE II. RESOURCE UTILIZATION IN FPGA

Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	6,298	28,800	21 %
Number of Slice LUTs	5,399	28,800	18 %
Number used as logic	5,080	28,800	17%
Number used as Memory	307	7,680	3%
Number of occupied Slices	2,926	7,200	40 %
Number of LUT Flip Flop pairs used	8,445		
Number with an unused Flip Flop	2,147	8,445	25%

Number with an unused LUT	3,046	8,445	36%
Number of fully used LUT-FF pairs	3,252	8,445	38 %
Number of slice register sites lost	1,231	28,800	4%
Number of bonded IOBs	148	480	30%
Number of LOCed IOBs	148	148	100%
IOB Flip Flops	295		
Number of BlockRAM/FIFO	25	60	41%
Total Memory used (KB)	900	2,160	41%
Number of BUFG/BUFGCTRLs	6	32	18%
Number of IDELAYCTRLs	3	16	18 %
Number of BSCANs	1	4	25%
Number of BUFIOs	8	56	14%
Number of DSP48Es	3	48	6 %
Number of PLL ADVs	1	6	16 %

- Resampling: The watermarked signal, originally sampled at 44.1 kHz, is re-sampled at 22.05 kHz and restored back by sampling again at 44.1 kHz.

Table I shows various attacks ans the percentage bit error rates calculated. Also figure 5.2 shows the original and watermarked version of the audio signal.

IV. RESULTS OF FPGA IMPLEMENTATION

The developed system for audio recording/playback is synthesized and configured in Xilinx ISE Design Suite. As can be seen in Table 5.2, the main consumed FPGA resources are logic components; in addition, the hardware resources in FPGA are adequate. The software platform Xilinx ISE Design Suite automatically utilizes the logic components to realize all components needed by soft core processor. Notably, the slice register utilisation is below 20%, which indicates that more complex circuitry can be implemented in the remaining resources of the FPGA.

The Genesys board includes a National Semiconductor LM4550 AC 97 audio codec (IC19) with four 1/8 audio jacks for line-out (J16), headphone-out (J18), line-in (J15) and microphone-in (J17). Audio data at up to 18 bits and 48-kHz sampling is supported, and the audio in (record) and audio out (playback) sampling rates can be different. The microphone jack is mono; all other jacks are stereo. The headphone jack is driven by the audio codec’s internal 50mW amplifier. The input given to this board at the time of testing was from a laptop through its line-in jack. The audio recording/playback performance was good with no audible distortions during playback.

V. CONCLUSION

In this project a new adaptive watermarking scheme based on the EMD is proposed. Watermark is embedded in very low frequency mode (last IMF), thus achieving good performance against various attacks. Watermark is associated with synchronization codes and thus the synchronized watermark has the ability to resist shifting and cropping. Data bits of the synchronized watermark are embedded in the extrema of the last IMF of the audio signal based on QIM. Extensive simulations over different audio signals indicate that the proposed watermarking scheme has greater robustness against common attacks than nine recently proposed algorithms. In all audio test signals, the watermark introduced no audible distortion. Experiments demonstrate that the watermarked audio signals are indistinguishable from original ones. These performances take advantage of the self-adaptive decomposition of the audio signal provided by the EMD. The proposed scheme achieves very low false positive and false negative error probability rates. This watermarking method involves easy calculations and does not use the original audio signal. In the conducted experiments the embedding strength is kept constant for all audio files. To further improve the performance of the method, the parameter should be adapted to the type and magnitudes of the original audio signal.

REFERENCES

1. Kais Khaldi and Abdel-Ouahab Boudraa, “Audio Watermarking Via EMD”, IEEE Transactions on Audio, Speech, And Language Processing, Vol. 21, No. 3, March 2013.

2. Alshammas H. A., “Robust audio watermarking based on dynamic DWT with error correction”, Proceedings of ITU Kaleidoscope: Building Sustainable Communities (K-2013), April 2013.

3. Elshazly A.R, Fouad M.M., “Secure and robust high quality DWT domain audio watermarking algorithm with binary image,” Proceedings of Seventh International Conference on Computer Engineering Systems (ICCES), Nov. 2012.

4. Al-Yaman, Al-Taee, “Audio-watermarking based ownership verification system using enhanced DWT-SVD technique,” Proceedings of 9th International Multi-Conference on Systems, Signals and Devices (SSD), March 2012.

5. Keqiang Ren, Huihuan Li, “Large capacity digital audio watermarking algorithm based on DWT and DCT,” International Conference on Mechatronic Science, Electric Engineering and Computer (MEC), Aug. 2011.

6. Duymaz E., Akan A., “A novel DCT based digital audio watermarking method,” 20th Signal Processing and Communications Applications Conference (SIU), April 2012.

7. Xia Zhang, Di Chang, Wanyi Yang, Qian Huang, et. al., “An audio digital watermarking algorithm transmitted via air channel in double DCT domain,” International Conference on Multimedia Technology (ICMT), July 2011

8. Maher E., Mohamed, Koubaa, Chokri B.A., “DCT Based blind audio watermarking scheme,” IEEE International Conference on Signal Processing and Multimedia Applications (SIGMAP), July 2010.

9. N.E.Huang et al., “The empirical mode decomposition and Hilbert spectrum for nonlinear and non-stationary time series analysis,” Proc. R. Soc., vol. 454, no. 1971, pp. 903995, 1998.