A STUDY ON AN EFFECTIVE MUSIC DISTRIBUTION SYSTEM FOR ONLINE DIGITAL MUSIC INDUSTRY

SU LATT SANDI

M.C.Tech. **FEBRUARY 2020**

A STUDY ON AN EFFECTIVE MUSIC DISTRIBUTION SYSTEM FOR ONLINE DIGITAL MUSIC INDUSTRY

By

Su Latt Sandi

B.C.Tech.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Computer Technology

(M.C.Tech.)

University of Computer Studies, Yangon

February 2020

ACKNOWLEDGEMENTS

Firstly, I am using this opportunity to express my gratitude to my parents who are the most important persons for me. I am thankful for their aspiring guidance, invaluable constructive criticism, and friendly advice throughout the whole of my life.

I would like to show sincere thanks to Dr. Mie Mie Thet Thwin, Rector of the University of Computer Studies, Yangon, for kindly giving me this opportunity and leading me working on a diverse exciting thesis.

My special thanks are passed to Dr. Khin Than Mya, Professor and Head of the Faculty of Computer Systems and Technologies, University of Computer Studies, Yangon, for her invaluable guidance and administrative support.

I would also like to express my gratitude to Dr. Thet Thet Khin, Professor and Course coordinator of the Master (thesis) course of the University of Computer Studies, Yangon, who shared her pearls of wisdom and supported with required components during the course of this thesis.

I would like to express my warm special thanks to my supervisor, Dr. Twe Ta Oo, Lecturer, Faculty of Computer Systems and Technologies, for encouraging on my research and for her priceless comments and suggestions. Her support, guidance, and overall insights in this audio scrambling field have made this an inspiring experience for me.

I would also like to give my appreciation and sincere honor to Daw Aye Aye Khine, Head of the Department of Language, University of Computer Studies, Yangon, for her kindly checking and editing my thesis from the language point of view.

I am also very thankful to Dr. Khin Mar Soe, Professor, Natural Language Processing Lab, University of Computer Studies, Yangon, and Dr. Wynn Htay, Principal of Japan IT and Business College, for enriching my ideas. They are more than generous with their expertise and precious time for reviewing my thesis.

I am also very grateful to all of my teachers from the University of Computer Studies, Yangon, for their valuable comments, suggestions, helpful hints, and fullest cooperation during the seminars of my thesis.

Last but not least, I am sincerely grateful and give special thanks to my parents and friends who believed in me, push, and advised me to complete my thesis and everyone who shared their truthful and illuminating views on a number of issues related to the thesis.

ABSTRACT

Due to the vast improvements in the Internet and telecommunication technologies and digital multimedia platforms, the music industry has already been moving into online space successfully these days. Services of digital music market such as online streaming, downloading, and online music stores are extremely popular among young consumers.

Online music distribution brings several benefits such as less coordination and distribution costs for distributors, and cost and time effectiveness for buyers. Along with the benefits, it has also brought some challenges: how to securely distribute songs online, how to control illegal access to songs, how to share teaser/sample music for potential buyers to taste the songs. The proposed system in this thesis is intended to satisfy the above requirements of online digital music industry.

The proposed system is developed based on the interesting relation between the Discrete Wavelet Transform (DWT) and the human auditory system. To understand the proposed method properly, detail and approximation coefficients resulting from the DWT decomposition of an audio/music signal should be clearly understood. The details are the high-frequency components of an audio signal, while the approximations correspond to the low-frequency components. The normal human ear is usually most sensitive in the low-frequency range. Thus, the approximation coefficients are more important for audio quality. Slight modification on those coefficients will severely degrade the audio quality, and whereas the details are not that much sensitive to the human ear.

Based on the above nature of the DWT, the proposed system is developed as follows. An audio/music signal is first broken down into different DWT wavelet layers. Then, the proposed audio scrambling method is applied on each layer with different keys from a pre-generated key table. The resulting music signal with scrambled detail coefficients has degraded quality but is still understandable the lyrics and the essence of the music. Thus, it can be used as teaser for potential buyers. Otherwise, if stronger security is needed for music distribution to buyers, only the approximations (or) all coefficients can be scrambled. This will yield the music with very low quality which is nearly noisy. Without knowing the keys, anyone can never restore the original music quality. In this way, unauthorized access to music can also be efficiently controlled.

The proposed system is implemented in MATLAB R2017a. Experimental results show that the proposed method is simple to implement and very effective in terms of low computational complexity and fast execution time. In addition, it supports flexible direct control on the music quality. Thus, the proposed system is applicable for online digital music industry.

TABLE OF CONTENTS

PAGE

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1

INTRODUCTION

This paper aims to develop an efficient audio scrambling method that can solve most of the challenges found in online digital music industry. This chapter firstly introduces the challenges of that industry and points out the inefficiency of conventional cryptographic techniques in solving those challenges. It also highlights the advantages of audio scrambling over traditional encryption for online music distribution. Finally, the chapter is concluded by presenting the objectives and organization of the thesis.

1.1 System Background

No one could deny the fact that the Internet has changed the music industry drastically over the last couple of decades. With the advancement of digital multimedia technologies and increased availability of multimedia data, digital music and video are sharing over the Internet to an extent that cannot be imagined. To be specific, the music industry has already been shifting successfully from traditional shops (physical sales) to online stores (digital market).

In U.S., as a result of the shift in music consumption, digital music accounted for over half of all the revenue generated by the music industry in 2017 and amounted to a total of US$ 2.8 billion that year [13]. According to the International Federation of the Phonographic Industry (IFPI)'s Global Music Report 2019, the global recorded music market grew gradually from 2015 to 2018. It increased by 9.7 % in 2018 with a total reported revenue of US$ 19.1 billion. The 37 % of that total revenue came from the 255 million users of paid streaming services as reported at the year's end [21].

Before online music services are popular, people had to go to local music shops to buy song albums. They also had to wait for international music to be available in local shops; it might also be very expensive and scarce. However, with online music stores, global users can now have easier and cheaper access to any desired music, no need to care about geographical locations. In addition, online stores allow users to choose the songs

they wish instead of having to purchase an entire album in which there may be only one or two tracks that the buyer enjoys.

Also for the music distributors, online distribution has allowed for potentially lower expenses such as lower coordination, distribution, and production costs as well as the possibility for redistributed total profits. The first company that was able to create a successful online service for legal sales and distribution of music was Apple Inc. The iTunes Music Store was the first online retailer that was able to offer the music catalogs from all the major music companies.

Recently in Myanmar, music distributors are more using online not only to advertise the new songs but also to distribute music to buyers. For example, Legacy Music Network is the first leading online music distributor in Myanmar [40]. It deals with individual artists and labels in Myanmar music industry since 2011. Myanmar Music Store (MMS) provided by Legacy is the largest online music library that offers over 500,000 tracks. The MMS advertises their songs by sharing the chorus or short clip of the music as teaser to potential buyers. Hence, the buyers can conveniently guess the essence of the music and buy any desired track online.

However, along with the merits, online music distribution has also brought some challenges for distributors such as loss of profits due to music piracy [9] [37]. The recording industry has suffered the most at the hands of technological change. At the end of the nineties, the explosion of online piracy services such as Napster allows users to download and share music without compensating the recognized rights holders [38]. Globally, the revenues from recorded music in its all forms fell by more than 40% between 1999 (its peak) and its nadir in 2011 [20]. It challenges the effectiveness of the Digital Rights Management (DRM) systems by creating the problems of copyright violations and insecure data distribution.

Also for teaser sharing, if the distributor uploads the whole song as teaser, there are potentially high risk of illegal downloads. Even for sharing the chorus as teaser, illegal downloading can still occur for using as ring tunes. Therefore, the profit of online music industry largely depends on the effective control of unauthorized access to music and its future is in danger.

In order to avoid illegal download, some music service providers are now using the whole song with degraded quality or ad-based version as teaser. As an example, the Internet radio and streaming service called Spotify [36] offers a freemium service where it models with two or more different music versions. The most basic version is free and the more advanced versions are offered on a subscription basis. To convert the users of free version to subscription version, the free version must have a number of increasingly annoying features such as advertising. As another example, JOOX music application [23], which is also a freemium service, provides most of its songs free with degraded quality. However, high-fidelity songs are only available for premium users and offered via paid subscriptions or by doing some requested tasks such as watching the advertisements.

The challenge for those freemium online music distribution services like Spotify and JOOX is to balance the free version in such a way that it must be good enough to entice the customers to become paying subscribers. However, it should not be so good that the customer satisfies the free version and not willing to upgrade to premium users.

This paper implements a system that can solve the above-mentioned challenges of online digital music industry: how to distribute songs online securely, how to control illegal access to songs, how to create teaser music to entice the buyers to buy the songs. The proposed system is implemented in the wavelet domain by deploying the effect of wavelet coefficients on the Human Auditory System (HAS). It is modeled to generate different-quality music files where low-quality files can be used as teaser and severely quality degraded files can be used to control illegal access.

The following section introduces the techniques that can possibly be used to solve the above-mentioned challenges of online music industry.

1.2 Audio Encryption vs. Audio Scrambling

No need to doubt, a straightforward way to protect any kind of data including audio is to apply the traditional cryptographic algorithms such as DES, AES, etc. These algorithms can effectively be used to degrade the audio quality. Encrypted audio will become like noise and without knowing the decryption key, anyone can never recover the original music quality. In this way, unauthorized access to music has also been controlled.

However, the above feature alone is not satisfactory for online music distribution. The primary reason is that the audio media is an instance of plaintext which has specific patterns of coherence that can be judiciously exploited [42]. Based on the audio format, e.g. uncompressed (WAVE, AIFF & PCM) or compressed (MP3 & AAC) formats, each audio file must follow the predefined multimedia standard [7]. Moreover, the HAS is more sensitive than the Human Visual System (HVS). Slight changes on the audio samples can create drastic damage on the perceptual quality of that audio. Thus, blind encryption on audio data will break the multimedia standard and as a result, the encrypted audio may not be played back by standard media players. Thus, encryption can be used to degrade the audio quality; however, that quality-degraded music cannot be used as teaser.

Fortunately, audio scrambling methods have come into handy for online music services. They can overcome the above-mentioned problem of encryption by obeying the media standard even after scrambling. Audio scrambling is similar to but not a direct application of usual cryptographic techniques such as AES. It aims to minimize the residual intelligibility of an audio signal with the use of a certain secret key. Unlike encryption that uses complex mathematical operations such as substitution, permutation, transformation, shifting, etc., most scrambling methods try to permute only the order of the audio samples without changing the values [1] [17]. Its purpose is to break the coherence between audio samples without breaking the media standard. Mostly, the perceptible quality changes introduced by audio scrambling are very little compared to the effect of encryption. Thus, scrambling methods are more appropriate for the HAS in terms of the perceptible quality control. This thesis presents an audio scrambling method that obeys the media standard.

1.3 Objectives of the Thesis

The objectives of the proposed system in this thesis are as follows:

- To develop an effective online music distribution system as today music industry is moving into online digital markets
- To implement a system that can generate teaser music usable not only for tasting the music quality but also for persuading the buyers to buy the songs
- To study the effect of DWT on Human Auditory System and deploy it to control the progressive audio quality

- To provide secure music distribution via scrambling, which is an alternative of traditional encryption

1.4 Organization of the Thesis

This thesis consists of five main chapters. Chapter 1 introduces the challenges of online digital music industry and the techniques that can be used to solve those challenges. Chapter 2 discusses the theoretical background of the inclusive components and methods of the proposed system. Chapter 3 and 4 elaborate on the detailed implementation of the proposed system and experimental results, respectively. Finally, Chapter 5 concludes this thesis by discussing the benefits, drawbacks, and future works of the proposed system.

CHAPTER 2

BACKGROUND THEORY

As stated in Chapter 1, traditional encryption algorithms are not satisfactory for online music distribution due to their nature of blind encryption. This thesis presents an audio scrambling method which is a better alternative for music quality control than traditional encryption. It also provides flexibility to music distributors. In this chapter, the overview of Digital Signal Processing (DSP) techniques, encryption and scrambling methods, and Discrete Wavelet Transform, which are the backbone of this thesis, are discussed.

2.1 Overview of Digital Signal Processing

Digital Signal Processing (DSP) is one of the most powerful technologies that will shape science and engineering in the twenty-first century. Revolutionary changes have already been made in a broad range of fields: communications, medical imaging, radar & sonar, high fidelity music reproduction, to name just a few. Each of these areas has developed a deep DSP technology with its own algorithms, mathematics, and specialized techniques.

DSP also has a strong relationship with two principal human senses: vision and hearing. Over previous years, DSP techniques have made significant changes in the areas of image and audio processing [39]. Let us consider an example for audio processing. In a high-fidelity music reproduction system, an input signal representing music recorded on a cassette or compact disc is modified in order to enhance desirable characteristics such as removing the recording noise or balancing several components of the signal (e.g., treble and bass). All of these transformations can perfectly be done by using very simple and basic DSP techniques.

Basically, in audio processing, an audio clip whose nature is analog must first be converted to a digital stream before applying the DSP techniques. For digital conversion, an audio clip must firstly pass through the sampling process which changes the analog signal to a discrete-time signal. Then, DSP operations of interest such as de-noising can be applied on that discrete-time signal and the result must be converted to digital via the quantization and encoding processes. That digital stream can be transmitted, stored, and modified much more efficiently with higher speed and quality. If necessary, the end result

after applying DSP techniques can be converted back to analog (audio). Therefore, DSP in audio signal processing is concerned with audio-to-digital conversion, digital signal processor (typically a single microchip), and digital-to-audio conversion [44].

(i) **Audio-to-Digital conversion (ADC)** takes incoming analog signals and converts it to a series of binary data points. For example, the voltage signals coming out of an electric guitar cable change to a string of 1's and 0's that represents the magnitude the incoming voltages after the ADC process. Most specialized ADCs are implemented as integrated circuits [4].

(ii) **Digital Signal Processor** preforms numerous computation techniques such as de-noising, filtering, compression, etc. on those binary data streams. A DSP processor generally includes program memory, data memory, a compute engine, and input/output functions [14].

(iii) **Digital-to-Audio (D/A) conversion** takes binary data that comprise the music signal, converts to analog, and outputs an analog electronic signal. For example, a string of 1's and 0's changes back to the corresponding voltages of electric guitar signal after the DAC process.

In this thesis, the proposed system takes the sampled audio data and applies the proposed audio scrambling method on it. Then, the end result is converted back to analog audio signal to playback the music and converted to digital for storing on the computer.

2.2 Audio Formats

An audio coding format is a content representation format, i.e. bit layout of audio data excluding metadata, for storage or transmission of digital audio. Based on the coding format, the DSP processor should deal with the audio data differently. There are two basic kinds of audio formats: compressed [6] and uncompressed. We can easily check the audio coding format by looking at the file extension; e.g. "*.wav" means uncompressed format and "*.mp3" means compressed audio format.

Compressed audio format here refers to a lossy audio file format that does not decompress audio files to their original data amount. Lossy methods provide high degrees of digital compression which result in smaller files. Most popular lossy audio formats are Moving Picture Experts Group Layer-3 (MP3) and Advanced Audio Coding (AAC). They

are designed to greatly reduce the amount of data required to represent an audio recording and still sound like a faithful reproduction of the original uncompressed audio for most listeners. Nowadays, the MP3 is a de facto standard of digital audio compression for the transfer and playback of music on most digital audio players as well as a common format for consumer audio streaming or storage. The AAC is a common format found on the iTunes Music Store. These formats offer a range of degrees of compression, generally measured in bit rate. The bit rate means the number of bits used to encode the audio samples per second. The lower the bit rate then the smaller the file size and the more significant the quality loss. Therefore, lossy compression is not recommended in professional settings where high quality is necessary [5].

Uncompressed audio format is a lossless audio file format that stores data without losing any information. Popular uncompressed audio formats are Audio Interchange File Format (AIFF) and Waveform Audio Format (WAV). The AIFF is an audio coding format created by Apple and can be played on Macs. An audio signal encoded by AIFF format is identical to the CD quality audio. However, AIFF encoded audio files are large in size; thus, they take up significant space and time for storage and transmission [5].

The WAV is a Microsoft and IBM audio file format standard for storing raw PCM and typically uncompressed audio bitstream on personal computers [26]. It is the audio format mainly used on the Windows operating systems. Even though an uncompressed WAV file is large and thus not appropriate for file sharing over the Internet, it is a commonly used file type suitable for retaining audio files of high quality. For example, WAV files are commonly used in applications like audio editing where the time needed in compressing and decompressing data is a concern or on a system where disk space is not a constraint.

The most important parameters for controlling the audio quality in each audio coding format are the sample rate, bit depth, and bit rate. The sample rate means how many times per second a signal is captured and is measured in “Hz” (a unit of frequency describing cycles per second). Sampling rates for audio mostly range from 8 kHz to 192 kHz. The higher the sample rate then the higher the audio quality but the larger the file size. Bit depth and bit rate are also two important aspects of digitized sound. Bit depth is the number of bits used to encode each sample and bit rate measured in kilobits per second

(kbps) is the sample rate multiplied by the bit depth. As for professional sound quality, an audio signal needs to be at least the sample rate of 44.1 kHz and bit depths of 24 or 32 bit [7]. Table 2.1 shows the comparison of compressed and uncompressed audio file formats with their associated quality and file size.

The proposed system in this thesis implements an audio scrambling method for WAV audio formats, which are most suitable for retaining high quality music. Unlike other media like text, even slight modification introduced by scrambling or encryption on music/audio data will yield significant impact on the audio quality. This thesis presents a scrambling method not only to protect the audio from unauthorized access but also to provide controllable degradation on the audio quality. The following section discusses the audio encryption and scrambling methods.

Table 2.1: Comparison of compressed (MP3) and uncompressed (WAV) formats [7]

Audio Format	**Sample Rate**	**Bit Depth**	**Quality**	**Size**
.WAV	8 k - 16 kHz	8 bits	Poor	Small
	16 k - 32 kHz	16 bits	Fair	Medium
	44.1 kHz	16 bits	Excellent	Large
	48 kHz and above	16 bits – 32 bits	Pristine	Very Large
Audio Format	**Sample Rate**	**Bit Rate**	**Quality**	**Size**
.MP3	8 k - 16 kHz	16 - 96 kbps	Poor	Very small
	16 k - 32 kHz	96 - 196 kbps	Fair	Small
	44.1 kHz	256 - 320 kbps	Good	Medium
	48 kHz	320 kbps	Excellent	Large

2.3 Encryption Methods

Encryption is a technique used to transmit secure information. Over the years, several encryption techniques have been developed for encrypting text data. However, there were very few techniques specially designed for encrypting multimedia data such as audio. Generally, the techniques which encrypt text data can also be applied on audio data. This section introduces some of them.

There are two major kinds of encryption standards in the field of cryptography: symmetric and asymmetric encryption algorithms [25]. **Symmetric encryption**

algorithms use only one secret key to cipher and decipher the information, which should have already been agreed by the sender and the recipient. That secret key can either be a number, a word, or a string of random letters. Figure 2.1 depicts the symmetric key encryption and decryption processes.

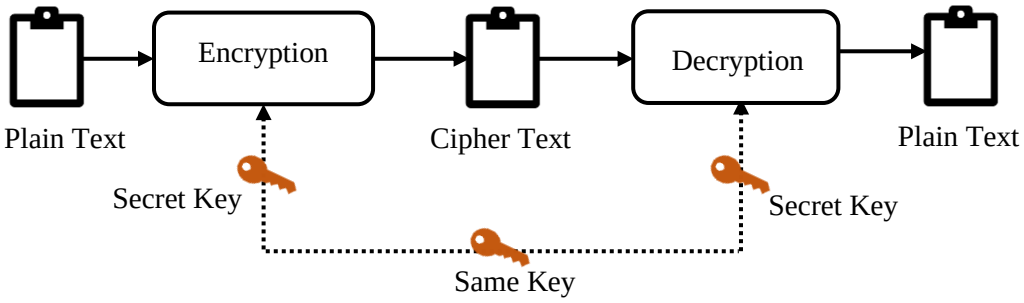


Fig. 2.1: Symmetric encryption

Some basic symmetric encryption algorithms are introduced below.

Data Encryption Standard (DES) is a symmetric-key block cipher algorithm primarily based on the symmetric rule published by the National Institute of Standards and Technology (NIST). DES operates on data blocks of 64 bits in size and each block is enciphered into a 64-bit ciphertext by means of permutation, substitution, and using a 56-bit secret key. DES algorithm is used in a wide variety of embedded systems, smart cards, SIM cards, and network devices like modems and routers requiring encryption [10].

Triple DES (TDES) is also a symmetric block cipher derived from the DES by using it three times. TDES was developed when the 56-bit key of the DES had been found to be not robust enough against brute force attacks and lots of alternative attacks. The TDES was created as the same algorithmic rule as the DES but with long key size. Despite of enhanced security, TDES is the slower algorithmic rule than the DES and thus has low performance in terms of power consumption [41].

Advanced Encryption Standard (AES) is also a symmetric block cipher with block size of 128 bits [3]. It uses variable key length of size 128, 192, and 256 bits. It is based on a design principle known as a substitution-permutation network. The cipher and decipher processes of the AES use a round function that is composed of four different byte-oriented transformations: SubBytes, ShiftRows, MixColumns, and AddRoundKey. The

number of rounds to be performed during the execution of the algorithm is dependent on the key length. It is well known for its tremendous data security [41].

Unlike the above-mentioned block ciphers, **Rivest Cipher 4 (RC4)** is a stream cipher that generates a pseudorandom stream of bits (a key stream) and uses it for encrypting the plaintext via the bit-wise XOR operation. Decryption is performed in the same way as the XOR with given data is an involution. The RC4 made use of a secret internal state while generating the key stream: a permutation of all 256 possible bytes and two 8-bit index pointers. The permutation is initialized with a variable key length (between 40 and 2048 bits) using the key-scheduling algorithm (KSA). Once this has been completed, the bits stream is generated by using the pseudo-random generation algorithm (PRGA). The main factors in RC4's success over a wide range of applications have been its speed and simplicity. Efficient implementations in both software and hardware are very easy to develop [31].

Asymmetric encryption, the next encryption standard of the cryptography, is known as the public key cryptography or asymmetric-key cryptography. It uses a pair of public and private keys to cipher and decipher the data. A public key means the one key in the pair that can be shared with everyone and the private key means the other key in the pair that is kept as the secret key. The sender can use the receiver's public key to encrypt a message and the encrypted message can only be decrypted with the receiver's private key [12]. Figure 2.2 depicts the asymmetric key encryption and decryption processes.

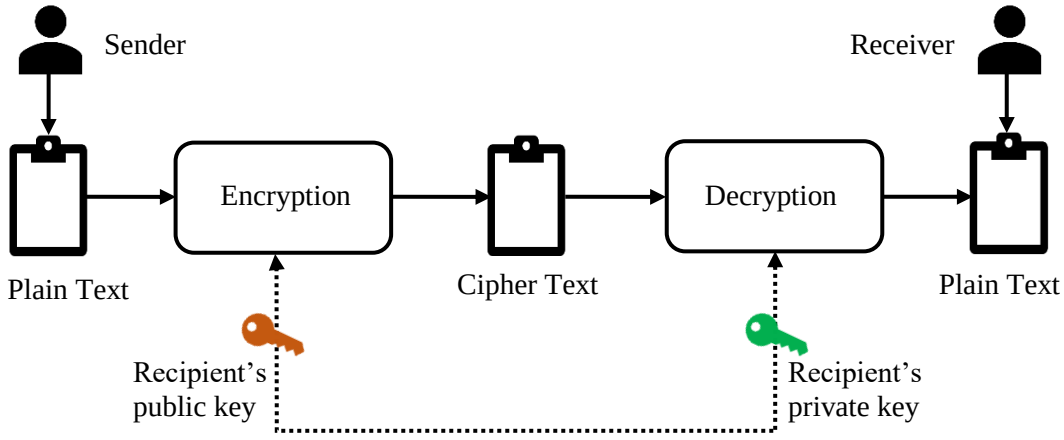


Fig. 2.2: Asymmetric encryption

The most widely used asymmetric encryption algorithm is the **Rivest Shamir Adleman (RSA)** algorithm. It is the public-key cryptosystem used for secure data transmission. In RSA, both the public and private keys can encrypt a message; the opposite key from the one used to encrypt a message is used to decrypt it. The public and private key generation is the most complex part of the RSA cryptography. Moreover, encryption strength is directly tied to the key size, and doubling the key size can deliver an exponential increase in strength, although it does impair the performance. It is used in the encryption of the software programs and the RSA signature verification is one of the most commonly performed operations in network-connected systems. However, the RSA algorithm is less commonly used to directly encrypt the user data [32].

All of the above-mentioned encryption techniques have already been applied for audio protection by many researchers. To pick out some of them as examples, Gadanayak, Pradhan, Raghunndhan, Radhakrishna, Sudeepa and Aithal [8] [30] presented how to apply the AES and modular multiplicative algorithms for audio protection, respectively. Experimental results indicated that uncompressed audio files which are larger in size took more time for encryption and decryption. Therefore, the authors applied the above algorithms on compressed audio files. They also proposed the idea of selective encryption, i.e. encryption is applied only on the selected important parts of an audio file rather than encrypting the whole file. The results showed that selective encryption is better than full encryption in terms of performance.

Sharma and Kumar proposed an audio protection system that applied the RSA algorithm on selected different frequency bands of an audio signal encoded in .wav format. Firstly, a time domain audio signal is first converted to frequency domain by using the Fast Fourier Transform (FFT). The FFT can separate a signal into different frequency regions with respective magnitude and phase values. The RSA algorithm is then selectively applied to the low frequencies which have higher magnitude values. It was mentioned that applying encryption on the lower frequency bands is more effective than applying on the higher frequency bands [33].

James, George, and Deepthi [34] also proposed an encryption method for compressed audio without degrading the audio quality. Encryption was done by only XORing the audio data with a key which was obtained by using the Linear Feedback Shift

Register. The proposed technique achieved the less hardware complexity and was capable of resisting different types of attacks like ciphertext only attack and known plaintext attack.

Traditional cryptographic algorithms mentioned above can be used for audio protection and to prevent unauthorized access to music. However, this feature alone is not satisfactory for online music distribution systems in which not only audio quality needs to be degraded but also that quality degraded file should be playable as teaser music to tease the potential buyer. However, the encrypted media file yielded by the above-mentioned systems cannot be playable by standard media players. Moreover, an audio file has its own encoding standard and encryption should not destroy the standard.

In addition, human auditory system is more sensitive than the human visual system. Slight modification on the audio samples can lead to a noisy signal. Most of the above-mentioned encryption algorithms use a mixture of complex mathematical operations like substitution, shifting, permutation, transformation, and so on. The amount of destructing the audio contents by those methods cannot be controllable. As a result, the encrypted audio files cannot be played back without decryption.

Fortunately, audio scrambling methods can overcome that problem by obeying the media standard even after scrambling and by providing flexible direct control on the audio quality.

2.4 Audio Scrambling

Nowadays, audio scrambling methods are widely used to provide confidentiality in audio distribution. Audio scrambling methods are kinds but not direct applications of usual encryption techniques mentioned above. They only try to reduce the residual intelligibility of an audio signal by breaking the coherence between data contents so that the signal is unintelligible to unintended recipients of the communication. Most audio scrambling methods are based on transposition/permutation and they neither inject any new values nor change the values of the existing contents. Thus, they do not destroy the audio format or neither increase file size.

A good scrambling method should meet the following requirements [42].

(1) **Security:** Even if the algorithm is distributed in public, descrambling should still be difficult.

(2) **Efficiency:** Scrambling and descrambling processes should not be too complex.

(3) **Perceptual quality:** Scrambled signal should achieve very low or nearly zero residual intelligibility and descrambling process must also be able to recover an audio signal with nearly original quality.

(4) **Media player:** Both the scrambled and descrambled audio should be playable with standard media player.

(5) **Audio duration:** The duration of the scrambled audio should not be differed from the original audio length.

In the past few years, a lot of researches have been conducted and several scrambling techniques have been proposed for minimizing the perceptibility of audio signals and for limiting the access to only authorized users.

Chen and Hu [16] proposed two effective transposition-based audio scrambling methods based on a secret key and the in-order traversal of a complete binary tree, respectively. Then, they were combined based on a new parameter to strengthen the security offered. However, there was no detailed evaluation on that combined scheme regarding how much security was strengthened, and time consumption and the perceptual quality yielded by those methods.

Li, Shang, and Zou [28] also presented a digital audio scrambling method based on Fibonacci transformation. The scrambling effect was good as the information of the audio was redistributed uniformly across the whole signal after just applying the transformation. Their algorithm could resist some attacks and the robustness of Fibonacci transformation was satisfactory. However, there was no discussion on audio quality.

Yan, Fu, and Kankanhalli [42] proposed a scrambling scheme for protecting the MP3 files. The basic idea was to apply the multiple rounds of XOR operations to Huffman codewords according to a pre-defined key table. Milosevic, Delic, and Senk also presented an audio scrambling algorithm that used Hadamard matrices. Both of the methods changed the position and values of the audio samples. Hence, they are very effective for reducing the perceptual quality of the music.

Poblete [29] introduced the possibility of real-time use of discrete wavelet transform with pure data analysis and re-synthesis of audio signals. It was confirmed that the wavelet transform could be a powerful and interesting tool for audio processing.

Zhou and Au [22] proposed two approaches of composing the keystream to be used for audio scrambling. The first one uses dynamic password generator and the second one uses pseudo-random number generator. It was said that the complexity of generating the keystream using dynamic password generator is higher than the pseudo-random number generator.

This thesis also presents an audio scrambling method that will be applied on the audio data encoded in WAV format. The proposed work is different from others in terms of low computational complexity, less time consumption in the key table generation, and direct control on the progressive audio quality.

The following section discusses the DWT as the proposed system was developed based on the effect of the DWT coefficients of an audio signal on the HAS.

2.5 Discrete Wavelet Transform

To understand the proposed system properly, this section discusses the DWT process and especially how the DWT affects the perceptibility of human ear.

The DWT is a transformation that can be used to analyze the temporal and spectral properties of non-stationary signals like audio. It was developed to overcome the shortcoming of the Short Time Fourier Transform (STFT) used to analyze non-stationary signals. While the STFT gives a constant resolution at all frequencies, the DWT uses the multi-resolution technique by which different frequencies are analyzed with different resolution.

The DWT has a huge number of applications in science, engineering, mathematics, and computer science [15]. Other applied fields that are making use of wavelets include human vision, signal and image processing, magnetic resonance imaging, acoustics, music, sub-band coding, speech discrimination, radar, astronomy, neurophysiology, optics, turbulence, earthquake-prediction, nuclear engineering, and pure mathematics applications such as solving partial differential equations [2].

There is a total of 16 different wavelet families as listed in Table 2.2 [43]. Among them, Daubechies wavelet family is deployed in this proposed system as it is the most popular among the wavelet families [11]. The names of the Daubechies family wavelets are written as *dbN***,** where *N* is the order, and *db* is the "surname" of the wavelet. The *db1*

wavelet is the same as Haar wavelet. In 1988, Daubechies constructed a family of easily implemented and easily invertible wavelet transforms that, in a sense, generalize the Harr transform. Like the Harr transform, the Daubechies wavelet is implemented as a succession of decompositions [19]. Moreover, wavelet transform using the Daubechies decomposition can provide a sparse representation for piecewise-linear signals.

Table 2.2: Wavelet family names [43]

No.	**Wavelet Family in Matlab**	**Wavelet Family Name**
1	'haar'	Haar wavelet
2	'db'	Daubechies wavelets
3	'sym'	Symlets
4	'coif'	Coiflets
5	'bior'	Biorthogonal wavelets
6	'rbio'	Reverse biorthogonal wavelets
7	'meyr'	Meyer wavelet
8	'dmey'	Discrete approximation of Meyer wavelet
9	'gaus'	Gaussian wavelets
10	'mexh'	Mexican hat wavelet
11	'morl'	Morlet wavelet
12	'cgau'	Complex Gaussian wavelets
13	'shan'	Shannon wavelets
14	'fbsp'	Frequency B-Spline wavelets
15	'cmor'	Complex Morlet wavelets
16	'fk'	Fejer-Korovkin wavelets

There are two main types in DWT: one-dimensional DWT (1D-DWT) used for audio analysis and two-dimensional DWT (2D-DWT) used for image analysis. Regarding the 1D-DWT used in this system, the analysis or decomposition process (DWT) and the synthesis or reconstruction process (IDWT) are shown in Figure 2.3.

In DWT, a discrete time-domain signal (sampled audio signal here in this proposed system) is analyzed by successive lowpass and highpass analysis filters [24]. Given an

audio signal of length N, it can be decomposed into $\log_2(N)$ levels at most. For example, an audio clip with 441,000 samples, it can be decomposed into 18 levels in maximum. At the first level, the analysis lowpass and highpass filters followed by downsampling processes produce the coarse approximations (*a1*) and the detail information (*d1*), respectively. The detail information corresponds to the high frequency components of an audio signal, while the approximations are the low frequency components. Downsampling by 2 doubles the frequency resolution as the uncertainty in frequency is reduced by half and halves the time resolution as the entire signal is now represented by only half of the number of samples. The second level repeats the same scheme on *a1* and produce *d2* and *a2*. This process is continued until the desired level is reached. For *i*-level decomposition, the DWT decomposes an audio signal into *i+1* layers of frequency coefficients $\{d_1, d_2, ..., d_i, a_i\}$. With this approach, time resolution becomes arbitrarily good at high frequencies and frequency resolution becomes arbitrarily good at low frequencies.

During the synthesis process IDWT, the original signal can be successfully reconstructed from the wavelet coefficients. The detail and approximation coefficients at every level are upsampled by 2 and passed through the highpass and lowpass synthesis filters, respectively, and then added. This process is continued through the same number of levels as in the decomposition process to obtain the original signal.

Usually, normal human ear is most sensitive in the low frequency range. Thus, among the wavelet layer coefficients, coarse approximations are more important for perceptual quality of the audio signals by human ear. In this proposed system, an audio signal is 4-level wavelet decomposed that yields a total of 5 wavelet layers with different frequency components. Then, the proposed audio scrambling method is applied on each layer by using different keys. As different wavelet layers have different significant effects on the HAS, modification introduced by the scrambling process on each layer will yield different quality audio files. This feature is really attractive for online music distribution.

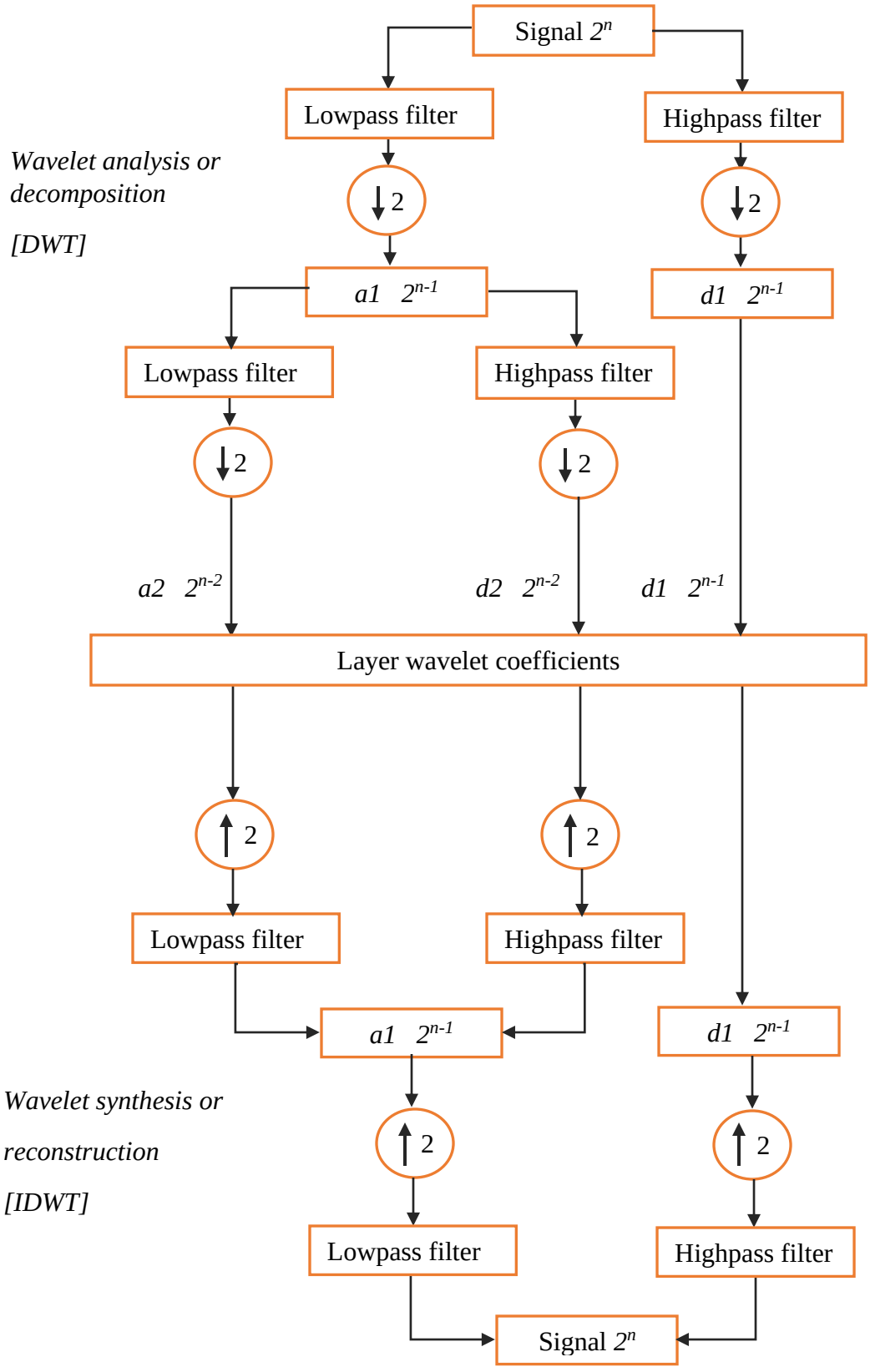


Fig. 2.3: Wavelet analysis and synthesis (DWT and IDWT) scheme

CHAPTER 3

THE PROPOSED SYSTEM

This chapter discusses the implementation of the proposed system in detail. It begins with the explanation of how to generate the keys used for scrambling and descrambling processes in this system. Then it discusses the implementation of each step of the proposed audio scrambling and descrambling methods in detail.

3.1 The Generalized Flow of the Proposed System

One of the main aims of the proposed system in this thesis is to provide different-quality audio files, i.e. teaser or high-fidelity music, which will be useful for online music stores. To achieve this aim, the proposed system makes use of the effect of audio wavelet layers on human auditory system. The generalized flow of the proposed system (scrambling process) is shown in Figure 3.1 in which raw PCM signal is first input to the DWT decomposition process. Then, the proposed scrambling method is applied on each wavelet layers by using different keys from a pre-generated key table. After the synthesis process (IDWT) with scrambled coefficients, quality degraded audio files are obtained.

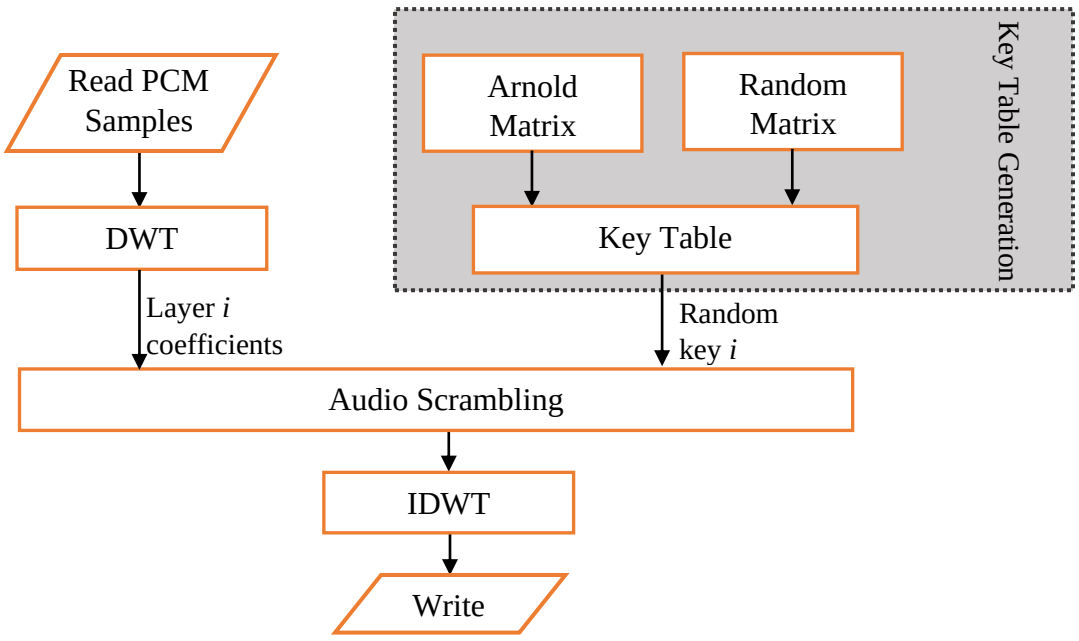


Fig. 3.1: The generalized process flow (scrambling process)

Figure 3.1 consists of four main processes: DWT (wavelet decomposition), key table generation, audio scrambling process, and IDWT (wavelet reconstruction).

As discussed in Chapter 2, the DWT can effectively be used to analyze the temporal and spectral properties of non-stationary signals like audio. When an audio signal is analyzed in DWT, it is first decomposed into detail and approximation coefficients by using an analysis filter. The approximation coefficients correspond to the low-frequency components of an audio signal, while the details are the high-frequency components. Normal human ear is usually most sensitive in the low-frequency range and thus, the approximations are more important for perceptibility. The experiments that will be explained in detail in the following chapter also confirm that there is drastic damage on the audio quality even for slight changes on the approximation coefficients. Based on this nature of the DWT, the proposed audio scrambling method is applied on detail coefficients to generate teaser music and applied on approximations to provide very low or zero-quality music. The IDWT is the reverse process of the DWT, and can refer Chapter 2 for more detail.

Another two main processes are the key table generation and audio scrambling process. The security of the proposed system depends on the key generation process; whereas, the progressive quality control and compatibility of the scrambled music with the media standard depends on the scrambling process. The following subsections discuss those processes in detail.

3.2 Key Table Generation

Security of the proposed system mainly depends on the secrecy, uniqueness, and randomness of the keys used in the scrambling and descrambling processes. As discussed above, the proposed system uses different keys for scrambling each wavelet layer. For an audio signal of length N, it can be decomposed into $\log_2(N)$ levels at most. For i-level decomposition, there are a total of i+1 layers and thus i+1 different keys are needed for scrambling. In this system, those keys are randomly chosen from a key table pre-generated based on the Arnold and Random matrices. Thus, the larger the size of the key table, the more randomness in selection of the layer keys and thus the stronger security the system achieves.

3.2.1 Arnold Matrix Generation

Arnold Matrix is mostly used to scramble the digital images and has many applications, especially in digital watermarking. In this system, Arnold matrix is used to transform the keys in the key table. Size of the Arnold matrix depends on the size of the key table. That is, if we plan to use an 8x8 key table, then we have to generate the Arnold matrix with the size of 8x8. As mentioned above, the larger the size of the key table then the stronger the security offered. In this proposed system, the key table size is chosen as 8x8 just for simplicity of implementation. Firstly, an 8x8 Arnold matrix is generated by applying the horizontal and vertical concatenation on the base Arnold matrix (*A*), defined in eq. 3.1.

$$A(p \times q) = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \tag{3.1}$$

where p x q is the size of the matrix. After the concatenation process, the resulting 8x8 Arnold matrix is shown below.

$$A_{(8\times 8)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 2 & 1 & 2 & 1 & 2 \end{bmatrix}$$

3.2.2 Random Matrix Generation

In this system, a matrix of random numbers is also used along with the Arnold matrix to generate the keys used in the scrambling and descrambling processes. This system uses the MATLAB software to generate those random numbers. When we create random numbers using software, the results are not random in a strict, mathematical sense. Similarly, the random numbers in MATLAB are also not unpredictable at all; however, they are generated by a deterministic algorithm designed to be sufficiently complicated so that its

output appear to be random and independent. The random numbers generated by MATLAB also pass the various statistical tests of randomness.

The size of the random matrix also depends on the key table size. If we plan to use a key table of size 8x8, then we have to generate the 8x8 random matrix. A random matrix (*R*) can be generated in MATLAB as follows.

$$R = randi(i\max, n), \tag{3.2}$$

where *imax* is the maximum random number to be generated, *n* is the matrix size. An example random matrix generated in MATLAB by using the command "*randi(120,8)"* and using seed number "3" are shown below.

$$R_{(8\times 8)} = \begin{bmatrix} 21 & 94 & 42 & 35 & 61 & 74 & 79 & 52 \\ 100 & 14 & 7 & 109 & 69 & 59 & 66 & 13 \\ 6 & 61 & 86 & 81 & 14 & 81 & 67 & 17 \\ 105 & 105 & 50 & 95 & 92 & 95 & 105 & 1 \\ 82 & 112 & 89 & 108 & 21 & 118 & 55 & 26 \\ 54 & 27 & 58 & 31 & 65 & 11 & 65 & 26 \\ 81 & 31 & 61 & 111 & 16 & 76 & 83 & 62 \\ 56 & 109 & 51 & 104 & 9 & 70 & 11 & 69 \end{bmatrix}$$

In this proposed system, scrambling keys are generated based on the random and Arnold matrices. Thus, we need to know the same random matrix for generating the keys to be used in descrambling. Otherwise, we can never recover the original audio quality even after descrambling.

In order to achieve that purpose, the proposed system carefully sets the seed number which is used to create the repeatable arrays of random numbers by saving and restoring the generator setting in MATLAB. If we use the same seed every time, it will yield the same sequence of random numbers. Without knowing the seed number used during scrambling, the user cannot generate the same random matrix and thus cannot derive the key needed to descramble the audio signal to recover its original quality. Please refer the following random matrix generated by using different seed number "5".

$$R_{(8\times8)} = \begin{matrix} 27 & 36 & 50 & 40 & 84 & 32 & 98 & 116 \\ 105 & 23 & 36 & 18 & 94 & 97 & 66 & 76 \\ 25 & 10 & 76 & 20 & 3 & 105 & 93 & 98 \\ 111 & 89 & 70 & 116 & 70 & 111 & 59 & 68 \\ 59 & 53 & 72 & 116 & 1 & 1 & 4 & 77 \\ 74 & 19 & 32 & 23 & 62 & 57 & 11 & 98 \\ 92 & 106 & 35 & 3 & 77 & 118 & 14 & 112 \\ 63 & 33 & 31 & 25 & 119 & 48 & 31 & 110 \end{matrix}$$

3.2.3 Key Generation Process

Finally, the key table (*K*) is generated by multiplying the Arnold matrix (*A*) and the random matrix (*R*) and then dividing by the key table size (*N*), as defined in eq. 3.3. The operation "*mod N*" means taking the remainder after division by *N*.

$$K = A \times R \bmod N. \tag{3.3}$$

The key table resulting from the Arnold matrix (section 3.2.1) and the first random matrix (section 3.2.2) is shown below.

$$AxR_{(8\times8)} = \begin{matrix} 505 & 553 & 444 & 674 & 347 & 584 & 531 & 266 \\ 820 & 808 & 610 & 1013 & 582 & 819 & 778 & 375 \\ 505 & 553 & 444 & 674 & 347 & 584 & 531 & 266 \\ 820 & 808 & 610 & 1013 & 582 & 819 & 778 & 375 \\ 505 & 553 & 444 & 674 & 347 & 584 & 531 & 266 \\ 820 & 808 & 610 & 1013 & 582 & 819 & 778 & 375 \\ 505 & 553 & 444 & 674 & 347 & 584 & 531 & 266 \\ 820 & 808 & 610 & 1013 & 582 & 819 & 778 & 375 \end{matrix}$$

$$K_{(8\times8)} = \begin{matrix} 57 & 41 & 60 & 34 & 27 & 8 & 19 & 10 \\ 52 & 40 & 34 & 53 & 6 & 51 & 10 & 55 \\ 57 & 41 & 60 & 34 & 27 & 8 & 19 & 10 \\ 52 & 40 & 34 & 53 & 6 & 51 & 10 & 55 \\ 57 & 41 & 60 & 34 & 27 & 8 & 19 & 10 \\ 52 & 40 & 34 & 53 & 6 & 51 & 10 & 55 \\ 57 & 41 & 60 & 34 & 27 & 8 & 19 & 10 \\ 52 & 40 & 34 & 53 & 6 & 51 & 10 & 55 \end{matrix}$$

To recap, the security offered by this proposed system is controlled by three steps: firstly, without knowing the seed number, it will be difficult to generate the same random numbers used in scrambling. Secondly, without knowing the key table size, we cannot generate the same key table. Thirdly, without knowing the random key used to scramble each wavelet layer, it is impossible to recover the full-quality sound.

In addition, as discussed in Chapter 2, there are various kinds of wavelet families available. Different wavelet family produces different wavelet layer of coefficients even for the same number of decomposition level. In this system, the Daubechies wavelet family is chosen as it is the most commonly used one among the wavelet families. However, the proposed system can be applied on any kind of wavelet family. Therefore, in order to enhance the security of the proposed system, the distributor can also choose the wavelet family in secrecy.

3.3 Audio Scrambling Method

After wavelet decomposition, the proposed audio scrambling method is applied on each wavelet layer by using different keys randomly chosen from the pre-generated key table. Flowchart of the proposed scrambling method is shown in Figure 3.2 and detailed procedure is as follows:

Step 1: Read the DWT coefficients of the selected layer.

Step 2: Choose the scrambling key from the pre-generated key table by randomly indexing the row and column number of the key table.

$$x = randi([1, p], 1), \tag{3.4}$$

$$y = randi([1, q], 1), \tag{3.5}$$

where x and y are the row and column index of the key table, and p x q is the size of the key table.

Step 3: Perform XOR the selected layer coefficients with the selected key.

$$C' = C \otimes K(x, y), \tag{3.6}$$

where C and C' are the coefficients before and after scrambling, $\otimes$ is the bit-wise XOR, and $K(x, y)$ is the random key selected. Based on the system requirements, the above steps must be repeated for all the layers needed to be scrambled.

Step 4: Perform IDWT to construct the scrambled audio. After IDWT, the scrambled audio file is saved in .wav format.

Conforming to the media standard is also one of the main aims of the proposed system. It was achieved in this system by using simple mathematical operation like XOR. The resulting scrambled audio clips can be easily playable by standard media players. Additionally, based on the system requirement, scrambled audio with desired quality level can be generated by choosing the wavelet layer to be scrambled.

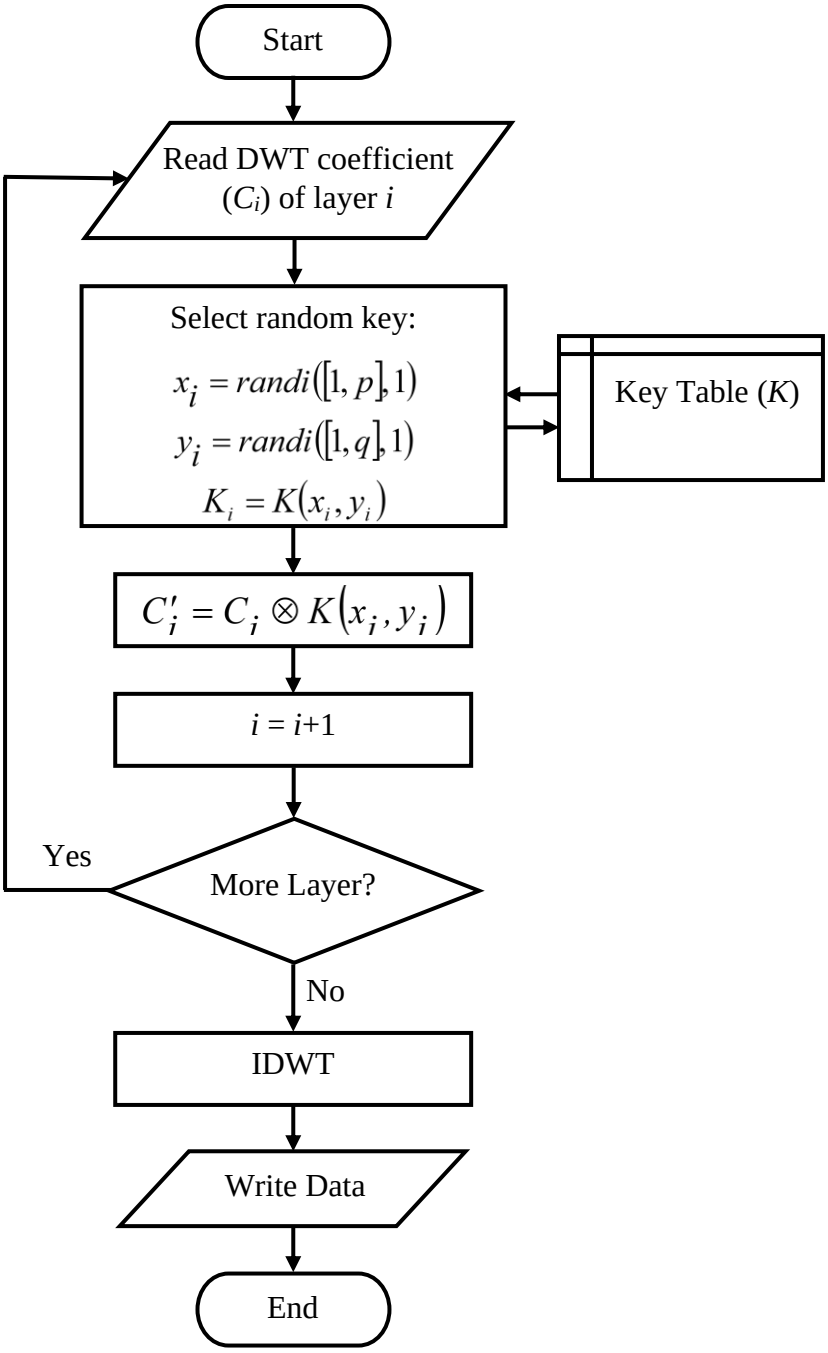


Fig. 3.2: Flowchart of the proposed scrambling method

3.4 Audio Descrambling Method

The generalized flow of the audio descrambling process is shown in Figure 3.3. It is exactly the same as the scrambling process. The scrambled audio is first decomposed into different wavelet layers. Note that the number of wavelet layers must be the same as the scrambling

process so that the original audio quality can be retained. Then, the proposed descrambling method is applied on each layer by using the same keys used during scrambling. Finally, the descrambled audio signal is reconstructed via IDWT.

Detailed process flow of the proposed descrambling method is exactly the same as the scrambling method given in Figure 3.2. To regain the original audio quality, the above procedure in Figure 3.2 must be applied again on the scrambled contents by using the same keys. Only if all the keys are correct, the audio quality can be recovered to its original. In addition, the output quality of the descrambled audio also depends on the number of descrambled layers. If *n* layers of the audio were scrambled, all *n* layers must be descrambled to retain full quality. For example, if the audio was scrambled 7 times using 7 different keys, the user must have all the 7 keys and perform the descrambling process 7 times before he/she can perfectly restore the original audio. Otherwise, it is impossible to recover the full-quality audio.

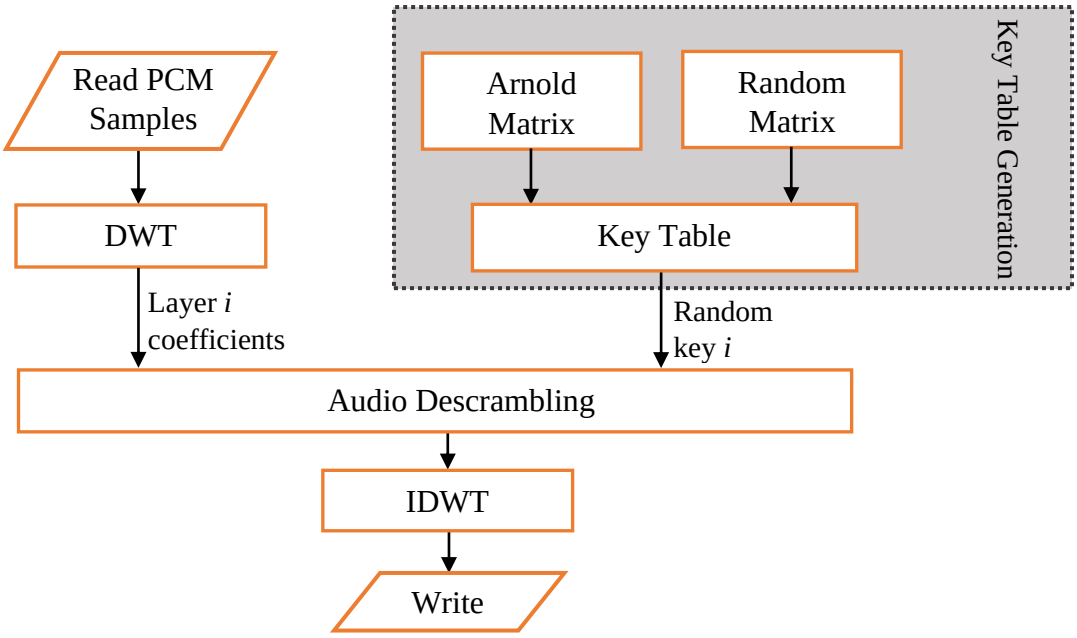


Fig. 3.3: The generalized process flow (descrambling process)

CHAPTER 4

RESULTS AND DISCUSSION

This chapter presents the analytic results of the performance of the proposed system. It begins with the evaluation of the progressive audio quality after scrambling and descrambling processes by means of both subjective (mean opinion score, MOS) and objective (signal-to-noise ratio, SNR) measures. The effect of scrambling and descrambling on audio signals are also visualized in waveforms. Moreover, the time consumption and effect of the proposed methods on audio file sizes are also evaluated.

4.1 Experimental Setup

The proposed system is implemented in MATLAB and this section presents the evaluation results of its performance on 25 different audio files. Each file is encoded in 16 bits per sample, 44.1 kHz sampling rate, and .wav audio format. Table 4.1 describes the music pieces used in the experiments, which are grouped based on genre.

For all of the following experiments, each audio clip is decomposed into five layers of wavelet coefficients {$d1$, $d2$, …, $d4$, $a4$}, i.e. four wavelet levels. Each layer is then scrambled by using different keys. Evaluations are then done in terms of (i) the effect of scrambled wavelet layer on audio quality, (ii) progressive audio quality, (iii) execution time, and (iv) the effect of scrambling on file size.

Table 4.1: Music pieces for experiments

Song	Genre	Avg. Length (sec)	Avg. Size (MB)
S1-5	Classical	23	1.62
S6-10	Pop	23	1.95
S11-15	EDM	22.8	1.95
S16-20	Rock	26.2	2.22
S20-25	Jazz	25.4	2.24

4.1.1 Signal-to-Noise Ratio (SNR)

This proposed system provides different-quality audio files based on the wavelet layer scrambled. It is confirmed here by evaluating the audio quality after scrambling each

wavelet layer by means of the SNR. The SNR is the objective measurement used to estimate the quality degradation between the original and the scrambled audio signals. It is defined as the ratio of the signal power to the noise power, as stated in eq. 4.1. As per the International Federation of Phonographic Industry (IFPI), an audio clip with SNR ≥ 20 dB is considered to be good quality [18]. Thus, the higher the SNR means the better the audio quality.

$$SNR(dB) = 10log\left(\frac{P_{signal}}{P_{noise}}\right), \tag{4.1}$$

where P_{signal} indicates the audio signal power and P_{noise} is the noise power [35]. SNR is usually expressed in decibels (dB).

4.1.2 Mean Opinion Score (MOS)

The MOS is a numerical measure of the human-judged overall quality of an experience. It is a commonly used measure for video, audio, and audiovisual quality evaluation, but not restricted to those modalities [27]. The MOS is expressed as a single rational number, typically in the range of 1 to 5, where 5 is the highest and 1 is the lowest perceived quality, as shown in Table 4.2. In this system, the MOS measures will be used to check the quality difference between the original and scrambled/descrambled audio clips.

Table 4.2: MOS rating

MOS	Audio Quality	Impairment Description
5	Excellent	Imperceptible
4	Good	Perceptible but not annoying
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying

4.2 Effects of Scrambled Wavelet Layer on Audio Quality

Table 4.3 shows the average SNR values after scrambling each wavelet layer. For this experiment, only one layer of wavelet coefficients is scrambled at a time in order to evaluate the importance of each layer for perceptible audio quality. The results confirm

that scrambling on different layer results different SNR, which means different audio quality. According to the IFPI, SNR ≥ 20 dB means good audio quality. Thus, the results show that scrambling on *d1*, *d2*, and *d3* layers produces good and acceptable quality music, whereas scrambling on *d4* and *a4* yields poor and bad-quality music.

Table 4.3 also shows that the SNR values are getting dropped from *d1* to *a4*. It is because *d1* is the highest in frequency among the wavelet layers and *a4* is the lowest in frequency. As previously stated, human ear is more sensitive in the low frequency range. Thus, scrambling on *a4* yields the lowest audio quality.

Considering the proposed system is to be used in music distribution, the number of layers to be scrambled can also be chosen based on the system requirement. If the distributor requires high-level access control, all layers should be scrambled. Otherwise, scrambling only the approximation layer is sufficient for degrading quality. If the distributor wants to share a new song as preview, scrambling only the detail layers is sufficient.

Table 4.3: Average SNRs after scrambling each layer

Song	**Genre**	**Scrambled Layer**				
		d1	***d2***	***d3***	***d4***	***a4***
S1-5	Classical	28.84	24.15	20.41	17.53	7.80
S6-10	Pop	26.55	22.39	19.78	17.70	7.77
S11-15	EDM	26.38	23.35	21.69	20.09	7.82
S16-20	Rock	24.02	19.80	16.88	15.31	8.70
S21-25	Jazz	31.36	26.97	23.29	19.27	7.29
Average		**27.43**	**23.33**	**20.41**	**17.98**	**7.88**

4.3 Evaluation on Progressive Audio Quality

This section proves that the perceptible audio quality also depends on the number of wavelet layers scrambled. The more the layers are scrambled, the worse the resulting audio quality is. Both the SNR and MOS matrices are used to evaluate the progressive audio quality achieved by the proposed system.

4.3.1 Results of Objective Evaluation

Table 4.4 and 4.5 show the average SNR values after scrambling and descrambling layer-by-layer. It is seen from Table 4.4 that the SNR values are decreasing when the number of scrambled layers are increased. Similarly, if we descramble more layers, the recovered audio quality is getting better. It verifies that the proposed system achieves the progressive audio quality.

The last column of Table 4.5 also shows that the *a4* layer plays a vital role in audio reconstruction. After descrambling all layers including *a4*, the average SNR values go high up to approx. 242 dB. It shows that the proposed system can perfectly recover the very high audio quality after descrambling. Unfortunately, it cannot recover the audio to its fullest quality because the proposed system is based on the wavelet transform. The wavelet reconstruction (IDWT) can reconstruct the audio signal to be perceptually equivalent as the original file but not statistically equivalent. If the system could recover statistically equivalent audio file, the SNR will be infinity. Figure 4.1 and 4.2 visualize how the SNR values are decreasing when more layers are scrambled and how they are increasing when more layers are descrambled, respectively.

Table 4.4: Average SNR results after scrambling additional layers

Song	**Genre**	**Scrambled Layer**				
		d1	***d1, d2***	***d1 to d3***	***d1 to d4***	***All***
S1-5	Classical	28.84	22.87	18.44	14.89	6.87
S6-10	Pop	26.55	20.98	17.30	14.46	6.90
S11-15	EDM	26.38	21.47	18.49	16.11	7.10
S16-20	Rock	24.02	18.39	14.55	11.87	6.82
S21-25	Jazz	31.36	25.61	21.24	17.11	6.83
Average		**27.43**	**21.86**	**18.00**	**14.89**	**6.90**

Table 4.5: Average SNR results after descrambling additional layers

Song	Genre	Descrambled Layer				
		d1	***d1, d2***	***d1 to d3***	***d1 to d4***	***All***
S1-5	Classical	6.91	7.01	7.28	7.80	241.73
S6-10	Pop	6.95	7.08	7.33	7.77	242.03
S11-15	EDM	7.16	7.27	7.44	7.71	241.42
S16-20	Rock	6.92	7.18	7.75	8.70	242.31
S21-25	Jazz	6.81	6.93	7.04	7.35	242.08
Average		**6.95**	**7.10**	**7.37**	**7.87**	**241.91**

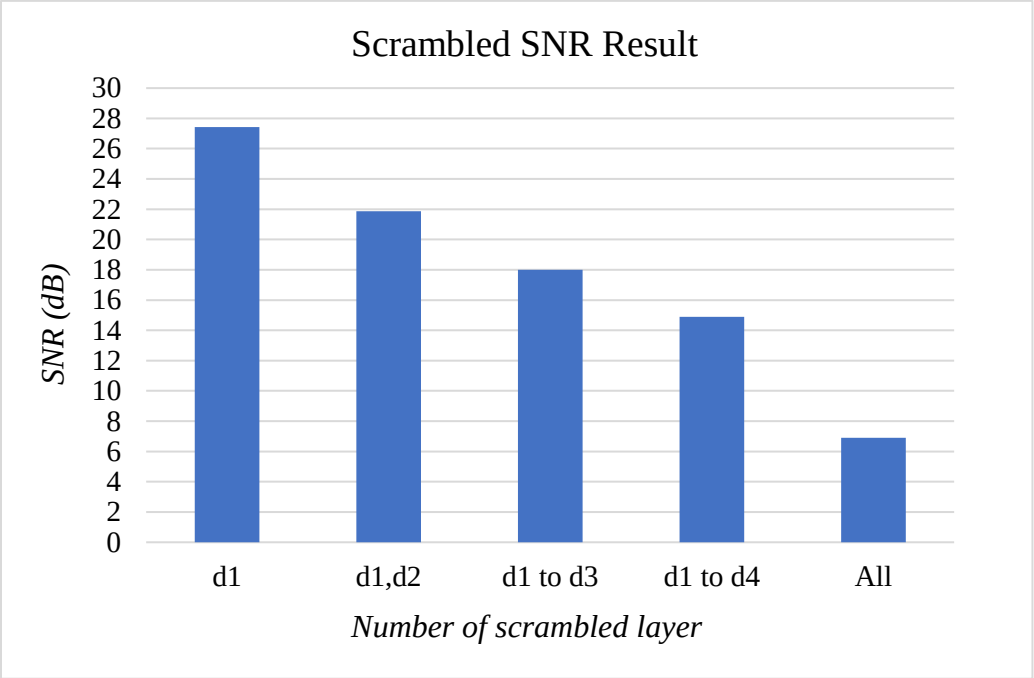


Fig. 4.1: SNR decreases based on the number of scrambled layers

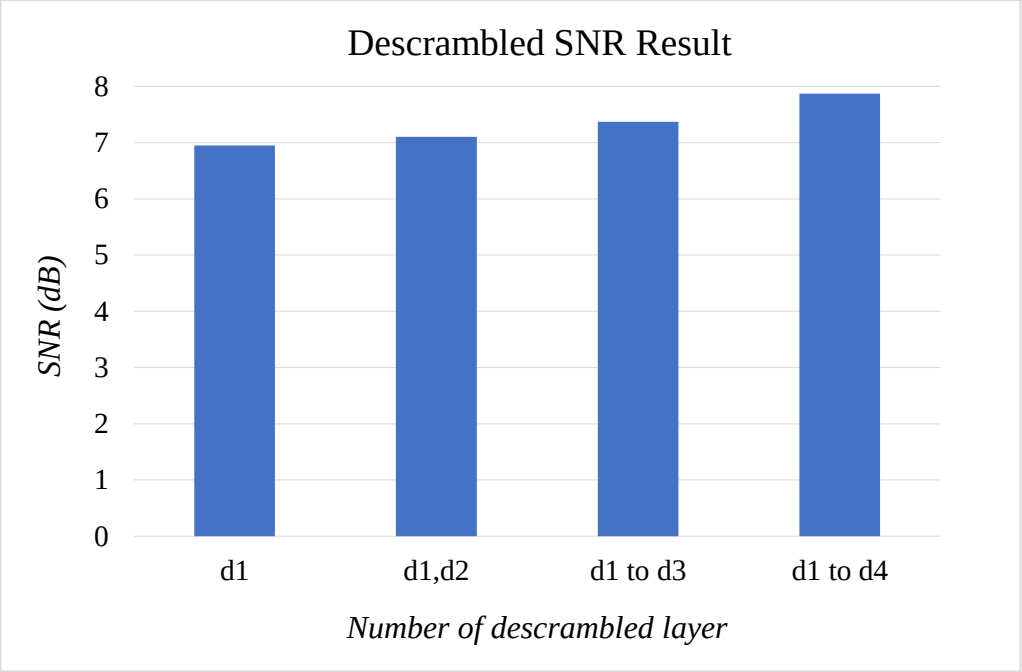


Fig. 4.2: SNR increases based on the number of descrambled layers

4.3.2 Results of Subjective Evaluation

Table 4.6 and 4.7 show the subjective evaluation results, i.e. MOS, judged by ten humans for the scrambled and descrambled audio clips, respectively. The results show that the listeners are able to perceive the gradual decrease in audio quality as the number of scrambled layer increases. Likewise, most of the evaluators are able to detect the quality improvement after descrambling more layers. For this kind of experiment, please note that the quality of the devices such as headphones or speakers used during evaluation test can also affect the evaluation result.

From the results in Table 4.6, it can be seen that the MOS ratings are getting lower while scrambling layer after layer, which means that the audio quality is getting dropped. As for descrambling, the MOS scores in Table 4.7 are getting higher when more layers are descrambled.

Table 4.6: Average MOS results after scrambling additional layers

Song	Genre	Scrambled Layer				
		d1	***d1, d2***	***d1 to d3***	***d1 to d4***	***All***
S1-5	Classical	3.60	3.00	1.90	1.64	1.00
S6-10	Pop	3.52	2.84	1.92	1.44	1.00
S11-15	EDM	3.44	3.00	2.22	1.56	1.00
S16-20	Rock	3.52	3.02	1.94	1.42	1.00
S21-25	Jazz	3.88	3.12	2.34	1.54	1.00
Average		3.59	2.99	2.06	1.52	1.00

Table 4.7: Average MOS results after descrambling additional layers

Song	Genre	Descrambled Layer				
		a4	***a4, d4***	***a4, d4, d3***	***a4, d4 to d2***	***All***
S1-5	Classical	1.46	1.98	3.08	3.94	5.00
S6-10	Pop	1.64	1.96	2.92	3.84	4.98
S11-15	EDM	1.76	2.12	2.92	3.66	4.86
S16-20	Rock	1.70	2.02	2.96	3.80	4.78
S21-25	Jazz	1.76	2.30	3.34	3.94	4.92
Average		1.66	2.08	3.04	3.84	4.91

Thus, the results mentioned above prove that the proposed system achieves progressive scrambling effect. Figure 4.3 and 4.4 visualize the MOS scores for scrambling and descrambling processes, respectively. As in SNR rating, the MOS scores also confirm that the proposed system achieves the progressive audio quality.

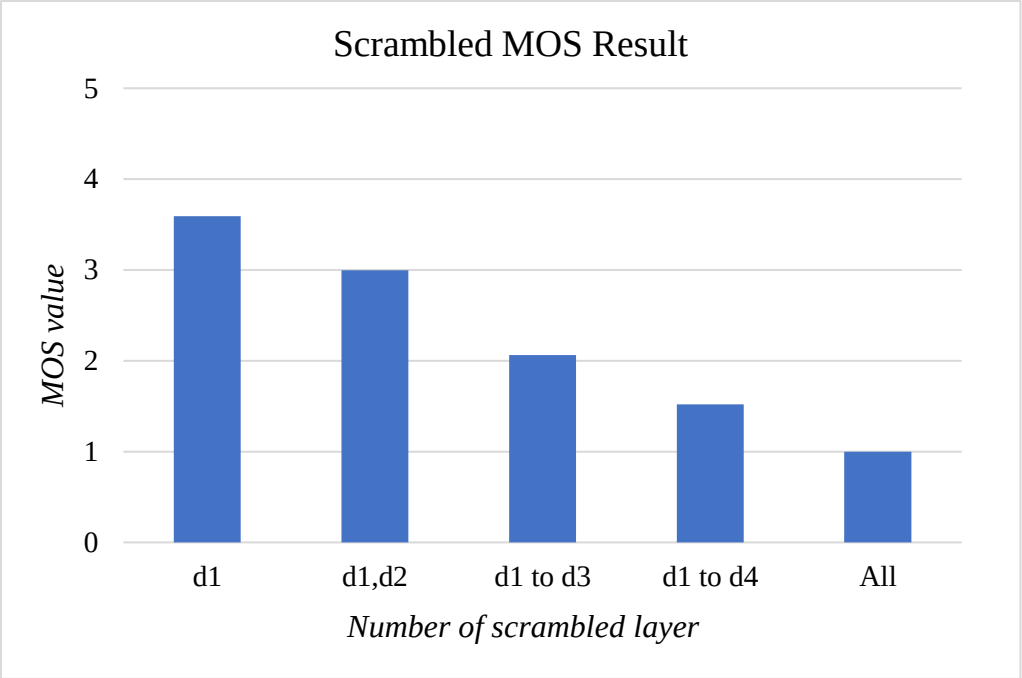


Fig. 4.3: MOS decreases based on the number of scrambled layers

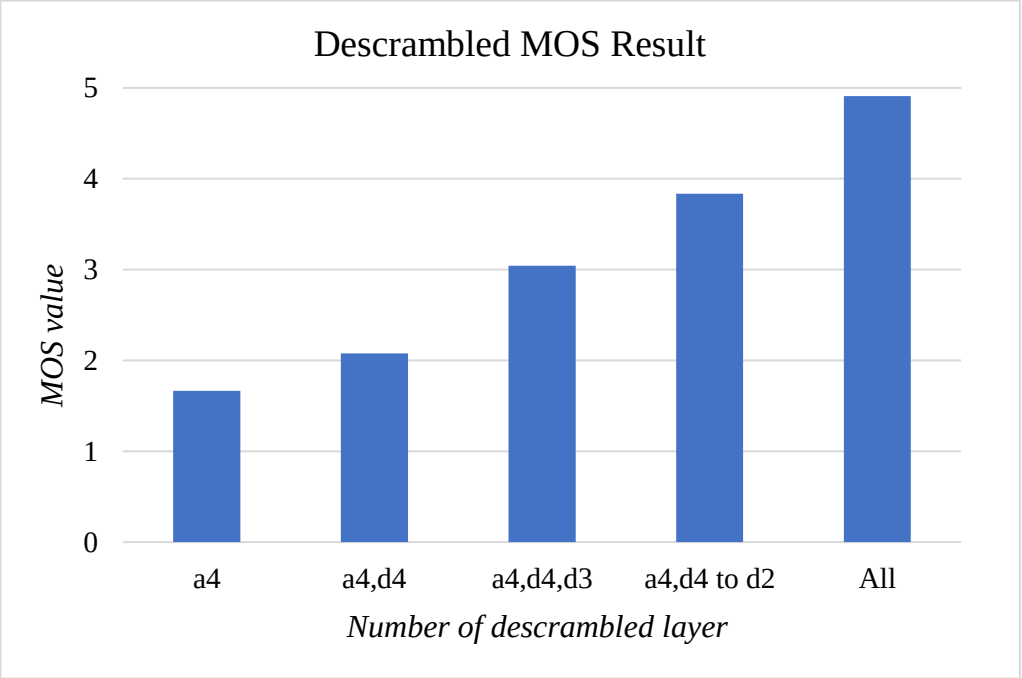
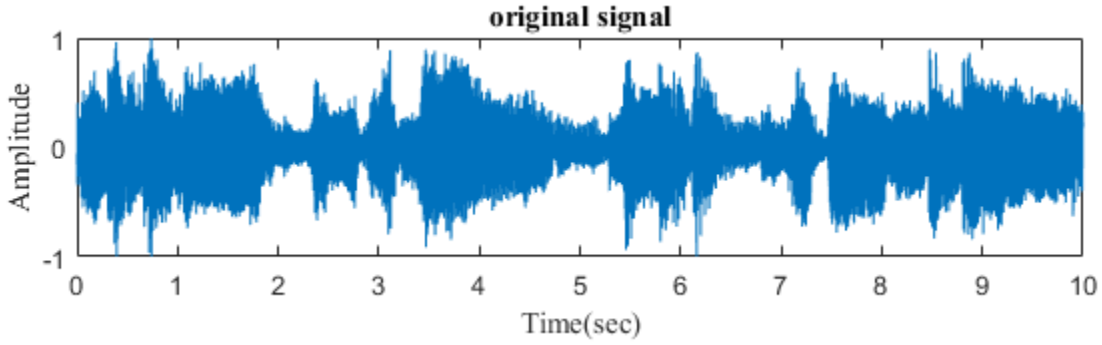


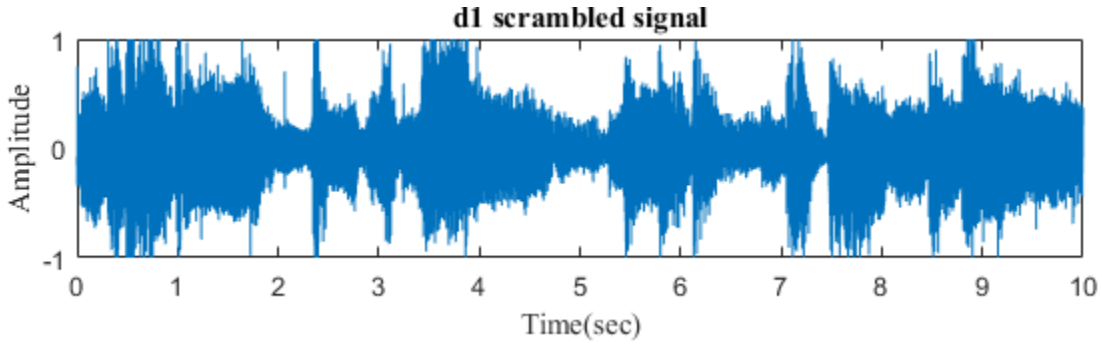
Fig. 4.4: MOS increases based on the number of descrambled layers

4.3.3 Waveform Visualization

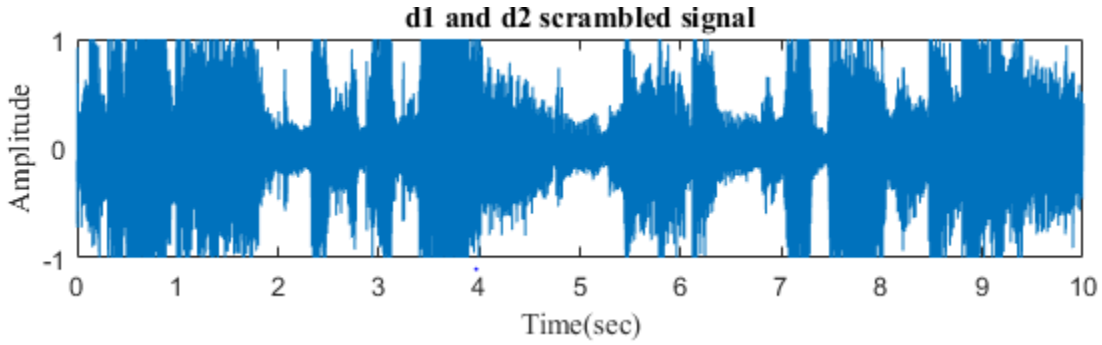
Figure 4.5 depicts the waveforms of an audio clip with different layer scrambled. Figure 4.5 (a) is the waveform of the original WAVE audio clip and Figure 4.5 (b) to (f) show the waveforms after scrambling the *d1* layer, *d1* + *d2*, *d1* to *d3*, *d1* to *d4*, and all layers, respectively. We can see from the figures that the structure and envelopes of the scrambled waveforms are getting more and more different than the envelope of the original signal. It means that the audio quality is also degrading after scrambling layer after layer.



(a)



(b)



(c)

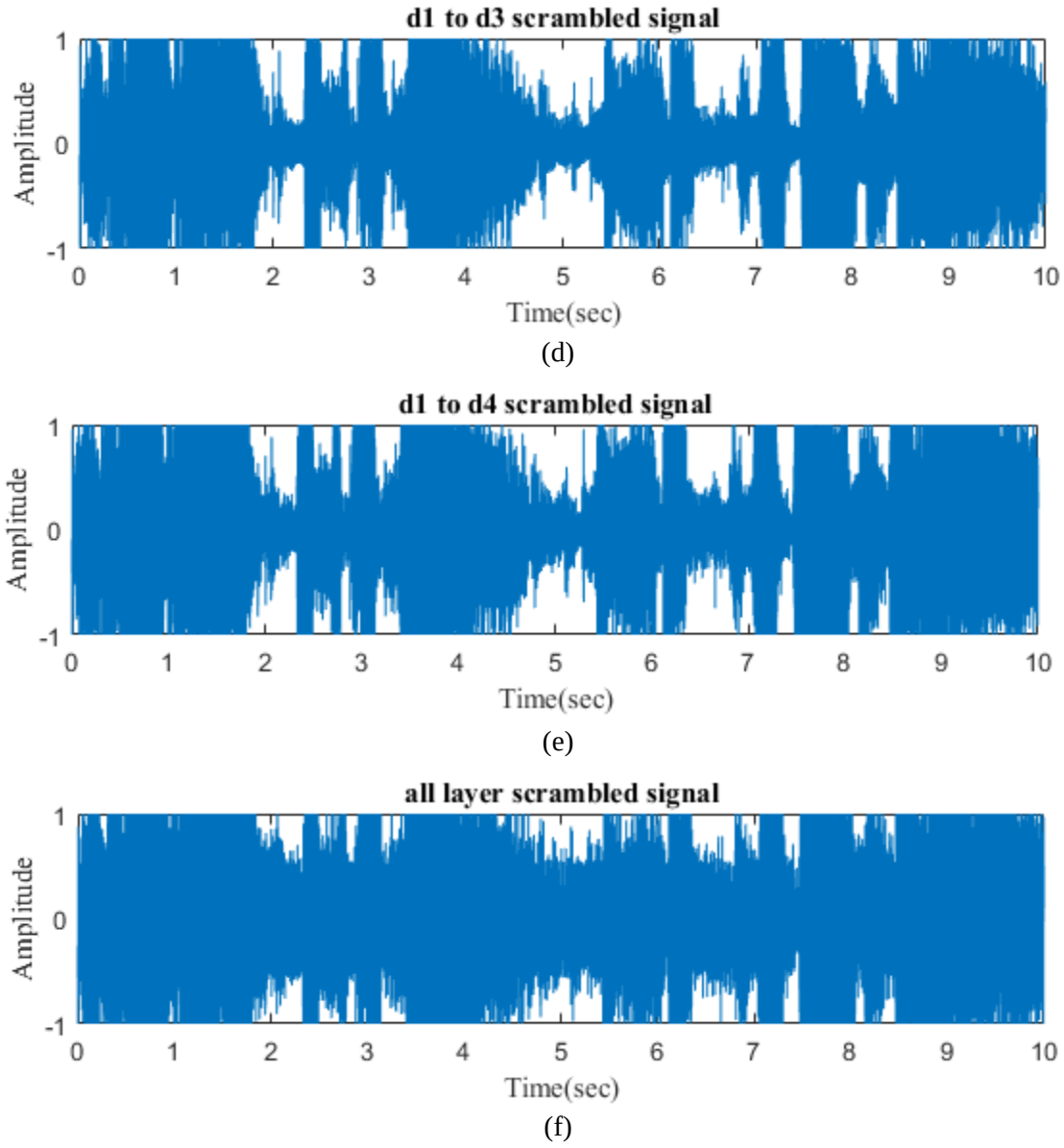
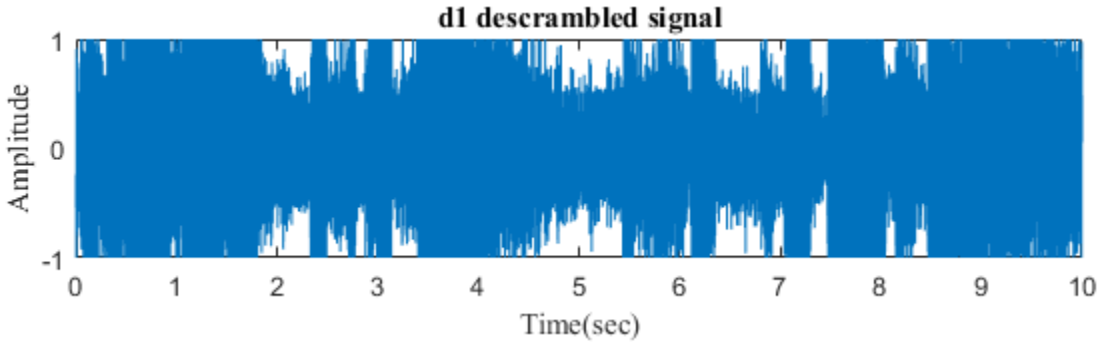
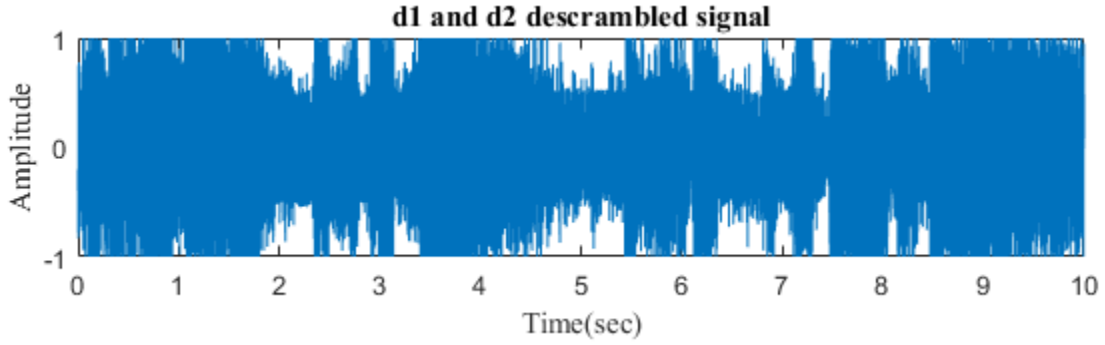


Fig. 4.5: Waveforms after scrambling (layer-wise)

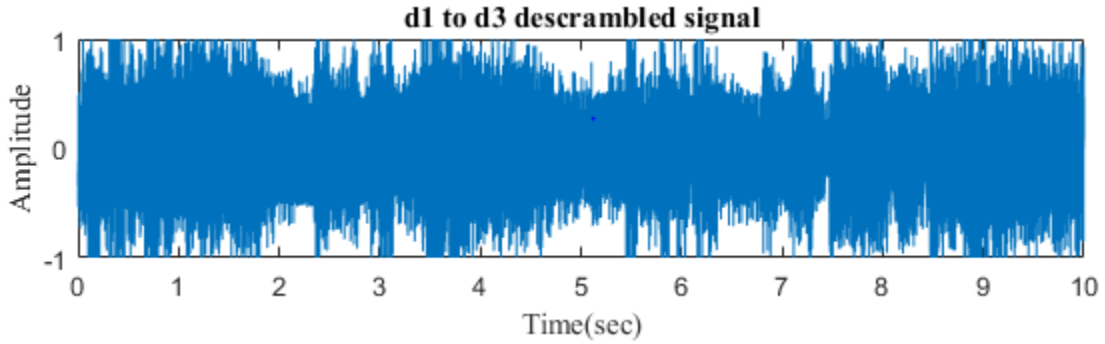
Figure 4.6 depicts the waveform visualization of an audio clip after descrambling layer after layer. Figure 4.6 (a) to (e) show the resulting waveforms after descrambling *d1*, *d1* and *d2*, *d1* to *d3*, *d1* to *d4*, and all layers, respectively. Figure 4.6 (f) shows the waveform of the original audio signal. By comparing the waveforms in Figure 6, it can be clearly seen that the proposed system can gradually reconstruct the audio signal to its original. It was possible to retain the original full-quality audio because the same keys as in scrambling were used for descrambling. Even if only one of the keys was wrong, it would not be possible to recover the original audio quality.


d1 descrambled signal
Amplitude
Time(sec)

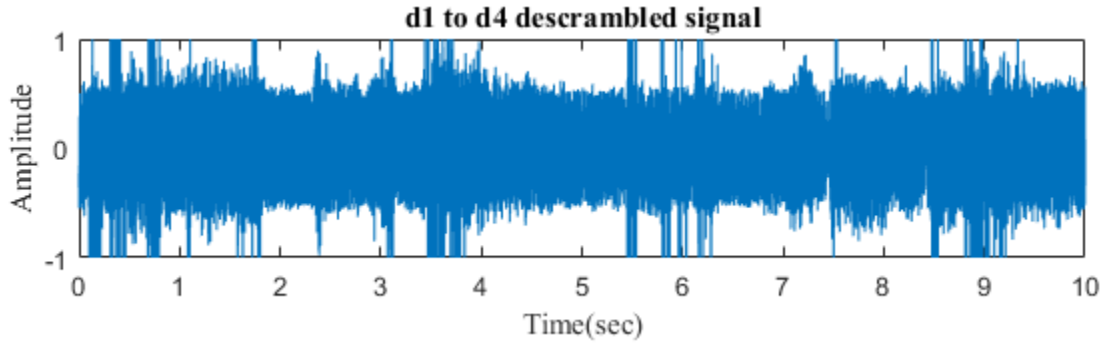
(a)


d1 and d2 descrambled signal
Amplitude
Time(sec)

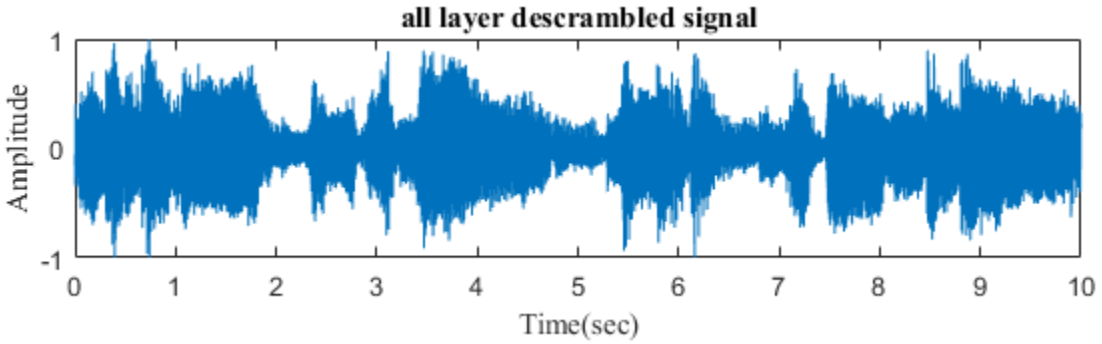
(b)


d1 to d3 descrambled signal
Amplitude
Time(sec)

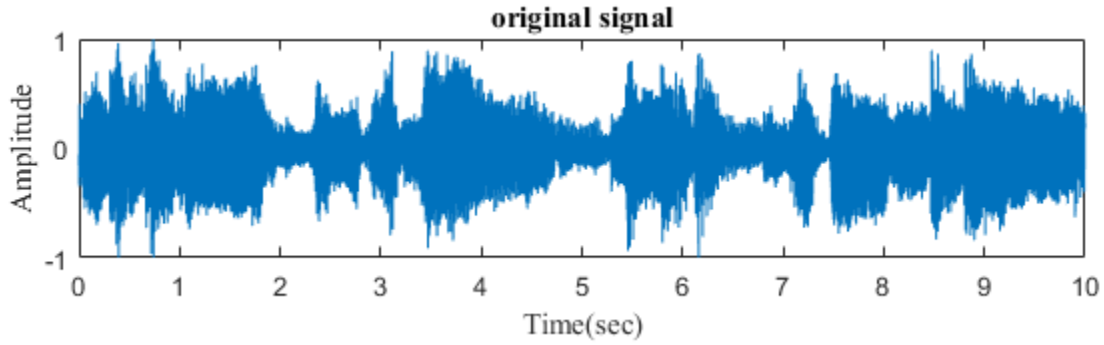
(c)



(d)



(e)



(f)

Fig. 4.6: Waveforms after descrambling (layer-wise)

To recap, by applying the proposed scrambling method in the wavelet domain that gives the progressive representation of a signal, an audio signal with desired quality level can easily be generated. Scrambling on different wavelet layers degrades the audio quality differently and thus ensures the progressive scrambling effect. To get the progressive representation of a signal, the other methods like the FFT filter bank can also be used instead of the wavelet transform; however, the wavelet representation is much more efficient for analysis and signal reconstruction because of its multi-resolution property.

Moreover, security offered by the proposed system can also be more strengthened with the control of unknown wavelet parameters. Without knowing the same wavelet family and wavelet decomposition levels used during scrambling, anyone can never recover full-quality audio.

4.4 The Effect on Execution Time and File Size

As per the experimental results shown in Table 4.8, it can be seen that the original and scrambled/descrambled audio clips are exactly the same in file size, no size blow-up. As for the average execution time, for a 24.1 sec long audio clip, the proposed system takes only 0.64 sec for scrambling and 0.30 sec for descrambling. From these results, it can be seen that the proposed system is very low in computational complexity and suitable to be used for real-time systems like online music stores.

Table 4.8: Results of average execution time and file size

Song	Duration (sec)/Size (MB)	Scrambled		Descrambled	
		Duration (sec)	Size (MB)	Duration (sec)	Size (MB)
S1-5	23/1.62	0.82	1.62	0.29	1.62
S6-10	23/1.95	1.00	1.95	0.29	1.95
S11-15	22.8/1.95	0.49	1.95	0.29	1.95
S16-20	26.2/2.22	0.47	2.22	0.32	2.22
S21-25	25.4/2.24	0.43	2.24	0.32	2.24
Average	**24.1/2.00**	**0.64**	**2.00**	**0.30**	**2.00**

4.5 The Proposed Application Scenario

The proposed system can be applied for online digital music services such as JOOX application that is currently popular among young consumers in Myanmar. The intended scenario is as follows. An audio clip is decomposed into 4-level-wavelet and all layers are scrambled using different keys. That file is used for secure music distribution to subscribers. Based on the subscription type, users can get 2, 4, or 5 keys for descrambling, as depicted in Table 4.9. For only 2 keys, the application will allow users to descramble *a4*

and *d4* layers and enjoy low-quality music. For VIP subscribers, they can get all 5 keys to recover high-fidelity music.

Table 4.9: The proposed application scenario

Quality Level	**Low**	**Medium**	**High**
Layer to be descrambled	*a4 & d4*	*a4, d4, d3 & d2*	*a4, d4 to d1*
Number of keys required	2	4	5

CHAPTER 5

CONCLUSION

Music industry has already been moving into online space successfully these days. Due to the improvements in communication technologies and easy file sharing services, the profit of this industry largely depends on the effective control of unauthorized access to music. When a new song comes out and if the music distributor uploads the whole song as teaser, there are potentially high risk of illegal downloads. Even for sharing the chorus as sample, illegal downloading can still occur for using as ring tunes.

This thesis has presented an effective music distribution system for online digital music industry. The main aim is to develop a system that can generate different low/medium and high-quality music files that can be used as teaser and secure music sharing with high-level access control, respectively. The proposed system was developed based on the DWT-based low-complexity audio scrambling method. As scrambling on different wavelet coefficients has different effects on perceptibility, experimental results like SNRs and MOS ratings showed that the proposed system can effectively generate different quality audio files based on the wavelet layer scrambled. Low- or medium-quality files can serve as teasers for potential buyers to taste the songs, whereas severely-degraded files can provide high-level access control for music distribution. Without the knowledge of scrambling keys, anyone can never recover the original song quality. As for valid users who know the keys, the proposed method can perfectly recover the original audio quality and also works well for all music genres.

Experimental results also showed that the proposed system has very low computational complexity (i.e. fast execution time). In addition, the proposed scrambling method can generate the scrambled files to be exactly the same in file size as the original files and thus it is appropriate for real-time applications.

5.1 Further Extension

The main aim of the proposed system is to be applied in real-world online music distribution systems. The intended application scenario is like JOOX music application where different-quality music files are shared to subscribers based on the subscription type.

As of now, we only implemented the proposed system on a standalone computer. In order to use in real-world applications, it should be implemented as a mobile application or a web service.

In addition, future research can also be carried on regarding the security aspect of the proposed system. As previously seen in Chapter 3, the Arnold matrix used for key generation consists of some duplicated values and may degrade the randomness of the keys. Thus, more secure key generation algorithms and secret keys sharing between music distributors and users should also be considered.

REFERENCES

[1] A. Srinivasan and P. A. Selvan, “A review of analog audio scrambling methods for residual intelligibility,” Innovative Systems Design and Engineering, vol. 3, no. 7, 2012.

[2] Amara Graps, “An introduction to wavelets,” Institute of Electrical and Electronics Engineers, IEEE Computer Society, vol. 2, no. 2, 1995.

[3] “Advanced encryption standard (AES)”, available online at: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

[4] “Analog-to-Digital converter”, available online at: https://en.wikipedia.org/wiki/Analog-to-digital_converter

[5] “Audio codecs: what they are and why they matter”, available online at: https://www.tomsguide.com/us/what-are-audio-codecs,review-4469.html

[6] “Audio file format”, available online at: https://en.wikipedia.org/wiki/Audio_file_format

[7] “Audio formats and file types”, available online at: https://soundbridge.io/audio-formats-file-types/

[8] B. Gadanayak and C. Pradhan, “Selective encryption of MP3 compression,” International Conference on Information Systems and Technology, 2011.

[9] D. Tabibzada, “How does illegally downloading music impact the music industry?,” Nov, 2015.

[10] “Data encryption standard”, available online at: https://www.sciencedirect.com/topics/engineering/data-encryption-standard

[11] “Daubechies wavelet”, available online at: https://en.wikipedia.org/wiki/Daubechies_wavelet

[12] “Difference between private key and public key", available online at: https://www.geeksforgeeks.org/difference-between-private-key-and-public-key/

[13] “Digital music-statistics & facts”, available online at: https://www.statista.com/topics/1386/digital-music/

[14] “Digital signal processor”, available online at: https://en.wikipedia.org/wiki/Digital_signal_processor

[15] “Discrete wavelet transform”, available online at: https://en.wikipedia.org/wiki/Discrete_wavelet_transform

[16] G. Chen and Q. Hu, “An audio scrambling method based on combination strategy,” IEEE International Conference on Computer Science and Information Technology, vol. 5, pp. 62-66, July, 2010.

[17] G. Dhanya and J. Jayakumari, “Permutation based speech scrambling for next generation mobile communication,” International Journal of Engineering and Technology, vol. 8, no. 2, pp. 707-713, Apr-May, 2016.

[18] H. Yi and C. L. Philipos, “Evaluation of objective measures for speech enhancement,” Interspeech2006, pp. 1447-1450, Sept, 2006.

[19] Ivan W. Selesnick, “Wavelet transforms- a quick study,” Polytechnic University Brooklyn, NY, Sept, 2007.

[20] “IFPI digital music report 2013: global recorded music revenues climb for first time since 1999”, available online at: https://www.billboard.com/articles/business/1549915/ifpi-digital-music-report-2013-global-recorded-music

[21] “IFPI global music report 2019”, available online at: https://www.ifpi.org/news/IFPI-GLOBAL-MUSIC-REPORT-2019

[22] J. Zhou and O. C. Au, “Security and efficiency analysis of progressive audio scrambling in compressed domain,” Innovation and Technology Commission of the Hong Kong Special Administrative Region, China, pp. 1802-1805, 2010.

[23] “Joox”, available online at: https://en.wikipedia.org/wiki/JOOX

[24] M. Vetterli and J. Kovacevic, Wavelets and subband coding, Prentice Hall PTR, Englewood Cliffs, New Jersey, 1995.

[25] Mansi and Raman Chawla, “An audio multiple shuffle encryption algorithm,” International Journal of Engineering and Computer Science, Haryana, India, vol. 4, pp. 14098-14104, Sep, 2015.

[26] Microsoft Corporation, “New multimedia data types and data techniques,” April 1994.

[27] "Mean opinion score", available online at: https://en.wikipedia.org/wiki/Mean_opinion_score

[28] N. Li, Y. H. Shang, and J. C. Zou, “An audio scrambling method based on Fibonacci transformation,” Journal of North China University of Technology, vol. 16(3), pp. 8-12, 2004.

[29] R. D. Poblete, “Manipulation of audio in the wavelet domain processing a wavelet stream using PD,” Institute of Electronic Music, 2006.

[30] Raghunandhan K R, Radhakrishna Dodmane, Sudeepa K B, and Ganesh Aithal, “Efficient audio encryption algorithm for online applications using transposition and multiplicative non-binary system,” International Journal of Engineering research and Technology, vol. 2, issue 6, June, 2013.

[31] “RC4”, available online at: https://en.wikipedia.org/wiki/RC4

[32] “RSA algorithm (Rivest-Shamir-Adleman)”, available online at: https://searchsecurity.techtarget.com/definition/RSA

[33] Sheetal Sharma, Lucknesh Kumar, and Himanshu Sharma, “Encryption of an audio file on lower frequency band for secure communication,” International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, issue 7, July, 2013.

[34] Shine P. James, Sudhish N. George, and P.P. Deepthi, “Secure selective encryption of compressed audio,” International Conference on Microelectronics, Communication and Renewable Energy, IEEE, 2013.

[35] “Signal-to-Noise Ratio (SNR)”, available online at: http://www.onmyphd.com/?p=snr.signal.noise.ratio

[36] “Spotify”, available online at: https://en.wikipedia.org/wiki/Spotify

[37] “The effects of illegal downloading on the muscic industry”, available online at: https://www.marshallmusic.co.za/2017/04/05/effects-illegal-downloading-music-industry/

[38] “The music industry in an age of digital distribution”, available online at: https://www.bbvaopenmind.com/en/articles/the-music-industry-in-an-age-of-digital-distribution/

[39] “The scientist and engineer's guide to digital signal processing”, available online at: https://www.dspguide.com/ch22/3.htm

[40] “The smart way to buy Myanmar music”, available online at: http://www.myanmarmusicstore.com/Default.aspx

[41] Vishakha B. Pawar, Pritish A. Tijare, and Swapnil N. Sawalkar, “A review paper on audio encryption,” International Journal of Research in Advent Technology, vol. 2, no. 12, Dec, 2014.

[42] W. Q. Yan, W. G. Fu, and M. S. Kankanhalli, “Progressive audio scrambling in compressed domain,” IEEE Transactions On Multimedia, Mar, 2008.

[43] “Wavelet families”, available online at: https://www.mathworks.com/help/wavelet/ug/wavelet-families-additional-discussion.html

[44] “What is digital signal processing (DSP)? And what does it mean for music?”, available online at: https://www.theguitarjournal.com/digital-signal-processing/

PUBLICATION

[1] Su Latt Sandi and Twe Ta Oo, “Effective Music Distribution System for Online Music Industry,” University of Computer Studies, Yangon, Myanmar, 2020.