International Journal of Multimedia and Ubiquitous Engineering
Vol.9, No.5 (2014), pp.241-250
http://dx.doi.org/10.14257/ijmue.2014.9.5.24

Raw Data Recovery from Pulse Code Modulation Pieces in the BitTorrent Environment

Jihah Nah[1] and Jongweon Kim[2]

[1]Copyright Protection Research Institute, Sangmyung University, Korea
[2]Dept. of Intellectual Property, Sangmyung University, Korea
[1]jihah.nah@gmail.com, [2]jwkim@smu.ac.kr (Corresponding Author)

Abstract

BitTorrent is an issue for copyright protection and infringements should be punished based on evidence of invasion of the rights of others. In the BitTorrent system, seeders possess the complete content and leechers have partial content, and the content is shared among peers. Copyrighted content is easily distinguished with complete content but it might not be obvious with partial content. Thus, we propose an original content recovery technology based on Pulse Code Modulation pieces obtained from the CD ripping of music held by leechers. The proposed technique uses a correlation-based method that exploits the similarity of music. We verified that the proposed algorithm could recover pieces of files by comparing the output value of a correlator with that of the input piece, using a number of wave file channels and 8/16 bits.

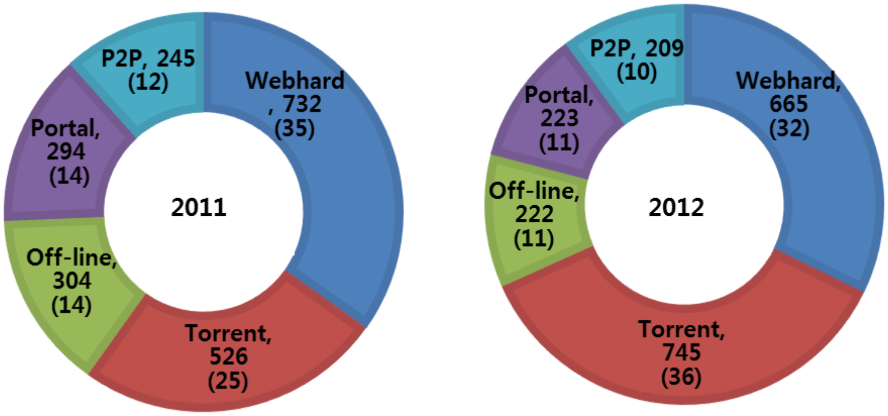
Keywords: *BitTorrent, Complete content, Correlator, Piece, Wave file*

1. Introduction

Copyright infringement by BitTorrent is a very serious problem and existing methods for its prevention, such as packet filtering in networks, are not practical solutions because of the problem of limited network resources and legal constraints [1-2]. These methods would have to process several packets at speeds above Gbps for a copyright filtering device to work on the gateway of a network operator for internet services because it cannot be installed on every system. These could be achieved using hardware but the high cost of this type of system would place a lot of pressure on users who pay for services. Nevertheless, a deep inspection of packets is required to identify the illegal distribution of copyrighted contents via BitTorrent, although this could cause major problems with invasion of privacy.

Recently, South Korea has started a crackdown on heavy BitTorrent uploaders and a heavy uploader was caught who was responsible for the large-scale illegal distribution of torrent seed files [3]. However, there is some controversy about whether a torrent seed file should be considered as literary property [1, 4]. The regulator insists that control of a seed file implies the sharing of copyrighted content, whereas the copyleft camp claims that a seed file is not a literary property so sharing it is not illegal. Irrespective of the legal situation, the Ministry of Culture, Sports, and Tourism announced that the infringement of copyright by BitTorrent is a growing problem. Webhard users constituted a growing component of copyright infringement in 2011, but BitTorrent formed the largest proportion at 36% in 2012 when it is up by 11% from a year ago [5] (see Figure 1).

ISSN: 1975-0080 IJMUE



Source : Ministry of Culture, Sports and Tourism of Korea

Figure 1. Changes in Piracy Methods

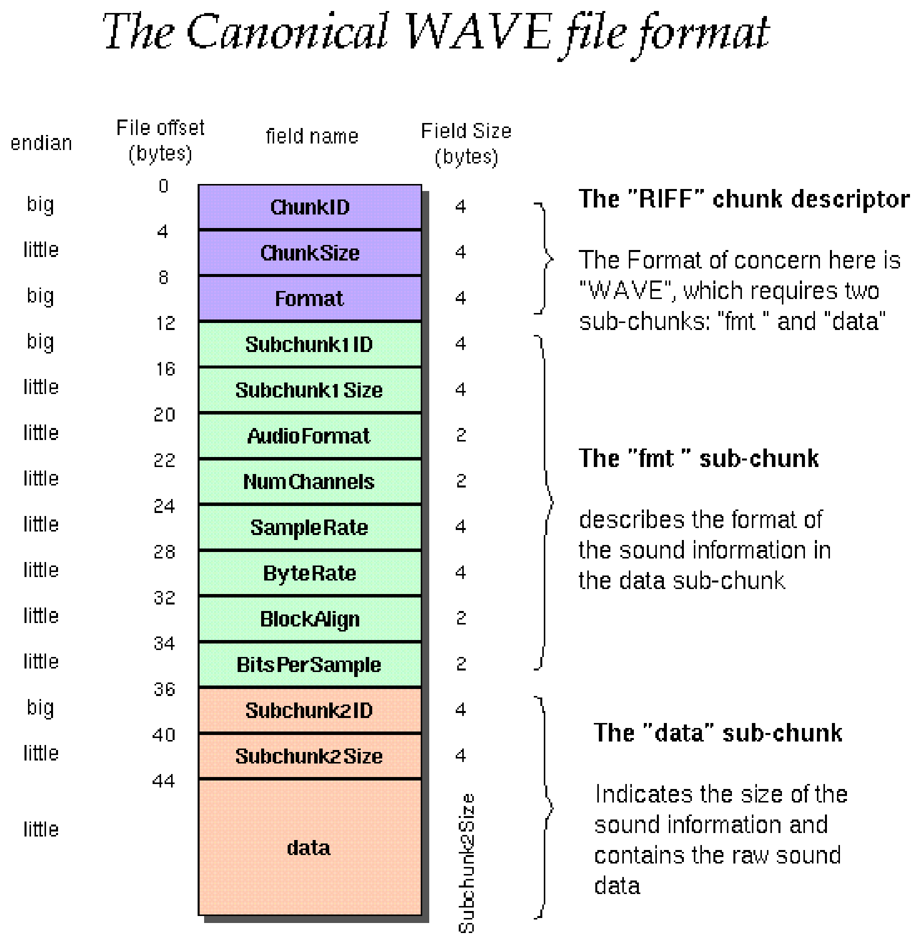
Peers are classified as seeders who have the complete content and leechers with incomplete content, when they share content via BitTorrent. If a leecher downloads 100% of the content, they can become a seeder. Even if a leecher cannot be a seeder, a leecher can upload part of the content. Thus, there is some controversy about whether a seed file or incomplete content can be considered as literary property.

In this study, a raw data recovery technique is proposed to identify whether incomplete content is copyrighted. If raw data can be recovered from incomplete content, the content can be recognized as copyrighted. The focus of this study was the PCM (Pulse Code Modulation) format, which is used for ripping CDs directly to obtain music content.

2. Composition of WAVE files

The WAVE file format is a subset of Microsoft's RIFF (Resource Interchange File Format) specification for the storage of multimedia files. A RIFF file starts with a file header, which is followed by a sequence of data chunks. A WAVE file is often simply a RIFF file with a single "WAVE" chunk, which comprises two sub-chunks: an "fmt" chunk specifying the data format and a "data" chunk that contains the actual sample data. We refer to this format as the "canonical form" [6]. Figure 2 shows the structure of a WAVE file.

Samples are stored consecutively in a single-channel WAVE file. In stereo WAVE files, channel 0 represents the left channel and channel 1 represents the right channel. The speaker position mapping is currently undefined for more than two channels. In multiple-channel WAVE files, the samples are interleaved [7].



The Canonical WAVE file format

endian	File offset (bytes)	field name	Field Size (bytes)
big	0	ChunkID	4
little	4	ChunkSize	4
big	8	Format	4
big	12	Subchunk1ID	4
little	16	Subchunk1 Size	4
little	20	AudioFormat	2
little	22	NumChannels	2
little	24	SampleRate	4
little	28	ByteRate	4
little	32	BlockAlign	2
little	34	BitsPerSample	2
big	36	Subchunk2ID	4
little	40	Subchunk2 Size	4
little	44	data	Subchunk2Size

The "RIFF" chunk descriptor

The Format of concern here is "WAVE", which requires two sub-chunks: "fmt " and "data"

The "fmt " sub-chunk

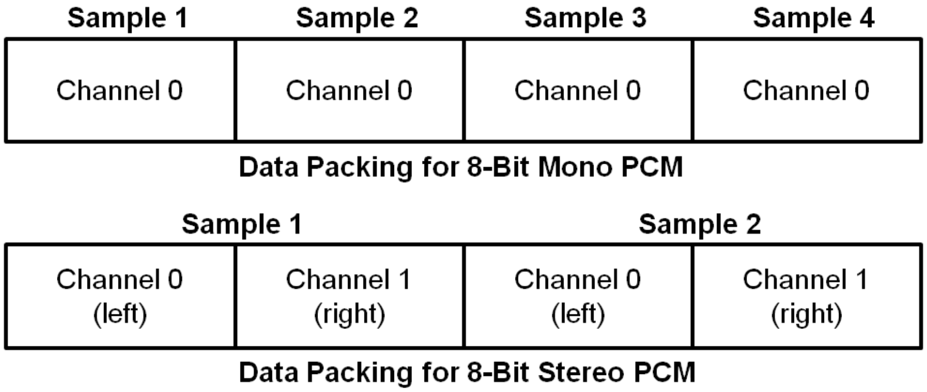
describes the format of the sound information in the data sub-chunk

The "data" sub-chunk

Indicates the size of the sound information and contains the raw sound data

Figure 2. WAVE File Structure

Figure 3 shows the data packing for 8-bit mono and stereo WAVE files.



Sample 1	Sample 2	Sample 3	Sample 4
Channel 0	Channel 0	Channel 0	Channel 0

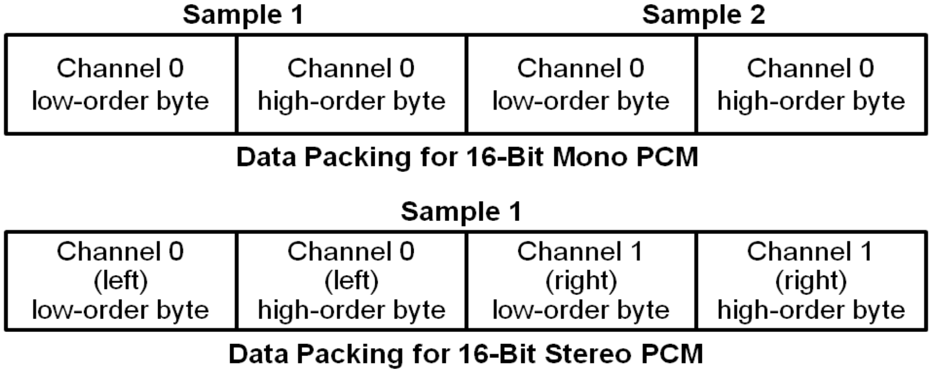
Data Packing for 8-Bit Mono PCM

Sample 1		Sample 2	
Channel 0 (left)	Channel 1 (right)	Channel 0 (left)	Channel 1 (right)

Data Packing for 8-Bit Stereo PCM

Figure 3. 8-bit WAVE Data for Mono and Stereo WAVE Files

Figure 4 shows the data packing for 16-bit mono and stereo WAVE files.



Sample 1		Sample 2	
Channel 0 low-order byte	Channel 0 high-order byte	Channel 0 low-order byte	Channel 0 high-order byte

Data Packing for 16-Bit Mono PCM

Sample 1			
Channel 0 (left) low-order byte	Channel 0 (left) high-order byte	Channel 1 (right) low-order byte	Channel 1 (right) high-order byte

Data Packing for 16-Bit Stereo PCM

Figure 4. 16-bit WAVE Data for Mono and Stereo WAVE Files

The difference between neighboring samples is very small when transforming WAVE files into the PCM format. In the case of stereo files, the difference between the sample values of two channels is greater than the difference between the sample values of the same channels.

3. Recovery algorithm for WAVE pieces

There has been no previous research into the recovery of copyrighted content. Most recovery studies deal with methods based on market requests used by digital forensics, such as the recovery of broken files or mistakenly deleted files. Thus, digital forensics searches for lost information or recovers deleted information. Data recovery is difficult for digital forensics using only a piece of a WAVE file because it contains virtually no information.

Header file information is required to recover a WAVE file in a playable form. However, the presence of header information is certainly not guaranteed for content stored using BitTorrent, which is downloaded as pieces in random order. Thus, a recovery method is needed for partial content that lacks header information. In particular, it must be able to recover a playable form by cutting the appropriate relevant part of the content because BitTorrent includes partial content and garbage data.

In general, WAVE files contain 16-bit stereo because they are ripped from CDs. However, channel information can be identified in the format, as mentioned in section 2, because of the structure of WAVE files. Table 1 shows the possible channel numbers and the arrangement of data chunks in WAVE files. In total, 12 cases should be considered because there are six channels with 8 or 16 bits per sample.

The characteristics of musical content are presented in the audible frequency band signal and changes are slow rather than rapid. The small difference between neighboring samples means that the correlation coefficient can be used to check for 8-bit or 16-bit data. After checking for 8-bit or 16-bit data, the number of channels can be classified based on the difference in the sample values between channels and the difference in the sample values between neighboring samples.

Table 1. WAVE File Channel Structure

The number of channel	Meaning	Data structure
1	mono	[data][data][data]...
2	stereo	[left][right][left][right]...
3	3 channel	[left][right][center][left][right][center]...
4	quad	[front left][front right][rear left][rear right]...
5	4 channel	[left][center][right][surround]...
6	6 channel	[left center][left][center][right center][right][surround]...

Sample correlations are compared using the following formula.

$$R(k) = \sum_{i=0}^{n-k-1} x_i \cdot x_{i+k} / \sum_{i=0}^{n-1} x_i^2 \quad (1)$$

Table 1 shows the information for the channel, which is divided into six. They are divided into five categories based on the correlation coefficient, i.e., mono, stereo, 3-, 4-, and 6-channels because quad and 4-channel are the same. Each channel is determined by the shift number k based on auto-correlation. For mono music, the correlation with the neighboring sample has the highest value except when k = 0, and 6-channel music has a correlation coefficient when k = 6. Two hypotheses can be stated based on these facts.

Hypothesis 1: Music data contains highly correlated signals, which means that the correlation with a neighboring sample is close to one. Thus, the highest correlation coefficient can be acquired, except autocorrelations.

Hypothesis 2: The number of channels corresponds to the shift number where the correlation coefficient is nearest to 1.

Therefore, the autocorrelation coefficient is one and music content has a signal with a very high correlation. Thus, the correlation of a sample after moving one sample has a value that is close to one. Therefore, the number of channels can be verified by calculating the correlation coefficients of samples. Figure 5 shows the shifted waves of 3-channel mode music content based on samples. If the music content contains three channels as shown in figure 5, the same result can be obtained for a sample shifted once and for the same channel when three samples are shifted.

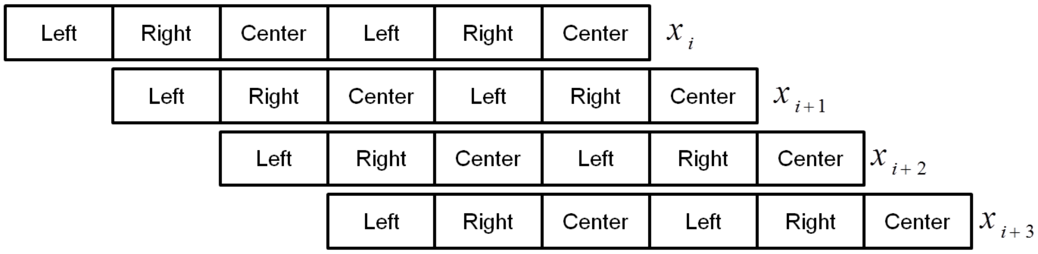

Left Right Center Left Right Center x_i
Left Right Center Left Right Center x_{i+1}
Left Right Center Left Right Center x_{i+2}
Left Right Center Left Right Center x_{i+3}

Figure 5. Original WAVE File and Shifted WAVE File

According to these basic principles, the flowchart for the wave piece recovery algorithm is shown in Figure 6. First, a piece is read and the order is produced for 8-bit and 16-bit arrays using the same data. The correlations are calculated between the original data and the one sample-shifted data. The shift numbers and correlation coefficients are recorded when the correlation coefficient is maximized and the correlation with the one sample-shifted value is recalculated. After determining the correlations with shift numbers of 1, 2, 3, 4, and 6, the piece is rearranged and recovered using the shift number with the maximum correlation coefficient.

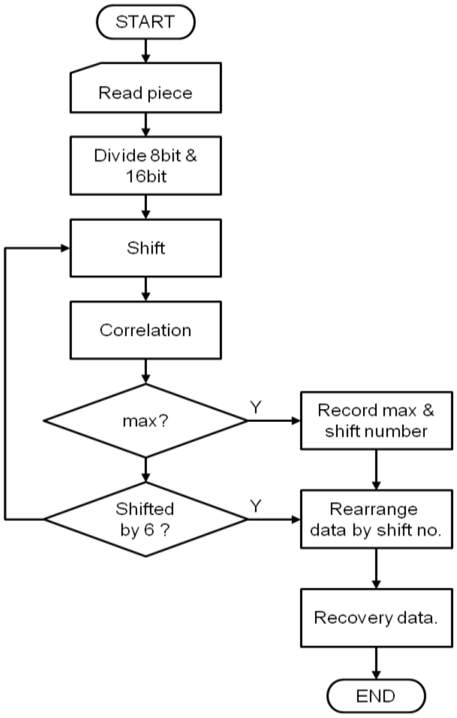


Figure 6. Flowchart of the Proposed Algorithm

4. Evaluations

To assess the validity of the proposed algorithm, several types of WAVE files were partitioned into pieces. In general, the default size of a piece is set to 1 MB in BitTorrent but it may be variable, depending on the size of the content. According to the specification of BitTorrent, the minimum size is 128 KB and the minimum length for a piece of 128 KB is 16 bits for 6-channel data. Therefore, samples larger than 10 K may exist. A higher number of samples may increase the reliability of the correlation coefficient but it requires a large volume of computational resources. Thus, 1024 samples were used for 16-bit data with two channels. In this evaluation, the music used as the test samples had the commonly used stereo and mono sampling frequencies of 22.05 kHz and 44.1 kHz.

Table 2. Test Samples used in the Evaluation

Sampling rate (kHz)	Channels		Number of pieces
22.05 (8-/16-bit)	Mono	Stereo	100
44.1 (16-bit)			100

600 pieces were used totally to test the algorithm, *i.e.*, two channels with three samples, each of which uses 100 pieces. After applying the algorithm to each case, histograms of the correlation coefficients were obtained, as shown in Figures 7–12. Figure 7 shows the histogram for 100 correlation coefficients from each correlator where the correlation algorithm was performed with 22.05 kHz, 8-bit, mono music. Overall, the results for the correlator with 8-bit mono music were are greater than 0.9, whereas the results for the remaining five correlators were distributed among several values. Similarly, Figure 8 shows the histogram for each correlator with 22.05 kHz, 8-bit, stereo music. The results for the correlator with 8-bit stereo music were distributed among values of greater than 0.9 whereas the results for the remaining five correlators were distributed among several values. As shown in Figure 7 and Figure 8, a histogram with values of greater than 0.5 could be obtained with any correlator using an 8-bit sample and the specific correlators could be separated from each other. In particular, the correlation coefficient obtained from an individual piece could separate the channel easily because the coefficient was greater than the correlation coefficients of other correlators. The proposed algorithm did not perform well with 8-bit music, but it could classify and recover low-resolution music in an effective manner.

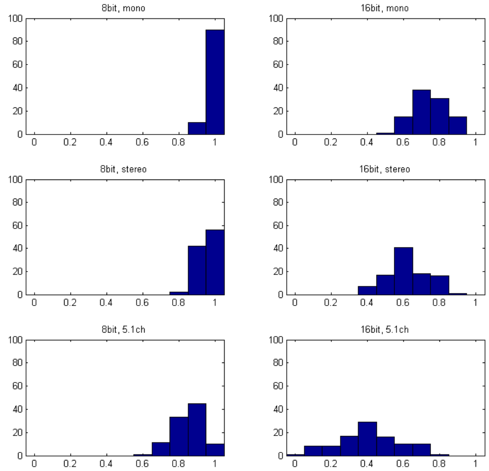


Figure 1. Results for 22.05 kHz/ 8-bit/mono music

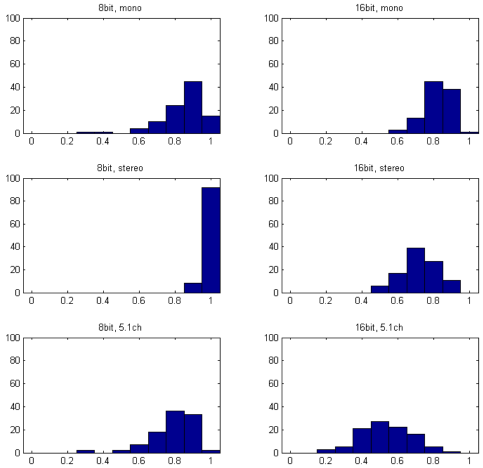


Figure 2. Results for 22.05 kHz/ 8-bit/stereo music

Figure 9 shows the histogram of the correlation coefficients obtained when 22.05 kHz, 16-bit mono music was passed through six correlators. The correlation coefficients of the 8-bit mono correlator and the 8-bit, 5.1-channel correlator were distributed close to zero, and the correlations were very low. For 16-bit mono, the histogram was distributed among values of greater than 0.9 and the input music could be recovered to yield reproducible data after verifying that it was 16-bit mono. Figure 10 shows the histogram of the results for six correlators with 22.05 kHz, 16-bit, stereo music. Similar to figure 9, the histogram of the correlation coefficients for 16-bit stereo had high correlation coefficients and the remaining correlators had histogram distributions that were close to zero, or highly dispersed values.

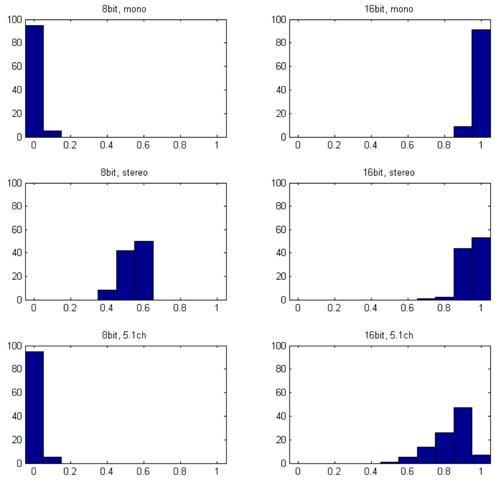


Figure 3. Results for 22.05 kHz/ 16-bit/mono music

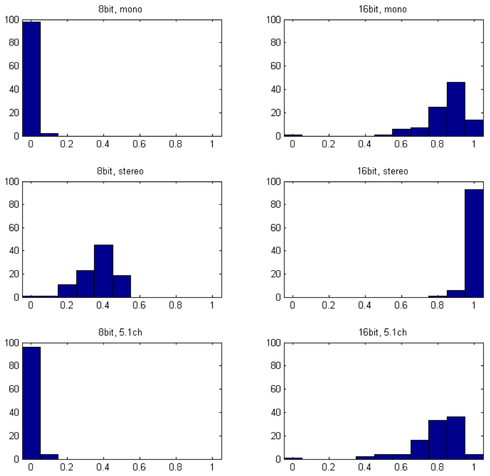


Figure 4. Results for 22.05 kHz/ 16-bit/stereo music

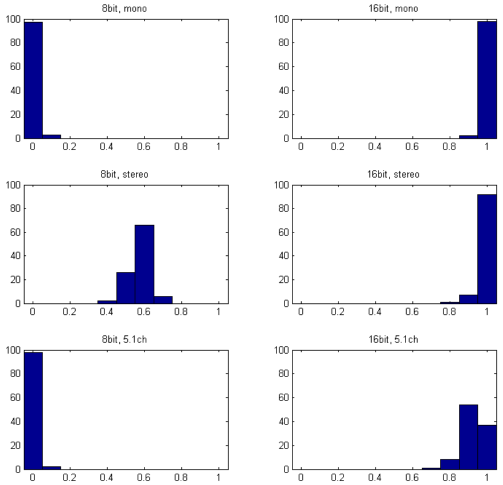


Figure 5. Results for 44.1 kHz/ 16-bit/mono music

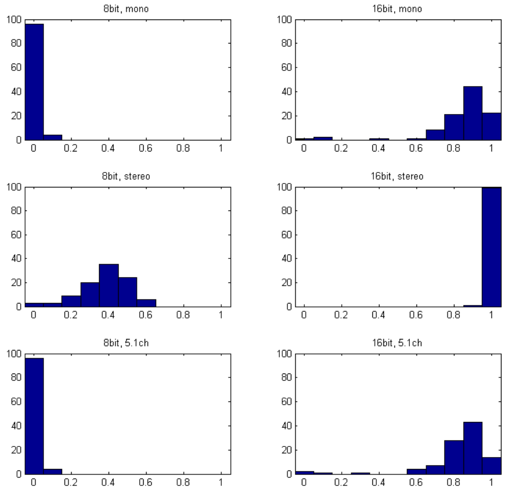


Figure 6. Results for 44.1 kHz/ 16-bit/stereo music

Figure 11 and Figure 12 show the histograms obtained for 44.1 kHz, 16-bit, mono and stereo music using six correlators. The results obtained with three 16-bit correlators demonstrated effective separation because the correlation coefficients were high compared with three 8-bit correlators. However, the 16-bit mono correlator and stereo correlator had value distributions that were close to one in figure 11, which were caused by very similar values because the correlation of the stereo correlator when analyzing mono music was calculated by shifting one sample more than the mono correlator. However, it could be distinguished systematically because the results obtained with the mono correlator were closer to one, which showed that the mono correlator had a correlation coefficient with one sample shift whereas the stereo correlator had a correlation coefficient with two sample shifts. In figure 12, the histogram of the 16-bit stereo correlator for input music contained distinctive values.

5. Conclusions

In this study, we developed an effective algorithm for recovering music content from WAVE files when they are circulated illegally via BitTorrent, specifically when they are configured as aggregated pieces rather than complete content. The proposed algorithm analyzes the PCM characteristics of WAVE files and reconstitutes reproducible data from pieces by identifying the number of bits and channels based on the correlation coefficients, because music data comprises highly correlated signals.

We evaluated the proposed algorithm using six sample types where each sample comprised 100 pieces of 22.05 kHz, 8/16-bit, mono/stereo music or 44.1 kHz, 16-bit, mono/stereo music. The proposed algorithm could recover each piece without error for all 600 pieces. The recovered music could be used to identify the title with a feature-based recognition technique, thereby distinguishing copyright infringements.

This algorithm can identify incomplete content rapidly and detect the illegal distribution of copyrighted content to protect copyright. This method may help to promote the copyright industry. In the future, this method will be extended to reproducible data from complex pieces.

Acknowledgements

This research project was supported by the Ministry of Culture, Sports, and Tourism (MCST) and by the Korea Copyright Commission in 2013.

References

[1] A. M. Mateus and J. M. Peha, “Quantifying Global Transfers of Copyrighted Content using BitTorrent”, Proceedings of the 39th Research Conference on Communication, Information and Internet Policy TPRC (2011), **(2011)** September 23-25, Arlington, U.S.

[2] R. Layton, P. A. Watters and R. Dazeley, “How much material on BitTorrent networks is infringing content?”, A validation Study, Document of Internet Commerce Security Laboratory, **(2010)** November. http://www.icsl.com.au/__data/assets/pdf_file/0017/127304/validation_study_ nov_2010.pdf

[3] News Release, Ministry of Culture, Sports and Tourism, http://www.mcst.go.kr/web/s_notice/press/pressView.jsp?pSeq=12751 , May **(2013)**

[4] http://www.imaso.co.kr/?doc=bbs/gnuboard.php&bo_table=article&wr_id=42966 visited Oct. **(2013)**

[5] http://www.hankyung.com/news/app/newsview.php?aid=2013053038821 visited Oct. **(2013)**

[6] https://ccrma.stanford.edu/courses/422/projects/WaveFormat/ visited Oct. **(2013)**

[7] Multimedia Programming Interface and Data Specifications 1.0, IBM and Microsoft, **(1991)** August.

Authors

Jihah Nah, she received the MS degree in electronics engineering from University of Seoul. She is currently a graduate student in Electrical and Electronic Engineering Department at Yonsei University, Korea. From 1993 to 2003, she was a member of the research staff of the ATM Switching Lab at ETRI. She was manager of the Embedded Software Team at KIPA from 2003 to 2008. Her research interests are in the areas of copyright protection, image processing and embedded software development.

Jongweon Kim, he received the Ph.D. degree from University of Seoul, major in signal processing in 1995. He is currently a professor in the Department of Intellectual Property at Sangmyung University in Korea. He has considerable practical experience in digital signal processing and copyright protection technology in the institutional, the industrial, and academic environments. His research interests are in the areas of copyright protection technology, digital rights management, digital watermarking, and digital forensic marking.