Analysis of FLAC Music Pieces Recovery

Ruichen Jin and Jongweon Kim

***Abstract*—In this paper, we propose a method of recovering the music piece of FLAC file downloaded by BitTorrent. It was decoded the split FLAC file via getting the complete frames from piece of FLAC file and adding the temporary header to decode. The percentage of successfully decoding is all above 90%. From the partial audio signal we extract the musical feature and calculate the cross-correlation with the features in database. The feature in database is extract from the complete music file. The Musical feature is detected by chromagram to get the key strength. The percentage of audio identification is growing tendency with the increase of piece size. It experiment shows probability of identify audio with getting the part of FLAC file distributed by BitTorrent.**

***Index Terms*—FLAC, BitTorrent, recovery, musical feature, chromagram.**

I. Introduction

Recently, due to the rapid development of network technologies and the widespread use of personal computers as multimedia systems, the problem of illegal distribution has become a social issue [1]-[5]. BitTorrent is the most popular form of modern P2P (peer-to-peer) file sharing. BitTorrent sharing has been the primary means for users to share software, music, movies, and digital books online, but most contents are distributed illegality.

FLAC is one of the audio formats that most people used. As people are more inclined high-quality music and their device has enough storage, more and more people prefer the music format-FLAC.

For protecting the copyright of musical work, it needs the metadata of the musical work. The metadata of FLAC file is in the header of file. When the file was FLAC as pieces, it cannot extract the metadata with the piece file. We proposed a method of audio recognition with a few pieces of FLAC file.

II. BitTorrent

BitTorrent (also known as "torrents") is protocol which supports P2P file sharing, and it works by downloading piece of files from different peers at the same time. It is designed to reduce server load and share files from various sources. BitTorrent sharing contains the process of swarming and tracking, where users download many pieces from different sources at once [6]. It is faster than downloading a large file from a single source. Swarming is splitting large files into hundreds of pieces, and then sharing those pieces across a swarm of linked users when specific servers help swarm users find each other. Swarm members use special BitTorrent client software to upload, download, and reconstruct the many file bits into complete files. BitTorrent files (seed) act as pointers during this whole process, helping users find other users to swarm with, and enforcing quality control on all shared files with hash information. (The hash is computed on each piece. These hashes are supplied in the BitTorrent file for peers to verify pieces.). Fig. 1 shows the BitTorrent tracker identifies the swarm and helps the client trade pieces of the file with other peer. Peer with BitTorrent client receives and sends multiple pieces of the file.

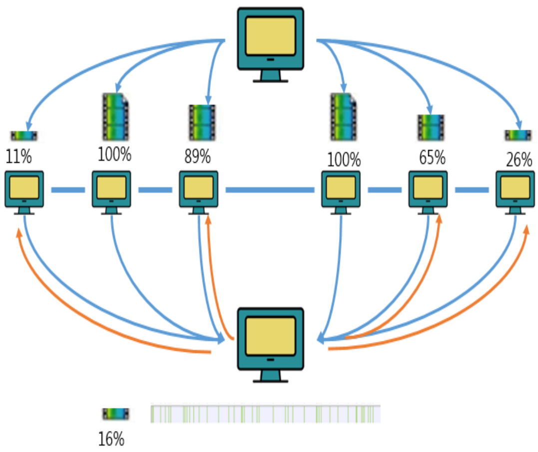

Fig. 1. File sharing with BitTorrent.

Every piece of the BitTorrent file has same size, like if the original file is of 10 MB then each peer could be of 1 MB each or as 256 KB each.

III. FLAC File Format

FLAC (Free Lossless Audio Codec) is a losslessly compressed codec (compressor-decompressor or coder-decoder) of digital audio which allows to be such that file size is reduced without any information being lost. Digital audio compressed by FLAC's algorithm can typically be reduced to 50–60% of its original size [7].

FLAC is an open format and a reference implementation which is free software. FLAC has support for metadata tagging, album cover art, and fast seeking.

A. File Structure

The basic structure of a FLAC bit stream consists of the four byte string "fLaC" at the beginning, the first four bytes are to identify the FLAC stream. Followed by mandatory the 'STREAMINFO' metadata block, zero or other metadata blocks, then one or more audio frames.

Manuscript received November 29, 2013; revised January 8, 2014.
Ruichen Jin is with the Dept. of Copyright Protection, Sangmyung University, Seoul, 110-743, Korea (e-mail: jinruichen@smu.ac.kr).
JongWeon Kim is with the Dept. of Intellectual Property, Sangmyung University, Seoul, 110-743, Korea (corresponding author, e-mail: jwkim@smu.ac.kr).

Fig. 2 shows the FLAC file structure detail. The metadata contains all the information about the stream except for the audio data itself. A decoder is allowed to skip any metadata types it does not understand. But the STREAMINFO block is mandatory. This block includes the information like the sample rate, number of channels, etc. Those can help the decoder manage its buffers, like the minimum and maximum data rate and minimum and maximum block size.

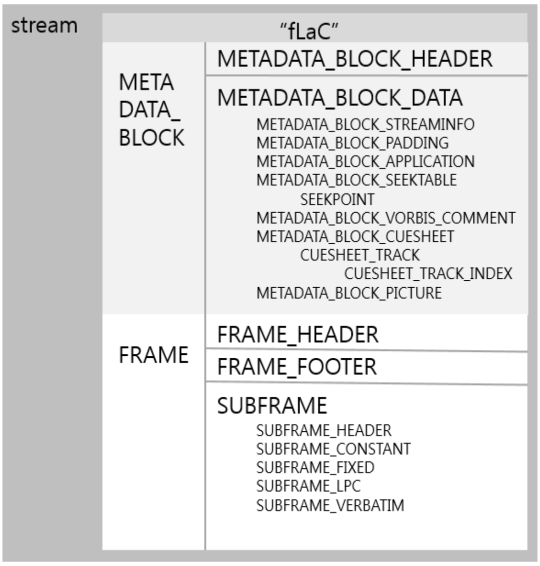

Fig. 2. FLAC file structure.

An audio frame is preceded by a frame header and trailed by a frame footer. The frame header contains the minimum necessary information for a decoder to play the stream. It also contains the block or sample number and an 8-bit CRC (cyclical redundancy check) of the frame header. The frame footer contains a 16-bit CRC of the decoded audio signals for error detection. After the header comes encoded audio data.

B. Pieces Recovery Experiment

The piece size can be set when generate BitTorrent file, and the sizes are among 16KB to 16384KB (16MB). The formal of the popular music (about 3min ~ 5min) which is FLAC format is about 25MB~40MB. Then the 128KB pieces contain about 1 second audio signals. Although the metadata block header contains the MD5 signature of the unencoded audio data to check the error detection, each frame has their own CRC code. For decoding the piece file, we need complete one frame at least. Through the experiment, we learned a piece file contains one more frames generally. We get the complete frames to read the piece file of FLAC files downloaded by BitTorrent client program and find the 14 bits Sync code "1111 1111 1111 10", and check some obvious parameters. We extract one or more complete frames between the first and last found sync code. The experiment is proceeded with 360 pop songs and implemented reference the FLAC library. We add the temporary metadata header at the head of the first frame because of some lacking parameter for encoding. The temporary metadata header is comprised of statistical data. We get the statistical data via analyzing hundreds of pop songs.

Table I shows the result of file encoding with piece file. We used the piece size among 128KB and 16MB. When we used the piece size 512 it can be decoded 336 of 360 and the percentage is 93.3%. The decoding percentage exceeded 90% of all. Fig. 3 shows the percentage of decoding is growing tendency with the increase of piece size.

TABLE I: THE RESULT OF PIECE FILE DECODING

Piece Size	Decoded	Percentage
128KB	330	91.67%
256KB	328	91.11%
512KB	336	93.33%
1024KB (1MB)	360	100.00%
2048KB (2MB)	354	98.33%
4096KB (4MB)	355	98.61%
8192KB (8MB)	360	100.00%
16384KB (16MB)	360	100.00%

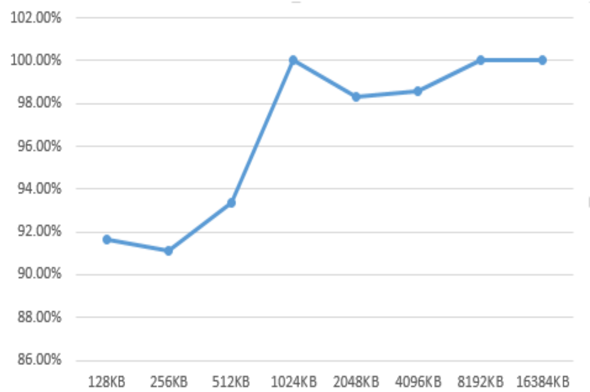

Fig. 3. The tendency of decoding percentage.

IV. MUSICAL FEATURE EXTRACTION

There are a few methods to extract the musical feature from audio. Each musical feature is related to one of the musical dimensions traditionally defined in music theory [8]. Zero-crossing rate is one of the simplest feature. It counts the number of sign changes of audio waveform. The Signal energy is computed using RMS (Root Mean Square) [9]. And many features can be derived from the FFT-based spectrum. It can be computed by the frequency domain or Mel-bands. The estimation of pitch is usually based on spectrum and auto-correlation [10]. Computing the cross-correlation of its pitch class distribution, with the distribution we can associate to each possible tonality can estimate the tonality of a musical piece.

A. Tonality Analysis

Tonality is a system of music in which specific hierarchical pitch relationships are based on a key "center". There were few researches devoted to estimate the tonality from audio signals [11], [12].The spectrum of the energy along the pitch is called chromagram. The Chromagram is obtained by converting from frequency domain to the pitch domain using a log-frequency transformation. The chromagram is wrapped by fusing the pitches belonging with same pitch classes. It shows a distribution of the energy with respect to the twelve possible pitch classes [13].

The musical feature is extracted by computing the cross-correlation of the pitch distribution and each possible tonality. Collected correlation of the highest will be key strength. Fig. 4 shows the diagram for the calculation of chromagram and estimation of key strength. This method analyzes audio and extract musical feature through an adaptation of the pitch class distribution to the chromagram representation.

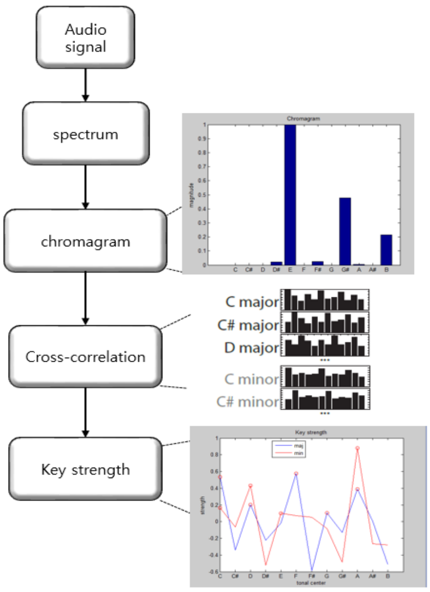

Fig. 4. Diagram for estimation strength.

B. Evaluation

The experiment was proceeding with the decoded audio data which obtained in above experiments. We create a database which contains the features of complete song beforehand. The pitch class distribution was calculated for extracting the feature with FLAC piece file. They are matched if they have the highest correlation value.

TABLE II: THE RESULT OF AUDIO IDENTIFICATION

Piece Size	Recognition	Percentage
128KB	46	13.94%
256KB	60	18.29%
512KB	124	36.90%
1024KB (1MB)	257	71.39%
2048KB (2MB)	325	91.81%
4096KB (4MB)	331	93.24%
8192KB (8MB)	357	99.17%
16384KB (16MB)	360	100.00%

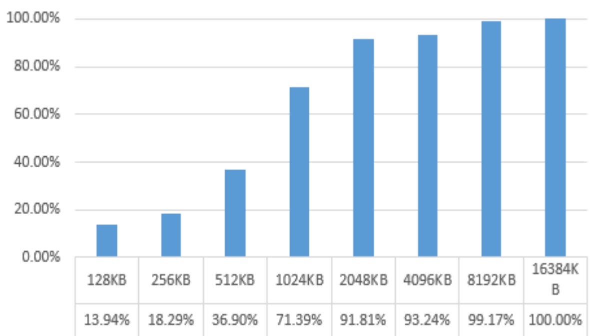

Fig. 5. The tendency of identification percentage.

Table II shows the results of audio identification after file decoding with piece file. We used the piece size between 128KB and 16MB. When we used the piece size 256KB it can be identified correctly 60 of 328 and the rate is 18.29%. Fig. 5 shows the percentage of identification is growing tendency with the increase of piece size. The piece size above than 1MB has stable tendency.

V. CONCLUSION

In this paper, we proposed a method to extract feature form recovery audio signal of FLAC piece file downloaded by BitTorrent. For recovering the split FALC file, we made a temporary header with statistics from analyzing the FLAC format. The musical feature is extracted by computing the cross-correlation of chromagram, and matched with the feature of piece file in database. The results of experiment shows when the piece size is gathers than 1MB, and it has stable tendency in decoding piece file and audio identification. The experiment presented us the probability of tracking the illegal contents without complete file.

ACKNOWLEDGMENT

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

REFERENCES

[1] J. Kim, N. Kim, D. Lee, S. Park, and S. Lee, "Watermarking two dimensional data object identifier for authenticated distribution of digital multimedia contents," *Signal Processing: Image Communication*, vol. 25, issue 8, pp 559–576, 2010.

[2] Y. Lee and J. Kim, "Robust blind watermarking scheme for digital images based on discrete fractional random transform," *Communications in Computer and Information Science*, vol. 263, pp 139-145, 2011.

[3] J. Nah, J. Kim, and J. Kim, "Video forensic marking algorithm using peak position modulation," *Applied Mathematics & Information Sciences*, vol. 7, no. 6, pp. 2391-2396, 2013.

[4] R. Jin and J. Kim, "A digital watermarking scheme using hologram quantization," in *Proc. International Conference on Signal Processing, Image Processing and Pattern Recognition 2012*, November 2012, vol. 342, pp 39-46.

[5] J. Lee and J.Kim, "Modeling of a copyright protection system for the BitTorrent environment," in *Proc. International Conference on Computer Applications for Security, Control and System Engineering*, *CCIS339*, 2012, pp. 46-53.

[6] T. Karagiannis, A. Broido, M. Faloutsos, and K. C. Claffy, "Transport layer identification of p2p traffic," in *Proc. the 4th ACM SIGCOMM Conference on Internet Measurement*, 2004, pp. 121-134.

[7] FLAC-Free Lossless Audio Codec. [Online]. Available: http://Xiph.org

[8] G. Tzanetakis and P. Cook, "Multifeature audio segmentation for browsing and annotation," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 1999, pp. 103-106.

[9] G. Peeters, "Music pitch representation by periodicity measures based on combined temporal and spectral representations," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2006, vol. 5.

[10] E. Pampalk, A. Rauber, and D. Merkl, "Content-based organization and visualization of music archives," in *Proc. the 10th ACM International Conference on Multimedia*, 2002, pp. 570–579.

[11] G. Martens, H. de Meyer, B. de Baets *et al*., "Distance-based versus Tree-based key recognition in musical audio," *Soft Computing*, 2004.

[12] Y. Zhu, M. S. Kankanhalli, and S. Gao, "Music key detection for musical audio," in *Proc. the 11th International IEEE Multimedia Modelling Conference*, 2005, pp. 30-37.

[13] E. Gomez, "Tonal description of polyphonic audio for music content processing," *INFORMS Journal on Computing*, vol. 18, no. 3, pp. 294–304, 2006.

Ruichen Jin received her B.S. degree in computer science and technology from Yn Bian University, China, in 2011. She is currently pursuing the Ph.D. degree in copyright protection, Sangmyung University, Korea. Her research interests are digital watermarking, multimedia forensics, digital signal processing, and information security.

Jongweon Kim received the Ph.D. degree from University of Seoul, major in signal processing in 1995. He is currently a professor of Dept. of Intellectual Property at Sangmyung University in Korea. He has a lot of practical experiences in the digital signal processing and copyright protection technology at the institute, the industry, and college. His research interests are in the areas of copyright protection technology, digital rights management, digital watermarking, and digital forensic marking.