MP3 Bit Rate Quality Detection through Frequency Spectrum Analysis

Brian D'Alessandro
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Newark, New Jersey 07102
bmd5@njit.edu

Yun Q. Shi
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Newark, New Jersey 07102
shi@njit.edu

ABSTRACT

The proliferation of the lossy MP3 format as the standard for audio transferred over the internet is of great concern to audiophiles, those who deeply care about good audio quality. Typically, the bit rate of an MP3 file is used as a relative measure of audio quality, however, this check fails if the audio has been transcoded from a lower bit rate to a higher bit rate. In this paper, we propose a method of detecting the original lower bit rate of a given audio file by analyzing its high frequency spectrum. Using a Support Vector Machine classifier and five classes of bit rates (CBR 128 kbps, 192 kbps, 256 kbps, 320 kbps, and VBR-0), our algorithm returned an average success rate of 97% in correctly detecting the original compressed bit rate of an audio file in the absence of any coding format knowledge other than the audio signal itself. Furthermore, our algorithm also detected the original lower bit rates of 320 kbps MP3s transcoded from 128 kbps and 192 kbps sources with a success rate of 99%.

Categories and Subject Descriptors

H.5.5 [**Information Interfaces and Presentation**]: Sound and Music Computing – *signal analysis, synthesis, and processing*

General Terms

Algorithms, Security, Verification

Keywords

MP3, Transcoding, Audio Quality, Forensics, Spectrum Analysis

1. INTRODUCTION

Over the past decade, the MPEG Layer III (MP3) compression standard has become the most popular means of transferring and storing audio electronically. File sharing services such as BitTorrent, and devices such as the iPod have all fueled the demand for MP3 and have pushed the format to mainstream usage. MP3 is desirable because of its high compression rate; a song compressed using MP3 might be only one tenth the size of the original CD-quality recording. To achieve compression rates this high, one cannot rely on lossless techniques alone, such as Huffman coding. Instead, some part of the information from the original audio must be discarded, which consequently makes MP3 a lossy format. Yet the information is discarded intelligently so that the final sound is still close enough to the original that casual listeners will not notice the compression.

Under the MP3 standard, an audio file can be compressed at different bit rates. Typically, the higher the bit rate, the better the audio will sound, since more bits are available to encode nuances present in the original signal. On the other hand, low bit rates may be subject to compression artifacts, that is, audible differences from the original [7]. Audio can be encoded either at a constant bit rate or a variable bit rate. A constant bit rate of 128 kbps is usually the lowest acceptable bit rate for decent music quality, while 320 kbps is the highest that most MP3 encoders will allow and is generally regarded as excellent quality.

The driving principle behind MP3 encoding is to eliminate parts of the audio which are deemed to be imperceptible to human hearing. There are many techniques to do this [5], but the one we will focus on is the elimination of high frequencies present in the audio. CD-quality music is encoded at a sampling frequency of 44.1 kHz, which according to Nyquist criteria, encapsulates signal frequencies up to 22.05 kHz. However, the human auditory system can only hear frequencies up to a maximum of 18-20 kHz, and this threshold typically degrades over the years for an individual. As a result of this natural observation and as a way to decrease the bit rate without sacrificing perceptive quality, MP3 encoders eliminate varying amounts of information contained in frequencies greater than 16 kHz.

Based on the examination of a few randomly selected songs, we observe that the amount of information removed from the high frequency range (16 kHz – 22 kHz) of music provides a clue as to the encoded bit rate of the audio, without any other information available, such as compression format or file size. Evidence of this can be seen in the power spectral density plots of Figures 1 and 2. In Figure 1, high frequencies above 16 kHz are entirely nonexistent in the 128 kbps plot, but gradually appear as the bit rate is increased towards 320 kbps. The song used in this example, Metallica's "Master of Puppets," is a fast paced hard rock song. However even for slower, softer music such as Sarah McLachlan's "Building a Mystery" in Figure 2, the frequency characteristics over the different bit rates exhibit the same trend.

MM&Sec'09, September 7–8, 2009, Princeton, New Jersey, USA.
Copyright 2009 ACM 978-1-60558-492-8/09/09...$10.00.

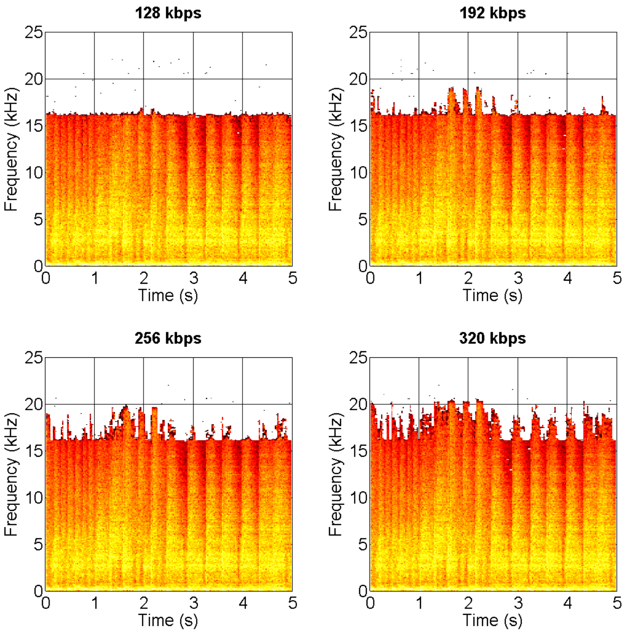

Figure 1. Power Spectral Density plot of a five second clip from Metallica's "Master of Puppets" at 128, 192, 256, and 320 kbps.

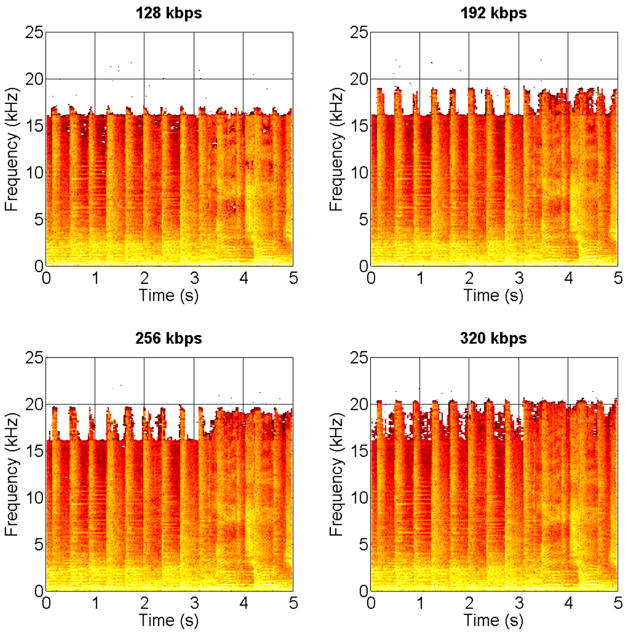

Figure 2. Power Spectral Density plot of a five second clip from Sarah McLachlan's "Building A Mystery" at 128, 192, 256, and 320 kbps.

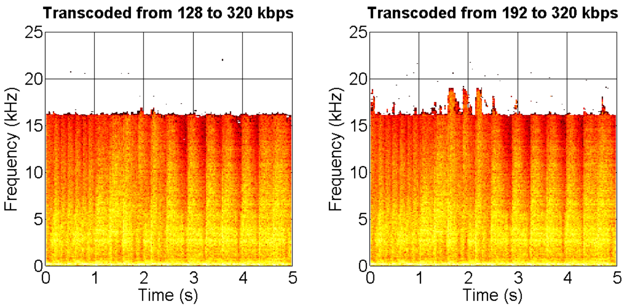

Figure 3. Power Spectral Density plot of a five second clip from Metallica's "Master of Puppets" at 320 kbps, which has been transcoded from 128 and 192 kbps.

This suggests that this type of frequency analysis is applicable over a wide range of genres of popular music.

In addition, even after the song has been manipulated, and perhaps transcoded into a higher bit rate than the original, the loss of high frequency characteristics remains. An example of this can be seen in Figure 3. A 128 kbps audio clip and a 192 kbps clip were both transcoded to a higher bit rate, 320 kbps. These 320 kbps audio clips will logically not sound any better than their source, so judging sound quality purely by looking at that bit rate is misleading. Deeper analysis must be done. We found that the high frequency characteristics of these seemingly "excellent quality" songs were still highly correlated with the characteristics of the original source; in fact, they were nearly identical. In Figure 3, the 320 kbps MP3 transcoded from a 128 kbps MP3 matched the spectrum seen in the native 128 kbps MP3 found in Figure 1, and likewise for the 192 kbps sourced transcode. Thus, by carefully observing this high frequency range, the true quality level based on original encoded bit rate can be determined, regardless of the bit rate claimed by the file itself.

Numerous papers have been published in the field of audio forensics, with applications such as audio manipulation authentication [6], steganography and steganalysis [4, 9], and watermarking [3]. Likewise, there has been previous investigation into the authenticity of digital compact disc recordings using the Aucdtect (CD authenticity detector) algorithm [8]. This algorithm is similar to our proposed method in that it uses a learned classifier based on frequency characteristics of a recording. However, the program created to utilize the algorithm will only function on CD audio streamed directly from the CD, not already compressed MP3 audio which has been ripped onto a personal computer or downloaded from an unknown source on the internet. In addition, since the algorithm only searches for MPEG artifacts in general, it can only discriminate the audio between full CD quality and lossy MPEG quality. No indication is made about the relative sound quality of the audio if it does happen to be in a lossy format. The objective of our algorithm, on the other hand, is to investigate how MP3 audio at multiple bit rates affects the frequency spectrum, and if those characteristics alone can be used to reverse detect the bit rate and thus evaluate relative sound quality.

One important benefit of such analysis is to verify the claimed quality of audio obtained from an unknown source, such as the internet. Music can easily be ripped from CD, streamed, downloaded, and transferred among peers; all though free mainstream software available to anyone. It is quite common to encounter an MP3 file which has been transcoded into a bit rate higher than its original compressed bit rate. This may happen maliciously (by anti-piracy organizations trying to discourage file

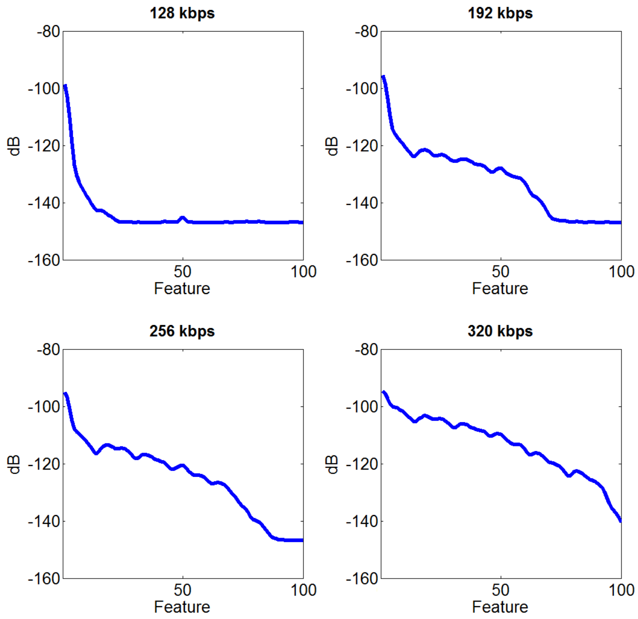

Figure 4. Feature plot of Metallica's "Master of Puppets" at 128, 192, 256, and 320 kbps. The plots correspond to a rough depiction of the song's frequency spectrum from 16-20 kHz at various bit rates.

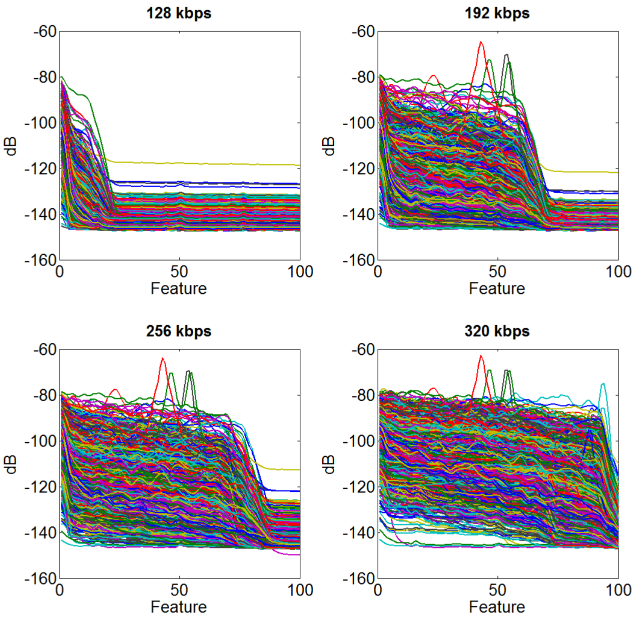

Figure 5. Feature plot of all 2512 songs used in analysis. The plots correspond to a rough depiction of the aggregated frequency spectrum from 16-20 kHz at various bit rates for all songs.

sharing, for example), or accidentally (such as by audio novices who are unfamiliar with the consequences of transcoding). To anyone who subsequently makes a copy of such an MP3 file, there is no evidence that transcoding occurred, other than the fact that the audible quality of the song may sound poorer than the quality typically expected by MP3s of the same bit rate which were directly compressed to that bit rate from its original lossless source. However, without reliable hearing and the true high bit rate version to compare to, proof of transcoding may be difficult just by ear. Instead, an automatic mathematical method of analyzing a suspected MP3 file to detect a low quality song disguised as a high bit rate song would be more reliable. In addition, by being able to verify the quality of an MP3 from an unknown or suspicious source, an assessment can be made about the validity of the source.

An audiophile is an individual who loves music, and there is no better way to listen to music than at the highest quality possible. It is of great concern therefore, when an MP3 has been misrepresented as high quality audio when in fact it has been transcoded from a lower quality file. If such deception could be detected automatically, that algorithm would be of immense forensic benefit to audiophiles and those who place great importance on listening to guaranteed high quality music.

Our general algorithm is as follows. First, each song used in our analysis was converted from full CD quality to a variety of lower bit rates with MP3 compression. The frequency spectrum of each MP3 was then analyzed, condensed, and converted into a feature set for each bit rate of each song in the test set. These features were used as inputs to a Support Vector Machine classifier which picked out the most useful features and used them to differentiate between MP3s of different bit rates with a good deal of success.

2. PROCEDURE

2.1 Data Set Collection

A total of 2512 different songs were used for analysis, representing some 94 different artists including such popular acts as Metallica, Oasis, Nirvana, In Flames, Journey, U2, Pearl Jam, and Radiohead. These songs were all sourced from CD-quality audio and then compressed using the LAME v3.97 MP3 encoder [1], by far the most popular and reliable MP3 encoder currently available. Each song was compressed into five different bit rates: four constant bit rates of 128 kbps, 192 kbps, 256 kbps, and 320 kbps, as well as one variable bit rate of VBR quality level 0 (highest). Variable bit rate encoding adjusts the bit rate temporally within each song depending on the level of compression needed at any particular moment. VBR-0 is the highest level of VBR encoding and corresponds to bit rates ranging from 224-300 kbps on average.

2.2 Feature Extraction

The five bit rates were then used as separate classes in a Support Vector Machine (SVM) classifier. SVM is a learning based classifier which finds the hyperplanes in a multidimensional space which best separate the features of each of the different classes. In order to find the correct demarcation between each of the given number of classes, the classifier must be trained with a subset of the original data and information about which classes the elements in those sets belong to. Transformation of the data is often useful in order to better separate the class data. The process to do this is through the use of a kernel function, which remaps data points into a feature space where the hyperplane divisions can be more easily defined.

In order to obtain the feature data, the source MP3 files were each decompressed into a 1411 kbps WAV file using the Fraunhofer IIS MP3 Surround Commandline Decoder V1.4 [2]. This was done because audio files in this format can easily be read into MATLAB, and as we have demonstrated, transcoding to a higher bit rate does not affect the frequency characteristics of the audio which we are observing. Furthermore, since this bit rate corresponds to the bit rate of full CD-quality audio, it is thus the highest 'common ground' format by which all further spectral analysis tests can be conducted for all compressed bit rates. MATLAB was used to calculate the actual power spectral density for the entire time domain of each song.

Since every data point in this frequency spectrum would be much too large to realistically consider as features, the dimensionality must be reduced. To do this, only the frequency range of 16 kHz – 20 kHz is considered, since this is suspected to be the only region where noticeable differences exist between the classes of bit rates. In addition, this range was then subdivided into 100 bands of approximately 43 Hz each, and the average power spectral density for each of these bands was calculated. These 100 values were used as the feature set of each song. Figure 4 shows the plot of these features for one song, while Figure 5 shows the plot of these features for all 2512 songs overlaid. While there are distinct differences in the shape of the curve for each bit rate, it is the job of the classifier to determine the correct separation between each class based on the feature set, and then to use those separations to classify a random unknown selection of songs.

2.3 Classifier Parameters

Support Vector Machines employ the use of a kernel function. In order to perform a better classification in the case when the relationship between the data and the classes is not linear, data samples can be remapped using various nonlinear kernel mapping functions [10]. Here we use the polynomial kernel, given by:

$$K(x_i, x_j) = (\gamma x_i^T x_j + C)^d$$

Parameter values of d=2, γ=1, and C=1 were used. This classifier was trained using features from 500 randomly selected songs, and then tested on the remaining 2012 songs. This procedure was then repeated five times, each time using a different set of 500 random songs for learning with the corresponding 2012 remaining songs for test. The results of these five iterations were averaged.

3. RESULTS

3.1 Bit Rate Differentiation

After the classifier was trained with the appropriate randomly selected training data and tested on the remaining set of songs, the confusion matrix was calculated. The resulting average confusion matrix for five iterations is shown in Table 1. The data bit rates above the five right-most vertical columns represent the corresponding bit rates as detected by the classifier, while the data bit rates to the left of each horizontal row represent the actual bit rates.

It is clear that the classifier was highly successful at differentiating between bit rates based solely on the frequency spectrum of each song. The average classification success rate is 97% on a test set of 2012 songs.

Table 1. Bit Rate Differentiation Confusion Matrix

		Detected Classes				
		128 kbps	192 kbps	256 kbps	VBR-0	320 kbps
Actual Classes	128 kbps	**0.997**	0.000	0.000	0.001	0.001
	192 kbps	0.004	**0.989**	0.002	0.003	0.002
	256 kbps	0.002	0.006	**0.924**	0.063	0.005
	VBR-0	0.004	0.006	0.048	**0.941**	0.001
	320 kbps	0.003	0.002	0.009	0.003	**0.983**

Table 2. Transcoding Detection Confusion Matrix

		Detected Classes				
		128 kbps	192 kbps	256 kbps	VBR-0	320 kbps
Actual Classes	128→320	**0.993**	0.002	0.002	0.002	0.002
	192→320	0.002	**0.994**	0.000	0.003	0.000
	320 kbps	0.002	0.002	0.004	0.003	**0.988**

The most difficult class to classify was the middle class, 256 kbps, with a success rate of only 92.4%. By examining the confusion matrix, 6.3% of the songs which were constant bit rate 256 kbps MP3s were instead classified as variable bit rate quality level 0, while 4.8% of VBR-0 MP3s were erroneously classified as 256 kbps. Undoubtedly, this is due to the fact that since VBR inherently has variation in its bit rate, this naturally leads to it being mistaken for a constant bit rate at a similar level, and vice versa. The distinction here is not as clear cut as it is for the other classes. For example, 128 kbps MP3 files were very distinctive in that nearly all had no frequencies higher than 16 kHz whatsoever. On the other extreme, 320 kbps MP3s were distinctive in that they nearly always had frequency components up to 20 kHz. This led to excellent results for the classification of these bit rates.

3.2 Transcoding Detection

Given the success of this first (and most important) step, the classifier can next be extended towards the problem of detecting transcoded MP3s. No mechanisms exist in MP3 encoding utilities to prevent a person from taking a relatively low quality MP3 file, say, 128 kbps, and then re-encoding it to a higher bit rate, say, 320 kbps. This up-coding really serves no purpose, since MP3 compression is lossy. Once high frequency information about the original audio file has been discarded to compress it down to 128 kbps, that information cannot be recovered (except perhaps with the use of experimental frequency enhancement algorithms [11]). Re-encoding to a higher bit rate will simply use the extra bits to encode the low bit rate signal more fully; in other words, the extra bits become redundant. The consequence of this is that it is impossible to recover the high frequency components lost in the original compression. A spectrum analysis of a 320 kbps MP3 file which has been transcoded from a 128 kbps MP3 file will look nearly identical to the original 128 kbps MP3 itself (see Figure 3).

To verify this, the sets of 2512 128 kbps MP3 files and 2512 192 kbps MP3 files were both transcoded to 320 kbps. The true 320 kbps MP3 files were used as a third class for comparison. These three classes were used as test input to the trained SVM classifier utilized previously. The resulting confusion matrix is shown in Table 2. Again, the data bit rates above the five right-most vertical columns represent the classified, or, detected bit rates, while the data bit rates to the left of each horizontal row represent the actual bit rates: 128 kbps transcoded to 320 kbps, 192 kbps transcoded to 320 kbps, and true 320 kbps.

It is clear from this confusion matrix that the same classifier used to differentiate between different MP3 quality levels can likewise be used to detect transcoded MP3s. 99.3% of the 128 kbps MP3s converted to 320 kbps before analysis were correctly identified as being sourced from 128 kbps, while 99.4% of the 192 kbps MP3s converted to 320 kbps before analysis were correctly identified as being sourced from 192 kbps. These classification rates are very similar to the classification rates of the true 128 kbps and 192 kbps MP3 files, undoubtedly because the spectral properties of the transcoded signal does not change much from the original bit rate.

4. CONCLUSION

In the age of the internet where files such as MP3s can effortlessly be transferred between individuals, verification of the quality of these files can become quite difficult to detect. Since transcoding from a low bit rate to a high bit rate is not restricted, MP3 songs of this type will undoubtedly arise, either by malicious users with the intent to deceive, or by individuals who simply do not know what they are doing and do not understand the consequence of their actions. Occasionally, the transcoding can be detected audibly by listening to a song, however, due to imperfections in human hearing and individuals with untrained ears, detection this way can be difficult, especially when trying to differentiate between bit rates that are already high, for example, between a 256 kbps and a 320 kbps MP3 file. Both of these compression levels are audibly close enough to CD-quality that most listeners cannot tell the difference in a controlled, double blind listening test.

Given these difficulties, the ability to reverse detect the original bit rate from a decompressed audio file is incredibly useful. From a forensics standpoint, this provides the ability to glimpse into the "history" of an encoded audio file and verify if the claimed bit rate is valid. If the detected bit rate is less than the claimed bit rate, then this is good evidence that the audio has been transcoded, and thus, that the original encoded audio file has been tampered with. Such evidence of digital tampering may come in handy in criminal investigations where the authenticity of a recording is of utmost importance. Similarly, high frequency analysis and the changes that occur therein at different compression levels may have applications in areas such as audio steganalysis. Finally, transcoding detection is beneficial to those who simply want to be reassured that the music they download, copy, and listen to is not of a deceivingly lower quality than they expect.

Our results show that such original bit rate detection is very possible when one considers the high frequency spectrum of the audio in question, and compares it with known spectrum patterns for various bit rates. An average of 1945 out of 2012 songs tested (97%) were correctly classified intro their original bit rates using the proposed method. Likewise, an average of 99.4% of songs transcoded from 128 kbps and 192 kbps to 320 kbps were correctly identified as being sourced from the lower original bit rate.

The high frequency spectrum is thus a reliable way of determining the true bit rate based quality of an MP3 song across a range of different songs, artists, and genres. In future work, we would like to explore a larger set of songs with more classes, such as additional VBR quality levels. We would also like to test the reliability of this method using different MP3 encoders, as well as different audio compression algorithms such as AAC.

5. REFERENCES

[1] http://lame.sourceforge.net/.

[2] http://www.all4mp3.com/tools/sw_fhg_cl.html.

[3] Arnold, M., Audio watermarking: Features, applications and algorithms. in *IEEE International Conference on Multi-Media and Expo*, (New York, NY, 2000), 1013-1016.

[4] Böhme, R. and Westfeld, A., Statistical characterisation of MP3 encoders for steganalysis. in *Proceedings of the Multimedia and Security Workshop 2004, MM and Sec'04*, (Magdeburg, 2004), 25-34.

[5] Brandenburg, K., MP3 and AAC Explained. in *AES 17th International Conference: High-Quality Audio Coding*, (1999).

[6] Chen, F., Li, W. and Li, X., Audio quality-based authentication using wavelet packet decomposition and best tree selection. in *Proceedings - 2008 4th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP 2008*, (Harbin, 2008), 1265-1268.

[7] Chi-Min, L., Han-Wen, H. and Wen-Chieh, L. Compression Artifacts in Perceptual Audio Coding. *Audio, Speech, and Language Processing, IEEE Transactions on*, *16* (4). 681-695.

[8] Djuric, A. Tau Analyzer - Aucdtect Algorithm Details. *http://true-audio.com/Tau_Analyzer_-_Aucdtect_Algorithm_Details*.

[9] Gang, L., Akansu, A.N. and Ramkumar, M., MP3 resistant oblivious steganography. in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, (Salt Lake, UT, 2001), 1365-1368.

[10] Hsu, C.W., Chang, C.C. and Lin, C.J. A practical guide to support vector classification, Department of Computer Science and Information Engineering, National Taiwan University, 2003.

[11] Sang-heon, O., Won-Jung, Y., Youn-ho, C., Kyu-Sik, P. and Ki-Man, K. A new spectral enhancement algorithm in MP3 audio. *Consumer Electronics, IEEE Transactions on*, *52* (1). 196-199.