COMPRESSION TRANSPARENT LOW-LEVEL DESCRIPTION OF AUDIO SIGNALS

J. Lukasiak, C. McElroy, E. Cheng
TITR, University of Wollongong
Wollongong, NSW, Australia
jasonl@elec.uow.edu.au

ABSTRACT

A new low level audio descriptor that represents the psycho-acoustic noise floor shape of an audio frame is proposed. Results presented indicate that the proposed descriptor is far more resilient to compression noise than any of the MPEG-7 low level audio descriptors. In fact, across a wide range of files, on average the proposed scheme fails to uniquely identify only five frames in every ten thousand. In addition, the proposed descriptor maintains a high resilience to compression noise even when decimated to use only one quarter of the values per frame to represent the noise floor. This characteristic indicates the proposed descriptor presents a truly scalable mechanism for transparently describing the characteristics of an audio frame.

1. INTRODUCTION

Historically, methods for locating multimedia data in large data repositories (such as the Internet) have relied on matching textual tags that have been linked with the multimedia data at the time of storage. These textual tags, or descriptions, are usually manually generated by either the content author or a designated expert. The versatility and validity of these tags is thus directly limited by the detail and accuracy contained in the associated description. For example, it may be possible to find a particular multimedia item via the author's name or media title. However, it is extremely unlikely that content specific features such as colour, melody or frequency structure would be identifiable using a text-based tag.

A more robust and potentially ubiquitous method for generating multimedia descriptions is via the use of algorithms that extract meaningful context or content description directly from the multimedia bit streams. This methodology is the realm of the new MPEG-7 standard [1]. The MPEG-7 standard provides a framework for describing multimedia content through a series of descriptors that are calculated directly for the multimedia content (low level descriptors) [2]. Having descriptors associated with multimedia data that describe the actual content of the data, provides the potential for powerful manipulation of content supply and consumption. The manipulation could involve finding all multimedia items in a database that have a blue background or selecting the audio segments that represent specific sources (such as dogs barking) [3, 4]. Using these descriptors for search and retrieval greatly reduces the search complexity required, when compared to searching on the raw data.

Previous work involving the MPEG-7 low level audio descriptors has indicated that when audio streams are compressed, the values generated for the MPEG-7 audio descriptors become ambiguous [5, 6, 7]. Also, it was shown in [6] that the ambiguity introduced into the descriptor values by compression has very high entropy and is not easily modeled or predicted. The ambiguity introduced, and the difficulty in modeling its effect, results in the low level audio descriptors forming very poor candidates for search and retrieval targets in practical situations [5, 6, 7]. For example, the low level descriptors produce extremely inconsistent results when employed in a simple search scheme that attempts to find a segment in an MP3 file using the low level descriptors generated from a CD [6]. The aim of this paper is to analyse the characteristics of the current MPEG-7 low level audio descriptors that cause ambiguity when compressed audio streams are used, and in turn, propose an alternative low level descriptor that exhibits transparency to audio compression.

In Section 2, the causes of inconsistency with the MPEG-7 low level audio descriptors are discussed and an alternative scheme is proposed. Section 3 compares the performance of the proposed descriptor with that of the current MPEG-7 descriptors. The major findings and conclusions are then discussed in Section 4.

2. COMPRESSION TRANSPARENT AUDIO DESCRIPTION

2.1. MPEG-7 low level audio descriptors

The research in [5, 6, 7] exploited a subset of five MPEG-7 low level audio descriptors; Audio Power (AP), Audio Waveform (AW), Audio Spectrum Envelope (ASE), Audio Spectrum Centroid (ASC) and Audio Spectral Spread (ASS). A full description of these can be found in [8]. These five descriptors were selected because they were frame-based and also provide a compact description of the underlying audio data. Subsequent to the analysis undertaken in [5, 6, 7], an additional frame-based low level descriptor was added to the MPEG-7 standard. This descriptor, the spectral flatness measure (SFM), is the basis for the audio fingerprint work detailed in [9] (this is now part of the MPEG-7 standard) and is defined as [8]:

$$SFM_b = \frac{Gm(b)}{Am(b)} \qquad b = 1:B \tag{1}$$

where $Gm(b)$ and $Am(b)$ represent the geometric mean and arithmetic mean of the power spectral coefficients of each sub-

0-7803-9332-5/05/$20.00 ©2005 IEEE

band respectively, and B represents the total number of sub-bands used. Each sub-band b represents a ¼ octave band related to 1 khz, with the lowest band edge being restricted to 250 Hz. Thus, the total SFM descriptor is a vector containing B coefficients.

Due to its inclusion as the basis for audio fingerprinting in MPEG-7, an analysis of the SFM descriptor performance in the presence of compression noise is included in Section 3. Also, for consistency, the same 5 descriptors used in [5, 6, 7] are studied for comparative purposes in this paper.

The MPEG-7 low level audio descriptors attempt to extract very accurate objective values from the audio bit stream. An example of this characteristic is clearly evident in the algorithm used to extract the AP descriptor [8]:

$$P(n) = \sum_{i=1}^{S} \left| x(i) \right|^2 \quad (2)$$

where S corresponds to the number of samples in the audio frame. The value calculated via (2) represents the total energy in the audio frame. However, extracting such an exact value from a signal that has distinct perceptual properties creates a definite mismatch when lossy compression of the stream is employed. The objective of lossy audio compression is to retain the perceptual quality of the original audio signal whilst reducing the number of bits required to represent the signal. This is achieved by exploiting the psycho-acoustic properties of the ear to hide quantisation noise in sections of the spectrum that are masked (inaudible to the human ear). This procedure implicitly maintains the psycho-acoustic noise floor of the audio signal.

By removing perceptually irrelevant redundancy from the signal, lossy compression often results in significant modification to the actual spectral shape of the signal. Thus, it is clear that using exacting objective measures to extract descriptors from signals that have undergone perceptual redundancy removal, creates descriptors that will only be appropriate for representing signals compressed with that particular scheme. That is, transparency or versatility of the descriptors across different formats of audio compression is improbable. While the SFM implicitly involves an averaging operation when calculating the spectral flatness of each sub-frame, it still appears probable that lossy audio compression will affect the resultant spectral flatness of some sub-bands. This should still result in the SFM incorrectly representing some compressed audio frames.

2.2. Proposed Low Level Descriptor

In order to extract a descriptor that is relatively transparent to audio compression, we propose that matching the descriptors' structure to the objective of the compression schemes is essential. To this end, we propose using a description of the psycho-acoustic model for an audio frame as a suitable low level descriptor. This descriptor would basically represent the shape of the psycho-acoustic noise floor for the audio file. As preservation of this noise floor is the primary objective for lossy audio coders, this descriptor should exhibit strong resilience to compression distortion.

The proposed Psycho-Acoustic Descriptor (PAD) is based on the psycho-acoustic model presented in [10], with modifications implemented to enhance robustness and improve scalability. The model in [10] was chosen because of the compact nature with which it represents the psycho-acoustic curve; 25 values per frame as opposed to 256 values for methods such as [11]. For each frame of audio the PAD is calculated using steps (a)-(i) below:

a) The signal is converted to the frequency domain using an N point FFT and the power spectral coefficients for each frequency bin f are calculated via:

$$P(f) = \mathrm{Re}^2(f) + \mathrm{Im}^2(f) \quad (3)$$

b) The frequency bins are then mapped to a bark scale using [12] :

$$z(f) = 13\arctan(0.0076f) + 3.5\arctan[(\frac{f}{7500})^2] \quad (4)$$

c) The spectrum is then divided into non-overlapped fixed bark-width bands. The number of bands is calculated as:

$$B = floor(\frac{z_{\max}}{W}) \quad (5)$$

Where $z_{\max}$ is the maximum bark value from (4) and W is the width of each band in barks.

d) The energy in each band is estimated using:

$$E_i = \sum_{lb}^{ub} P(z) \qquad i = 1:B \quad (6)$$

Where lb and ub represent the lower and upper band bark values.

e) The effect of inter-band masking is included by convolving E_i with a spreading function defined as [12]:

$$SF_{db} = 15.81 + 7.5(x + .474) - 17.5\sqrt{1 + (x + .474)^2} \text{ in dB} \quad (7)$$

f) The relative threshold offset in dB is calculated as [10]:

$$O_i = \alpha(14.5 + i) + (1 - \alpha)5.5 \quad (8)$$

$$where \quad \alpha = \min(\frac{SFM_{dB}}{-60}, 1) \quad (8a)$$

$$SFM_{dB} = 10\log_{10}(\frac{Gm}{Am}) \quad (8b)$$

g) The masking threshold is calculated as [10]:

$$T_i = 10^{\log_{10}(E_i) - (O_i/10)} \quad (9)$$

h) The threshold is adjusted back to the bark domain by normalizing each band by the inverse of the energy gain due to the spreading function [10]. The adjusted threshold is labeled T_i' .

i) The adjusted threshold values are normalized (so that the curve represents unit energy) and converted to dB:

$$Tnorm_i = \frac{T_i'}{\sum_{i=1}^{B} T_i'} \quad (9)$$

$$PAD_i = 10\log_{10}(Tnorm_i) \quad (10)$$

In addition to the method described in steps (a)-(i), silent frames are detected prior to PAD calculation and these are simply represented as a vector of zeros. This process removes ambiguity generated when attempting to describe the threshold of silent frames, and hence, improves the PAD's reliability in transparently describing the characteristics of an audio frame.

The other primary modifications to the model described in [10] are the band calculation in step (c) and the normalization in step (i). Using bands of equal bark width, as opposed to the fixed critical band values of [10], allows fine grain variation of

	PAD25	PAD6	SFM24	SFM6	ASE	AP	AW	ASC	ASS
Comb	99.95	97.76	97.98	91.09	99.85	15.70	49.60	15.40	11.40
Inst	99.99	97.69	99.23	91.63	97.50	14.80	38.90	11.50	7.50
Average	99.96	97.72	98.54	91.33	98.68	15.25	43.75	13.45	9.45

Table 1: Percentage correct frame matches

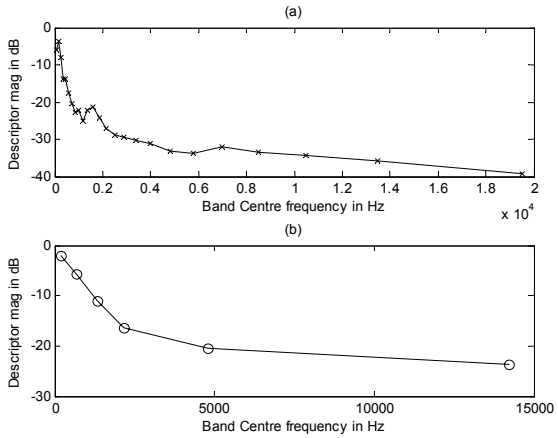

Figure 1: Examples of the PAD (a) 1 bark wide (b) 4.2 bark wide

the bandwidths (and hence the total number of bands) while retaining the ability to represent the entire spectrum. For example, setting W in (5) to 1 bark results in a PAD with 25 values per frame and the bandwidths are essentially identical to the critical bands used in [10]. However, setting W to 4.2 bark, results in a more compact PAD containing only 6 values per frame. A representation of the PAD for an audio frame using both 1 bark wide and 4.2 bark wide bands is shown in Figure 1. Figure 1 clearly demonstrates that the PAD using a bandwidth of 4.2 bark represents the underlying shape of the 1 bark wide PAD, with the fine detail removed.

The normalization of step (i) removes ambiguity introduced into absolute amplitude values by compression schemes and volume variations. This produces a descriptor that represents only the shape of the current frame's psycho-acoustic curve. Also, the conversion to dB in step (i) ensures that the PAD is relatively immune to small variations in band amplitude between input file types.

3. DESCRIPTOR RESILIENCE TO COMPRESSION NOISE

To evaluate the transparency of the PAD for locating audio segments regardless of compression effects, an MP3 [12] encoder was used to compress and uncompress 90, 16-bit 44.1kHz sampled audio files. Each of the audio files was of approximately 10 seconds duration, with 50 files representing instrumental signals (inst), and 40 representing combinational signals (comb) such as pop music. The MP3 encoder was chosen as it represents the most commonly used audio encoder and is also relatively old; hence it should contribute more severe compression distortion than more modern coders such as AAC or AC3.

The subset of MPEG-7 low level audio descriptors, defined in Section 2.1, and the proposed PAD were then calculated for both the files which had been compressed/uncompressed, and the original files. A frame size of 20ms was used to calculate the descriptors. The descriptors were then used to locate a specific frame in the compressed file, using the descriptors generated for the target frame from uncompressed data. The searching scheme selected the frame in the compressed file that minimized the Mean Squared Error (MSE), defined as:

$$MSE = \frac{1}{r}\sum_{n=1}^{r}(x_n - \overline{x_n})^2 \qquad (11)$$

where x_n is the descriptor from the original file, $\overline{x_n}$ is the descriptor from the compressed file and r is the dimension of the descriptor per frame. It should be noted that when the descriptors are generated from original uncompressed data, the above scheme returns the correct frame for all descriptors.

For testing purposes, the PAD was used with bandwidths of 1 bark (resulting in 25 values per frame) and 4.2 bark (6 values per frame), and the high edge for the SFM [8] was set to 16 kHz (resulting in 24 values per frame) and 708 Hz (6 values per frame). In addition, the complexity reduction described in [8], where spectral coefficients are grouped together in the SFM calculation for bands above 1 kHz was not used. This produces the most accurate representation of the SFM descriptor for comparative purposes.

The average results for the percentage of correct frames identified by each descriptor, for each file type, are shown in Table 1. The results in Table 1 indicate that the PAD using 1 bark wide bands (PAD25) produced the most reliable results. In fact, across the range of test files the PAD failed to correctly identify just 0.04% of frames. This represents a failure of only 4 frames in every 10000 and clearly supports the hypothesis that matching the descriptors' structure to the objective of the compression schemes should result in near transparent description.

The best performing of the MPEG-7 descriptors were the ASE and the SFM with an upper cutoff of 16 kHz (SFM24). While the error performance of these descriptors may be sufficient for some applications, they both fail at an average rate of over 1 frame in every 100. In comparison, for a 4 minute song encoded with 20ms frames, the PAD25 would on average fail identifying just 5 frames, whereas the SFM24 and ASE would fail identifying 175 and 158 frames respectively. In addition, whilst the PAD25 is consistent in performance for both instrumental and combinational files, the SFM24 and ASE both exhibit quite large discrepancies in performance across the different file types.

Comparing the performance of the PAD and SFM with only six values per frame, indicates that the PAD6 performs significantly better than SFM6. This result suggests that the PAD descriptor is truly scalable in dimension. This characteristic is important for applications where storage size or bit rate are limited (most applications of digital audio). The better performance of PAD6 when compared to SFM6 is most likely due to a combination of the descriptor's compression transparent nature and the fact that PAD6 continues to represent the entire

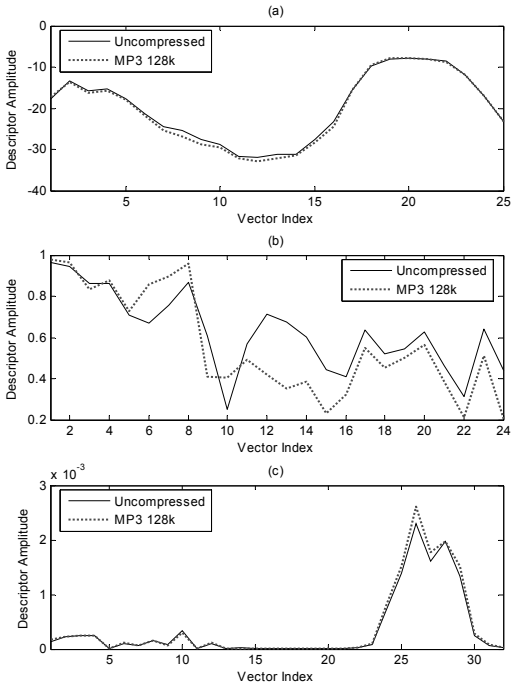

Figure 2: Descriptors for a single audio frame; (a) PAD25, (b) SFM24, (c) ASE

spectrum of the audio frame, while SFM6 represents only a limited portion of the spectrum.

The PAD25, SFM24 and ASE descriptors calculated for both the uncompressed and MP3 representations of a single 20ms audio frame are shown in Figure 2. Examining Figure 2 clearly indicates that for this frame, the SFM descriptor is the most effected by compression. The significant alteration of the SFM shape with the introduction of compression noise indicates that this descriptor may be relatively ambiguous between similar audio frames. In contrast, the PAD and ASE are relatively unaffected by the compression noise. However, when comparing the PAD and ASE the *y*-axis scale should be considered. It is evident that the absolute error is far larger for the ASE descriptor. This large error could prove detrimental to the descriptor's performance in the MSE search criteria used in (11).

4. CONCLUSIONS

A new low level audio descriptor is proposed. This descriptor represents the psycho-acoustic noise floor shape of an audio frame. Thorough testing indicates that the proposed descriptor is far more resilient to compression noise than any of the MPEG-7 low level audio descriptors. In addition, the proposed descriptor maintains high resilience to compression noise even when decimated to use only one quarter of the values per frame to represent the noise floor. This characteristic indicates the proposed descriptor presents a truly scalable mechanism for transparently describing the characteristics of an audio frame. The results presented support the hypotheses that developing a low level audio descriptor whose structure matches the underlying objective of the compression schemes is essential if transparency to compression is a requirement.

While the author's acknowledge the excellent search performance reported for the audio fingerprinting method presented in [9], it is felt that using a more robust low level descriptor than the MPEG-7 SFM could only improve the repeatability and robustness of such a system.

5. REFERENCES

[1] ISO/IEC JTC1/SC29/WG11N5525, "Mpeg-7 Overview (version 9)", International Organisation for Standardisation, Pattaya, March 2003.

[2] S. Chang, T. Sikora and A Puri, "Overview of the MPEG-7 Standard", IEEE Trans. On Circuits and Systems for Video Tech., Vol. 11, No. 6, pp. 688-695, June 2001.

[3] M. Casey, "MPEG-7 Sound Recognition Tools", IEEE Trans. On Circuits and Systems for Video Tech., Vol. 11, No. 6, pp. 737-747, June 2001.

[4] S. Quackenbush, A. Lindsay, "Overview of MPEG-7 audio", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 11 Issue 6, pp. 725-729, June 2001.

[5] J. Lukasiak, D. Stirling, S. Perrow, N. Harders, "Performance of MPEG-7 Low Level audio descriptors with compressed data", Proc. of ICME, Vol. 3, pp. 273-276, Baltimore, USA, July 6-9 2003.

[6] J. Lukasiak, D. Stirling, S. Perrow, and N. Harders, "Manipulation of Compressed Data Using MPEG-7 Low Level Audio Descriptors", Journal of Telecommunications and Information Technology, Vol. 2, pp. 83-91, June 2003.

[7] J. Lukasiak, D. Stirling, M. Jackson, N. Harders, "An examination of Practical Information Manipulation Using the MPEG-7 Low Level audio Descriptors", Proc. of 1st W/shop on the Internet, Telecommunications and Signal Processing, pp. 149-154, Wollongong, Australia, December 2002.

[8] ISO/IEC 15938.4:2003, "Information Technology Multimedia Content Description Interface, Part 4: Audio", International Organisation for Standardisation, 2002.

[9] J. Herre, O. Hellmuth and M. Cremer, "Scalable Robust audio Fingerprinting using MPEG-7 Content Description", IEEE Workshop on Multimedia Signal Processing, pp.165-168, Dec 2002.

[10] J. Johnston, "Transform Coding of Audio Signals Using Perceptual Noise Criteria," IEEE J. Sel. Areas in Comm., pp.314-323, Feb. 1988.

[11] ISO/IEC, JTC1/SC29/WG11 MPEG, "Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbits/s-Part3: Audio", IS11172-3 1992.

[12] T. Painter and A. Spanias, "Perceptual Coding of Digital Audio", Proc. of IEEE, Vol. 88, Issue 4, pp. 451-515, April 2000.