ERROR CONCEALMENT OF MPEG-2 AAC AUDIO USING MODULO WATERMARKS

Samuel Cheng[1] * *Heather Yu*[2], *and Zixiang Xiong*[1]

[1]Dept of Electrical Engineering, Texas A&M University, College Station, TX 77843, USA
[2]Panasonic Information and Networking Technologies Lab, Princeton, NJ 08540, USA

ABSTRACT

We propose an error concealment scheme for MPEG-2 compressed (AAC) audio using a novel modulo watermarking technique. It can be used on top of other error control schemes. After the modulo watermark is embedded, an MPEG-2 AAC audio only shows negligible file size increase and moderate SNR penalty. For audio transmission over packet-switch networks (e.g., the Internet), using our watermark-based concealment scheme shows consistent SNR gain over using conventional concealment schemes.

1. INTRODUCTION

Reliable transmission of digital audio over packet-switched networks (e.g., the Internet) is a challenging task because the Internet is a best-effort network that offers no QoS guarantee. Although channel coding can be used to protect the audio from packet loss, it usually introduces extra redundancy/payload. On the other hand, error concealment [1, 2, 3, 4], which typically extracts features from the received audio and uses them to recover the lost data, is very attractive in audio transmission as it improves the perceptual quality without the need of additional payload.

There are two issues in error concealment: complexity of the receiver and inaccurate extraction of enhancement features at the decoder. Both can be addressed by extracting the features at the encoder and transmitting them to the decoder along with the audio. However, this method has the same disadvantage as using channel coding in that an extra payload is required. This extra payload not only uses up more bandwidth, but necessarily modifies the audio format if neither a common area nor a user data area is available. This format change makes the audio no longer decodable by an ordinary decoder.

In this work, we apply data hiding techniques [1] to embed these enhancement features for error concealment of MPEG-2 AAC audio [5, 6]. Specifically, a novel modulo watermarking scheme is deployed for the *first* time to hide the enhancement features. Modulo watermarking, which extracts the hidden data as the modulo of the sum of a watermarked integer signal samples, is an example of one-to-many embedding schemes. In other words, several different watermarked signals can contain the same hidden data. This property gives the watermark encoder freedom in selecting a watermarked signal with small perceptual distortion.

Portions of the AAC encoded audio such as audio headers are naturally more important than the others. When the encoded audio is transmitted via a noisy channel (e.g., the Internet), unequal error protection is usually applied to ensure almost no corruption on these portions. In this work, we assume the headers are very well protected and can be fully recovered. However, frequency coefficients, which are less important, may be lost during transmission. When this happens, we extract the enhancement features from embedded watermarks and use them for error concealment.

After the modulo watermark is embedded, an MPEG-2 AAC audio only shows negligible file size increase (< 0.1 %) and moderate SNR penalty (< 0.7 dB). For audio transmission over packet-switch networks, using our watermark-based concealment scheme shows consistent SNR gain over using conventional concealment schemes with zero replacement or the frame duplication.

2. AAC ERROR CONCEALMENT

2.1. Advanced Audio Coding (AAC)

AAC [6], which is included in the MPEG-2 audio standard, is the non-backward compatible successor of MPEG-1 Layer 3 audio coding (MP3). AAC encoding consists of four steps: frequency transform, quantization, entropy coding, and bitstream multiplexing. AAC employs modulated discrete cosine transform (MDCT) typically with 1024 samples per frame. The 1024 frequency samples in each time frame are separated into 49 frequency bands. Within the same frequency band, samples are considered to have similar perceptual effect to the human ears and hence share the same quantization step size. Perceptual modeling is applied to the MDCT coefficients to estimate the maximum amount of distortion that can be withstood by each coefficient. The quantization step size is iteratively modified until both the rate is below the target bit rate and the distortion is below the maximum acceptable value obtained from the perceptual model. Huffman coding is used to encode the quantized co-

*Work done when the first author was an intern at the Panasonic Information and Networking Technologies Lab.

0-7803-7448-7/02/$17.00 ©2002 IEEE

efficients and the quantization step size. Finally, the coded indices will be multiplexed into a single bitstream.

2.2. Proposed Error Concealment Scheme

Since a coefficient is most effectively estimated by its nearest neighbors, ideally, adjacent coefficients along both time and frequency axes should not be packed together, because the sources of estimation will be lost as well when the packet is dropped. However, we do not impose this as a requirement of our scheme, because we target at overlaying our scheme on any other error control scheme [7].

As coefficients inside a frequency band share similar perceptual behavior, we choose to group them together for estimation. Denote (n,i)-band as the the i^{th} band at the n^{th} time frame and assume coefficients $b[n,k]$ in (n,i)-band are lost, where $k \in \mathcal{K}_i$ and and $\mathcal{K}_i$ is the index set of the i^{th} band. We estimate $b[n,k]$ as $\hat{b}_0[n,k] = 0, \hat{b}_1[n,k] = b[n-1,k], \hat{b}_2[n,k] = b[n+1,k]$, or $\hat{b}_3[n,k] = \frac{1}{2}(b[n-1,k] + b[n+1,k])$.

For each of the above four choices, we define $c[n,i]$ as the index that minimizes the mean square error. That is,

$$c[n,i] = \text{argmin}_{c\in\{0,1,2,3\}} \sum_{k\in\mathcal{K}_i} (b[n,k] - \hat{b}_c[n,k])^2.$$

$c[n,i]$ is pre-computed and embedded into the original AAC audio. Embedding $c[n,i]$ into the (n,i)-band itself will not work because when we need this information, the band is lost, so is $c[n,i]$. We split $c[n,i]$ into two bits and embed them separately in the two neighboring bands.

Define

$$d[n,i] = \begin{cases} 0, & \text{if } c[n-1,i] \in \{0,1\} \wedge c[n+1,i] \in \{0,2\}, \\ 1, & \text{if } c[n-1,i] \in \{2,3\} \wedge c[n+1,i] \in \{0,2\}, \\ 2, & \text{if } c[n-1,i] \in \{0,1\} \wedge c[n+1,i] \in \{1,3\}, \\ 3, & \text{if } c[n-1,i] \in \{2,3\} \wedge c[n+1,i] \in \{1,3\}. \end{cases}$$

The higher and the lower bit of $d[n,i]$ tell whether the current band is suitable for estimating the band in the next time frame ($(n+1,i)$-band) and in the last time frame ($(n-1,i)$-band), respectively.

For example, suppose the (n,i)-band is lost, from the lower bit of $d[n+1,i]$ and the higher bit of $d[n-1,i]$, we can determine whether the current band should be estimated from any of its neighbors. When it is estimated from both sides, it is scaled by 1/2. If one of its neighbors is lost, we estimate the current band from the remaining neighbor. If both neighbors are loss, then we assume $c[n,i] = 0$ and replace the coefficients by zeros.

3. WATERMARK EMBEDDING OF ENHANCEMENT INFORMATION

3.1. Choices of Watermarking Schemes

Watermarking schemes [8, 9, 10] can be categorized into two classes: robust watermarking [9] and fragile watermarking [11, 12]. Robust watermarking is designed to withstand common signal processing attacks (< 10 bits/sec for audio), while fragile watermarking is sensitive to any modifications but has a much higher embedding rate (~ 1000 bits/sec for audio).

Since there are two bits for each $d[n,i]$ and one $d[n,i]$ per band, for a dual channel audio clip with sampling rate 44100 Hz, the embedding rate is about $44100/1024 \times 49 \times 2 \times 2 \doteq 8$kbits/sec, which is too high for robust watermarking. Therefore, fragile watermark is the only possible option.

A typical fragile watermarking scheme is least bit modulation (LBM). One can embed a bit into a host signal by simply replacing the least significant bit of one signal sample by the embedding bit. The information embedding rate of LBM can be very high. For example, if we embed a bit into each sample of dual channel audio with a sampling rate of 44100 Hz, the embedding rate is up to $44100 \times 2 \doteq 80$kbits/sec in theory. However, since only the least significant bit is modified, the watermark can be removed easily by truncating the last bit. Fortunately, unlike dealing with copyright protection applications, deliberate attacks to our watermark is not likely.

3.2. Fragile Modulo Watermarking

Since different signal samples may have different susceptibilities to distortion, we should adaptively select the embedding locations. However, for LBM, both the encoder and the decoder have to agree upon a predefined embedding locations, because there is no side-information in telling the decoder the embedding locations. Note that it may not be a problem for some other applications in which a key is available for decoding, because the key itself can serve as the side-information. However, for the error concealment problem, it is not reasonable to require a user to provide a "key" before enhancement is performed.

To enable flexible encoding, we propose a novel fragile watermarking technique that does not require the decoder to have the knowledge of the exact embedding locations. Let $\mathbf{x} = x_1, x_2, ...x_N$ be an arbitrary integer host signal sequence. We embed an integer $k \in [0, K]$ by enforcing the following:

$$\sum_{i=1}^{N} x_i \equiv k \mod K.$$

Note that LBM is a special case of modulo watermarking when $N = 1$ and $K = 2$.

There is more than one possible watermarked signal containing with the same embedded information. The encoder has the freedom of choosing locations of modifications so that the watermarked signal is perceptually closest to the original. Despite that, the decoder does not really need to know these locations where modifications have been made.

3.3. Enhancement Information Embedding using Modulo Watermarking

One limitation in applying our fragile watermarking is that it can only be deployed after quantization, otherwise the watermark will be destroyed. Moreover, since it is very hard to embed watermark into a Huffman coded signal, we embed the enhancement features into the quantization indices, which are obtained after partial decoding. After watermarking, the modified indices will be encoded using Huffman coding with the original codebook.

With the freedom of embedding given by modulo watermarking, the question left is what indices and by how much they should be modified. Ideally, this can be done by applying perceptual modeling to the original audio. For example, if we know one coefficient can afford a distortion of 10 units and its current quantization step size is 2 unit. Then we know that we can approximately vary the corresponding index by 5 steps without affecting the quality [1].

However, the perceptual model may not be accessible, because the file has already been compressed. Although we can estimate the model parameters from the decompressed audio, the estimation is not accurate in general. Therefore, we employ a heuristic approach as follows without using the perceptual model:

To embed $d[n,i]$ into the quantization indices $q[n,k]$ of (n,i)-band, $k \in \mathcal{K}_i$, let $l \equiv \sum_{k\in\mathcal{K}_i} q[n,k] - d[n,i] \mod K$, where K is the number of different values that can be embedded.

In this work, we pick K as four. Assume $0 \le l < K/2 = 2$, we embed modulo watermark $d[n,i]$ in the following three steps:

1. Among all indices that lie within range $[I_{\min}, I_{\max}]$, select the l largest in magnitudes.
2. Declare embedding failure and leave indices unchanged if less than l indices can be found in step 1.
3. Subtract each of those indices by 1.

If $4 > l \ge 2$, replace l as $4 - l$ and proceed all steps except modifying the last one with addition instead of subtraction.

Since the enhancement features (d's) are independently stored, they are useful even when only a fraction of them is retrieved correctly. Therefore, embedding failure in the scheme is acceptable.

The lower limit $I_{\min}$ in the first step restrains modification of small value indices, because they are more probable to have high susceptibility to distortion. In particular, no distortion should be imposed on zero indices. $I_{\min}$ also serves as a design parameter in trading error free distortion with error concealment capability. As $I_{\min}$ increases, it is more likely that the embedding of $d[n,i]$ fails and leaves the indices with no distortion. However, inaccurate $d[n,i]$'s will make the error concealment process less efficient. In our experiment, $I_{\min}$ is simply set to be 1.

$I_{\max}$ is equal to the maximum possible value available in the Huffman table less 1. This prevents indices from being out of bound after modification. Large indices are selected for modification because they can withstand larger distortion.

4. EXPERIMENTAL RESULTS

4.1. Increase of Audio File Size After Watermark Embedding

The Huffman codebook used in the original audio is optimized in the AAC encoder. Since we modify the indices but keep the old codebook, it is expected the size of the compressed file will increase after watermark embedding. However, the increase should be small because we only change relatively few indices. Table 1 indeed confirms this – the size increase is less than 0.1 % over all test audio clips.

In contrast, from Section 3.1, we need 8 kbits/sec if an explicit overhead is written to the audio. This corresponds to 8/256=3 % of total file size for an audio encoded at 256 kbits/sec.

clip1	clip2	clip3	clip4	clip5	clip6	clip7
0.02%	0.02%	0.06%	0.01%	0.03%	0.06%	0.06%

Table 1: Percentage change in audio clip size after watermarking.

4.2. Audio Quality after Watermark Embedding

After modulo watermarks are embedding into an AAC audio file, we expect the quality of the decoded audio clip to deteriorate somewhat. However, our test shows that the perceptual quality of the watermarked audio clips is acceptable in office or lab environment. As an objective measure, we compare the SNR difference of each AAC coded audio clip before and after watermark embedding. The SNR decrease due to watermark embedding is between 0.03 dB and 0.68 dB (Table 2).

4.3. Error Concealment Results

We assume the AAC audio coefficients are packetized and transmitted via a noisy channel. Each packet consists of coefficients from one time frame. A packet is either correctly

[1] Uniform quantization is assumed in this simplified example.

	AAC audio	After watermarking	SNR changes
clip1	32.87	32.69	0.18
clip2	18.18	17.95	0.23
clip3	17.13	17.10	0.03
clip4	31.50	31.29	0.21
clip5	28.66	27.99	0.67
clip6	24.47	23.79	0.68
clip7	26.73	26.69	0.04

Table 2: SNR change (in dB) after embedding enhancement information.

		Packet loss ratio				
		0.01	0.02	0.05	0.1	0.2
clip1	Ours	22.79	20.99	15.80	13.25	9.91
	Ref.1	20.92	16.99	12.90	10.01	6.74
	Ref.2	18.60	15.06	10.63	7.61	4.24
clip2	Ours	16.93	15.94	13.92	11.80	9.49
	Ref.1	16.01	14.56	11.87	9.47	6.82
	Ref.2	15.02	13.01	9.90	7.20	4.42
clip3	Ours	16.12	15.23	13.06	11.16	8.65
	Ref.1	15.73	14.39	11.81	9.50	6.87
	Ref.2	14.41	12.49	9.36	6.71	3.92
clip4	Ours	23.74	19.62	15.27	12.42	9.55
	Ref.1	20.64	17.37	12.88	9.99	6.98
	Ref.2	17.18	14.22	10.15	7.25	4.09
clip5	Ours	23.93	21.20	14.91	12.63	9.30
	Ref.1	22.17	18.75	12.73	10.35	6.92
	Ref.2	19.35	15.08	10.13	7.67	4.53
clip6	Ours	20.73	18.82	16.81	13.62	10.59
	Ref.1	19.99	17.06	13.17	10.57	7.19
	Ref.2	16.73	14.19	9.18	6.61	3.19
clip7	Ours	23.33	21.10	15.19	13.26	9.87
	Ref.1	20.07	17.46	12.16	9.97	7.05
	Ref.2	18.82	15.87	8.59	6.26	3.36

Table 3: SNR comparison (in dB) of three different error concealment schemes: our scheme (upper), zero replacement scheme (middle), blindly duplication from previous time frame (lower).

received or lost. A periodic packet loss is assumed in our simulation with a fixed packet loss ratio. We compare our scheme with two reference schemes (Ref.1 [13] and Ref.2 [14]). In Ref.1, all lost coefficients are set to 0. In Ref.2, the previous adjacent time frame is copied to the current lost one (Table 3).

Our watermark-based concealment scheme gives higher SNR than Ref. 1 and Ref. 2 in all cases. The slight drop in SNR due to watermark embedding is quickly offset by the gain obtained from our concealment scheme even at a small packet loss ratio of 0.01. Moreover, the gain is more conspicuous as packet loss ratio increases.

5. CONCLUSION

We have proposed an error concealment scheme for AAC audio using digital watermarking, which can be overlaid on other error control schemes effectively. A novel modulo watermarking technique is described and incorporated into our scheme. After the modulo watermark is embedded, an MPEG-2 AAC audio only shows negligible file size increase and moderate SNR penalty. For audio transmission over packet-switch networks, using our watermark-based concealment scheme shows consistent SNR gain over using conventional concealment schemes. Although simulations are done on AAC audio in this paper, our scheme can be easily extended to other media formats.

6. REFERENCES

[1] P. Yin, H. Yu, B. Liu, "Error Concealment Using Data Hiding," *Proc. IEEE ICASSP'01*, May 2001.

[2] B. W. Wah, X. Su, and D. Lin, "A survey of error-concealment schemes for real-time audio and video transmissions over the internet," *In Proc. Int'l Symposium on Multimedia Software Engineering*, pp. 17-24, Dec 2000.

[3] K. Cluver and P. Noll, "Reconstruction of missing speech frames using sub-band excitation," *IEEE-SP Int'l Symposium on Time-Frequency and Time-Scale Analysis*, pp. 227-280, June 1996.

[4] Y.L. Chen and B.S. Chen, "Model-Based Multirate Representation of Speech Signals and Its Application to Recovery of Missing speech Packets," *IEEE Trans. Speech and Audio Processing*, vol. 5, no. 3, pp. 220-231, May 1997.

[5] M. Bosi, K. Brandenburg, Sch. Quackenbush, L. Fielder, K. Akagiri, H. Fuchs, M. Dietz, J. Herre, G. Davidson, and Yoshiaki Oikawa, "ISO/IEC MPEG-2 Advanced Audio Coding," *Journal of the Audio Engineering Society*, Vol. 45, no. 10, pp. 789-814, Oct 1997.

[6] MPEG-2 advanced audio coding, International Standard IS 13818-7, ISO/IEC JTC1/SC29 WG11, 1997.

[7] P. A. Chou, A. E. Mohr, A. Wang, and S. Mehrotra, "Optimal error control for receiver-driven layered multicast of audio and video," *IEEE Transactions on Multimedia*, vol. 3, pp. 108-122, March 2001.

[8] L. Boney, A. Tewfik, and K. Hamdy, "Digital watermarks for audio signals," *Proc. ICMCS'96*, pp. 473–480, Hiroshima, Japan, June 1996.

[9] I. Cox, J. Killian, F. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, pp. 1673-1687, Dec 1997.

[10] I. Cox, et. al., "Digital Watermarking," Morgan Kaufmann Publishers, 2001.

[11] M. Yeung, F. Mintzer, "Invisible watermarking for image verification," *Journal of Electronic Imaging*, Vol 7, No 3, pp 578-591, 1998.

[12] C. Wu, et. al., "Fragile imperceptible digital watermark with privacy control," *SPIE Electronic Imaging'99: Security and watermarking of multimedia content*, vol. 3657, Jan. 1999.

[13] J. Gruber, L. Strawczynski, "Subjective effects of variable delay and clipping in dynamically managed voice systems," *IEEE Trans. Communication*, 33-8, pp. 801-808, Aug 1985.

[14] ETSI, "Substitution and muting of lost frames for full rate speech channels," Recommendation GSM 6.11, 1992.