Data Embedding In Audio Signals[1]

K. Gopal Gopalan
Department of Engineering
Purdue University Calumet
2200 169th Street
Hammond, IN 46323
219-989-2898
gopalan@calumet.purdue.edu

Daniel S. Benincasa, Stanley J. Wenndt
Multi-Sensor Exploitation Branch
Air Force Research Laboratory
32 Brooks Road
Rome, NY 13441
315-330-3555, 315-330-7244
benincasad@rl.af.mil, wenndts@rl.af.mil

Abstract - This paper presents results of two methods of embedding digital audio data into another audio signal for secure communication. The data-embedded, or stego, signal is created for transmission by modifying the power spectral density or the phase spectrum of the cover audio at the perceptually masked frequencies in each frame in accordance with the covert audio data. Embedded data in each frame is recovered from the quantized frames of the received stego signal without synchronization or reference to the original cover signal. Using utterances from Texas Instruments Massachusetts Institute of Technology (TIMIT) databases, it was found that error-free data recovery resulted in voiced and unvoiced frames, while high bit-errors occurred in frames containing voiced/unvoiced boundaries. Modifying the phase, in accordance with data, led to higher successful retrieval than modifying the spectral density of the cover audio. In both cases, no difference was detected in perceived speech quality between the cover signal and the received stego signal.

TABLE OF CONTENTS

1. INTRODUCTION

Data embedding is a form of steganography that is concerned with ways of inserting a given secret message or data in an innocuous cover message, such as an image, video, audio, or computer code [1, 2]. Digital data embedding in audio signals has many applications. These applications include covert communication by securely hiding encoded/encrypted information in audio signals, copyright protection of transmitted audio signals, and embedding information for describing, modifying, and tracking of audio signals. By providing different access levels to the embedded data, the quality of the audio signal and the ability to hear the hidden message can be controlled. Transmission of battlefield information via an auxiliary or cover audio signal could play an essential role for the security and safety of personnel and resources. This paper presents the initial results from a study of two basic techniques for embedding binary data in audio signals.

2. REVIEW OF DATA HIDING IN AUDIO

Most of the work in data hiding has been concentrated on hiding a small amount of information such as copyright data or a watermark in images and video segments [2-5]. However, general requirements, challenges and principles of hiding data in an audio are the same as those for embedding information in a video. Robustness of the hidden data, for example, is a key requirement for successful embedding and retrieval of the data. In other words, standard signal processing operations, such as noise removal and signal enhancement, must not result in loss or degradation of the embedded information. Additionally, for covert communication, the embedded information must withstand channel noise and intentional attacks or jamming on the signal. Also important in covert communication is the resilience of the hidden information to stay hidden to pirates during their intentional or unintentional attempts at detection. A measure of effectiveness of data embedding is the probability of detection of hidden data. Clearly the more robust the host medium – image, video, or audio – to attacks and common operations, the higher would be its effectiveness.

[1] U.S. Government work not protected by U.S. copyright.

Additional requirements specific for embedding data in audio signals vary with the applications. In general, the embedded data must be perceptually undetectable or inaudible. While this may not be strictly required or even needed for watermarking of audio for browsers on the Internet, covert communication calls for the hidden message to be truly imperceptible. Tamper resistance of the hidden message, on the other hand, is more crucial in battlefield covert communication than in protecting ownership of the cover audio. Additionally, extraction of the hidden message must not require access to the host (cover) audio. Clearly, lack of the original host signal that was used to embed the message makes it difficult to extract and adjudge the quality and quantity of the hidden data. For covert communication, however, this challenge must be met even at the cost of degraded quality of the message-embedded audio. Other requirements, such as robustness to transmission channel noise, and linear and nonlinear filtering, are also important in hiding data in audio. Security requirements in covert communication dictate that an unauthorized user must not be able to detect the presence of hidden data unless he has the key to the insertion of data. This may require encryption of the data prior to its insertion in the host audio.

Some of the most common techniques for hiding data in images employ the properties of human visual system. The least significant bits of an image may be altered in accordance with the data to be embedded, for example [4, 6]. The technique in this case relies on the low sensitivity of the human visual system to contrast. Variations of this technique include embedding pseudo random noise sequence that appears as quantization noise, and modifying the Discrete Cosine Transform (DCT) or wavelet transform coefficients, etc. for watermarking. Other methods also exploit imperceptible brightness levels to add tags, identification strings, etc. More recently, spread spectrum techniques, in which the watermark to be embedded in an image is spread throughout the spectrum of the image, have been widely considered [2, 3, 7, 8]. For video, blue color has been used to embed watermark based on the least sensitivity of human visual system to modifications in the blue band.

The notion of creating an imperceptible data-embedded image based on the human visual system threshold has been extended by several researchers to embed data in host audio [4, 7-10]. In general, the procedure exploits the frequency and temporal masking properties of the human auditory system (HAS) to modify the cover audio in such a way that changes due to the embedded data are inaudible. Other methods to watermark a host audio use replacement of spectral components in the high, middle, or other pre-selected frequency bands in accordance with the sequence to be embedded. In addition, several techniques involving the use of spread spectrum noise sequence have been reported. By far the methods employing the psychoacoustical masking properties of HAS in some form appear to better meet the challenges and requirements of audio data embedding. The next section outlines the basics of determining the frequency masking thresholds for a given audio. Following this outline, two methods used to embed and recover data and their results are described.

3. Psychoacoustical Masking Thresholds of Hearing

Auditory masking is a perceptual property of the human auditory system in which the presence of a strong tone renders the hearing of a weaker tone in its temporal or spectral neighborhood imperceptible. Also, a pure tone is masked by a wide-band noise if the tone occurs within a critical band. Frequency masking is based on the observation that the human ear cannot perceive frequencies at lower energies when these frequencies are present in the vicinity of tone- or noise-like frequencies at higher energies. Temporal masking occurs in which a low-level tone becomes undetected when it appears immediately before or after a strong tone. Many psycho-acoustic experiments have been reported to verify the spectral and temporal masking phenomena [11-15]. The design of high quality audio coders, such as Moving Picture Experts Group (MPEG) coders, is based on the property of the psychoacoustical model [16-20]. As with the design of coders, the masking phenomenon can be used to embed data in an audio with negligible perceptual difference between the original, unembedded audio and the data-embedded audio. The general procedure used is shown in Figure 1.

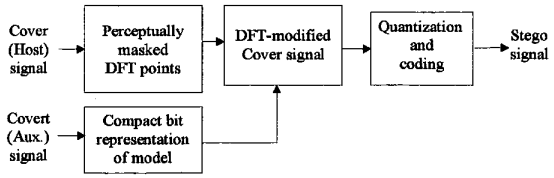

Figure 1 - General procedure for embedding data in perceptually masked locations

The first step in exploiting the masking property for coding or data embedding is to determine the masking threshold level. For an utterance of speech the masker frequencies – tonal and noise-like – and their power levels are computed from frame to frame. A global threshold of hearing based on the maskers is determined for each frame. Also, the sound pressure level for quiet, below which a signal is inaudible, is obtained. The complete procedure is described in the ISO International Engineering Consortium MPEG audio coder implementation standards, which is also presented in [17-20].

As an example, Figure 2 shows the normalized power spectral density (PSD), absolute quiet threshold, and threshold of hearing for a frame of utterance. The lowest spectral component around 2800 Hz in this figure, for instance, indicates that this component, being below the

masking threshold level at that frequency, cannot be perceived in hearing. We notice that with the threshold at approximately 65 dB and the PSD at 32 dB, raising the PSD of the signal at 2800 Hz by as much as 30 dB will still render the component inaudible. Many other such 'psychoacoustical perceptual holes' can be detected in several frequency ranges. The PSD values at these holes can be modified by information to be embedded without affecting the message quality of the frame. This is the basis used in the present work for embedding data in audio.

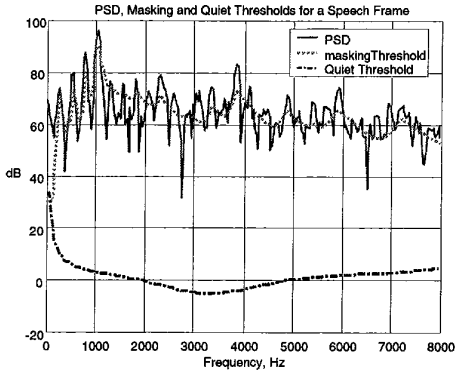

Figure 2 - Power spectral density, global masking threshold for a frame of speech and absolute quiet threshold

4. Data Embedding Procedures

Experiments were conducted to determine the audibility of embedded tones at masked frequencies. In the first experiment, the threshold for a single tone (sinusoid) at a given power level was determined. Several tones were added to the original sinusoid at various amplitudes and verified their audibility as a function of the amplitude and frequency of the original tone. The experiment was then extended to embedding tones in voiced speech. Continuant sounds – vowels /i/ and /ae/ - were used as cover audio to embed tones. Depending on the level and frequency of the tones, the masking ability of the continuant sounds was verified. Although only the spectral magnitudes of the "cover" vowel sounds were used to hide the "covert" tones, the experiments demonstrated the possibility of embedding imperceptible tones to represent concealed data.

Two methods of embedding were considered for experimental verification, both based on the psychoacoustical masking. The first method altered the magnitude of the power spectrum at the perceptual holes of each frame of a host speech utterance; the second method modified the phase of the utterance. For the purpose of studying the feasibility of these methods, a random set of 10 bits each was used as data to be embedded in each frame of an utterance from the Greenflag database and an utterance from the TIMIT database.

For both methods, the procedure begins with the calculation of the power spectral density and the global masking threshold using tone and noise maskers present in each frame of speech. Utterances in the databases were obtained at a sampling rate of 16 kHz with 16 bits per sample. Frames of 512 samples are used with Hanning (raised cosine) window. Power spectral density, normalized to 96 dB, is obtained using a 512–point Discrete Fourier Transform (DFT). Power normalization enables the use of the same masker spreading function at each frequency. Absolute quiet threshold, based on young listeners with acute hearing and given by

$$T_Q(f) = 3.64\left(\frac{f}{1000}\right)^{-0.8} - 6.5e^{-0.6\left(\frac{f}{1000}-3.3\right)^2} + 10^{-3}\left(\frac{f}{1000}\right)^4 \qquad (1)$$

where f denotes frequency in Hz, is calculated. Following the procedure given in [19, 20], frequency maskers based on tones and wideband noise in each critical band, and the global masking threshold $T_G(k)$ at each frequency index k are calculated for each frame. From these values the perceptual holes or frequency indices {k} such that

$$T_Q(k) < P(k) < T_G(k) - 5 \qquad (2)$$

are determined. If there are at least 10 frequency indices at which the PSD of a frame is down by at least 5 dB from the corresponding masking threshold values, but above the quiet threshold, that frame is considered suitable for data embedding. Because of the relatively high quiet threshold levels at low and high frequencies (below 100 Hz and above 7000 Hz) only, the holes in the range of 100 – 6000 Hz are used. (Avoiding high frequency range for spectral modification also retains the embedded data when speech is low-pass filtered or otherwise reduced in bandwidth for compression or coding.)

Modulation of Frame PSD by Data - In this method, the PSD values { P(k) } of a frame with 10 or more perceptual holes are modified to { P′(k) } by the data bits { b(k), k = 1, … 10 } as follows.

$$P'(k) \approx \begin{cases} 0.3T_G(k), & if \quad b(k) = 0 \\ 0.7T_G(k), & if \quad b(k) = 1 \end{cases} \qquad (3)$$

(The approximation above results from the normalization of PSD to a fixed value of 96 dB, which causes a different power scale factor, added to each frame.) If a frame has more than 10 locations satisfying Eq. (2), the PSD values at locations above the first 10 are set to the minimum of the global threshold value for that frame. This reduces the possibility of channel noise, for example, raising the PSD values at the receiver to values comparable to those at the data-embedded locations. After making the modified PSD values and the phase angles of the discrete Fourier transform of the frame symmetrical, the frequency spectrum of the data-embedded frame is inverted to obtain the time domain

samples for the modified frame. The samples are then quantized to 16 bit integers for transmission.

At the receiver, the 16 bit integers for the frame are processed to obtain the masking threshold and the PSD. Allowing for changes in the PSD, due to quantization, the embedded data { d(k), k = 1 … 10 } are recovered as

$$d(k) = \begin{cases} 0, & 0.2T_g(n) < P_r(n) < 0.4T_g(n) \\ 1, & 0.6T_g(n) < P_r(n) < 0.8T_g(n) \end{cases} \qquad (4)$$

where { n } are the frequency indices at which P_r, the received signal PSD values are above the quiet threshold but below the masking threshold by at least 5 dB.

Test results - As an illustration, the speech frame shown in Figure 2 are modified at the transmitter after determining the perceptual null indices. With 75 locations satisfying Eq. (2), the first 10 are used for data embedding. These 10 indices correspond to frequencies

f_h = 343.8, 375.0, 593.8, 625, 843.8, 875, 906.3, 937.5, 1093.8, and 1125 Hz,

At these frequencies, the power spectrum was modified by a random 10-bit combination in accordance with Eq. (3).

The PSD values at the remaining 65 locations were set to the minimum threshold of masking without the dB normalization scale factor. From the modified spectral density values and the original phase of the unmodified frame, time samples are obtained by inverse DFT. These time samples are quantized to 16 bits for transmission. Figure 3 shows the power spectrum of the received frame with data embedded, as above, in the frame shown in Figure 2. The corresponding time samples of the cover audio frame (unmodified) and the received (data-embedded) frame are shown in Figure 4.

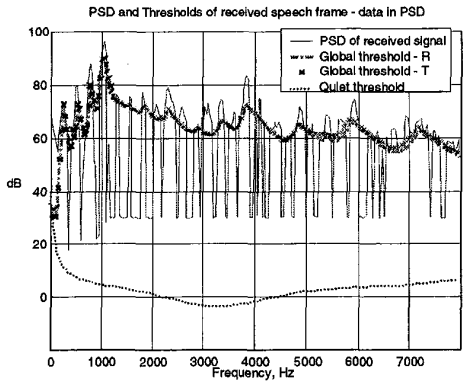

Figure 3 - Power spectrum of the received frame with data embedded in PSD in the frame shown in Figure 2

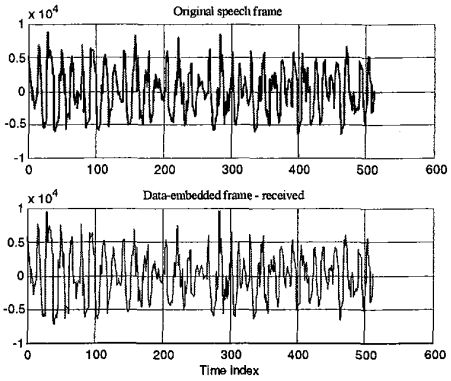

Figure 4 - Time samples of a frame before and after embedding data in the PSD

As can be seen from the two figures, modification to the frame PSD does not appear to significantly alter the time domain waveform. The real test, however, is in the listening of the modified utterance and its perception. With the modified PSD retaining the spectral densities below the masking threshold at the frequencies altered, perceptual quality of the utterance must remain as that of the original cover audio. This was verified in informal listening tests for the Greenflag utterance, ckd133.16 - "Rueben seven-one cleared take-off pushbutton five" (male). See Figure 5. For this utterance, with a total of 55 frames, 10-bit random data was successfully embedded and retrieved from each of the 52 frames using 64-bit (unquantized) stego samples for transmission. Using 16-bit representation of the stego signal, however, resulted in only 45 frames correctly recovering the data. Two frames in each case could not be used for data insertion because of fewer than 10 masked frequency points available in those frames.

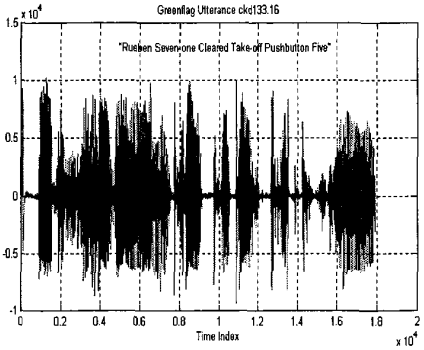

Figure 5 - Time samples of Greenflag utterance ckd133.16

Tests on two TIMIT utterances – from a male and a female speaker – showed similar results with no noticeable difference in the perceptual quality of the stego signals. Table 1 shows the results for the three utterances used in the tests.

Table 1. Number of frames with successful data retrieval Vs. Total number of frames

Utterance →	ckd133 – Greenflag	Female_sa1 - TIMIT	Male_sa2 - TIMIT
Success Rate with 64 bits	52/55	155/190	134/148
Success Rate with 16 bit quantization	45/55	70/190	66/148
Number of Unembedded Frames	2/55	1/190	2/148

Use of 64 bits for transmission, as is the case with Matlab processing, clearly results in better data retrieval than quantizing the stego signal to 16 bits before transmission. This indicates that the PSD may have been modified in the lower end of the 64-bit representation. When quantized to 16 bits, therefore, changes made to the PSD values are truncated. A different set of thresholds with a larger difference for embedding bits 1 and 0 may result in more significant bits being modified, and hence, survive the truncation to 16 bits.

Another possible reason for the low recovery is that a large majority of the frames in both utterances are unvoiced – "we had your dark suit in greasy washwater all year" (female) and "don't ask me to carry an oily rag like that" (male). With wideband noise-like spectrum for an unvoiced frame, there are fewer masking points, and these points change with quantization. The problem is more pronounced in frames containing transitions between voiced and unvoiced frames.

Modulation of Frame Phase by Data - This method is based on the observation that, in general, the phase spectrum can be altered at perceptually masked spectral points. While this change in phase modifies the waveform, perceptual quality of speech is not affected, particularly if the phase change occurs in a midband of frequencies. Based on this, Tilki and Beex reported encoding of data bits by altering the phase of every fourth point (after 2 kHz) in a 2048-point DFT by $\pm(\pi/8)$ radian relative to a reference point phase [9]. With this differential phase change, the authors reported successful encoding and decoding for storage media requiring simple synchronization.

Instead of differential phase change, the present method alters the absolute phase at masked spectral points. This ensures that changes in time samples are rendered inaudible. Also, with absolute phase change of $\pm\phi$, no synchronization is needed at the receiver. Figure 6 shows the phase spectra of a frame of speech at the first 10 perceptually masked locations for an utterance from the TIMIT database. With ±45° phase, the embedded data for the frame corresponds to

b = [0 0 0 1 1 1 0 1 1 0]

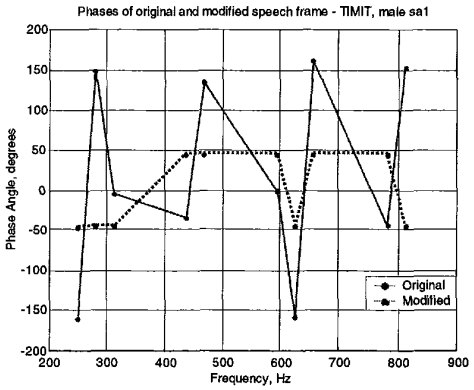

Figure 6 - Phase spectra at the 10 masked locations of the original and data-embedded frame

At other masked locations the magnitude and phase are left unchanged. Time samples of the original and the phase modified (received) frames are shown in Figure 7.

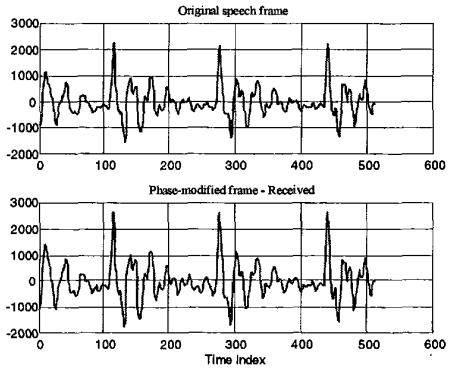

Figure 7 - Original and data-embedded (phase-modified) frames of speech

Test result - The method worked well with phase changes as low as ±5° with consistent recovery of embedded data in voiced frames. Recovery from frames that are entirely unvoiced resulted in a few bit errors. Embedded frames that contained a transition between voiced and unvoiced speech or those that were extremely small in amplitude – as with silence and the onset of plosives – caused severe errors in retrieved bits. For a noisy speech such as from the Greenflag database, with significantly large amplitudes even

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.