DATA EMBEDDING IN AUDIO SIGNALS USING PERCEPTUAL MASKING

by: Jacob Chernoi
& Dror Friirich
Supervisor: Ariel Sagi

Technion - Israel Institute of Technology
Department of Electrical Engineering

Project A
Summer 2004
Project Goals

- Embedding data, onto a given audio signal. Writing the embedded signal onto an audio CD, and detecting the data with low error.
- Achieve a high as possible embedding rate.
- Maintain the original sound quality and minimize detection error for the embedded data.
Implementation

MTF-masking threshold function

- According to MPEG’s Layer1 Psycho-Acoustic Effect Model, given an audio signal, it is possible to calculate the MTF.
- Any sound spectrally weaker than the MTF, will be undetected by a human ear.
- The MTF in silence is also known as the human hearing threshold.

We are using only the FMTF – frequency masking threshold function.
SCS – Scalar Costa Scheme

**Encoding**: Data \( \{d_1,d_2,\ldots\} \) is embedded on to the host signal \( x \), by discretizing the continues host signal coefficients. We are using \( \alpha = 1 \). \( d \in [0,1] \), therefore:

\[
s_n = x_n + \alpha \left( Q_\Delta \left( x_n - \Delta \left( \frac{d_n}{D} \right) \right) \right) - \Delta \left( \frac{d_n}{2} \right) + \Delta \left( \frac{d_n}{2} \right)
\]

**Decoding**: From the received signal \( r_n \), we get \( y_n \):

\[
y_n = Q_\Delta \{r_n\} - r_n \Rightarrow \hat{d}_n = \begin{cases} 0, & |y_n| \leq \frac{\Delta}{4} \\ 1, & |y_n| \geq \frac{\Delta}{4} \end{cases}
\]
The data bit is embedded onto one coefficient of the host signal in the frequency domain. Ideally, one can embed a bit* of data in each frequency coefficient, using this method.

*Depends on the size of the alphabet, it is one bit in the binary case.
DHT-Discrete Hartley Transform

- The DHT is a Fourier related transform.
- The DHT transforms real vectors into real vectors.

\[
DHT\{X[n]\}[k] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X_n \text{Cas}(\frac{2\pi kn}{N}), \quad 0 \leq k \leq N - 1
\]

Where:

\[
\text{Cas}(\frac{2\pi kn}{N}) = \cos(\frac{2\pi kn}{N}) + \sin(\frac{2\pi kn}{N})
\]

Where the DFT and DHT obey the following:

\[
DHT\{X\}[k] = \Re(DFT\{X\}[k]) - \Im(DFT\{X\}[k])
\]
Algorithm for Data Embedding

- DHT
- Calculating MTF
  - Division into subbands
  - Embedding the data into freq. coefficients using SCS
- Binary Coding
- IDHT

Audio Signal X(n) → Calculating MTF → Division into subbands → Embedding the data into freq. coefficients using SCS → IDHT → Data-Embedded Audio Signal Y(n)

Data Signal d(n) → Binary Coding → Embedding the data into freq. coefficients using SCS
Embedding - Results

The input to the Encoder are 512 samples audio frames. The Encoder calculates the MTF for each frame, from which it derives the \( \Delta[n] \) function.

The graph shows the MTF subband minima, and its quantized version, Delta. The MTF subband minima, and Delta in dB.

The graph shows the original (red) and embedded (blue) signals, DHT's.
Algorithm for Data Decoding

1. DHT
2. Calculating MTF
3. Parameter Estimation
4. Decoding the data using SCS
5. Binary Decoding

Input: Data-Embedded Audio Signal Y(n)
Output: Data Signal d(n)
Decoding – Parameter Estimation

The Decoder receives 512 long frames of encoded, and possibly noised audio. It calculates the MTF on the signal, and then Estimates the $\Delta[n]$ function, out of a final set, using an ML, maximum likelihood, Estimator.

The graph shows the Delta function received from the Encoder (blue), and the decoded Delta. There are errors in the detection, which our decoder is able to correct, so that there is no detection error (at no noise).
Decoding - Results

In the case of a noisy channel, we are trying to minimize the detection error, as a function of WNR, which is the ratio between the embedded data and the noise, in dB.

The graph shows the decoding error (WNR), for our decoder, and the “perfect” decoder, which does not need to estimate the Delta.
References


