EE 338: Evaluation Component 5

Digital Audio Watermarking

Group 18

Adway Girish	180070002
Andrews George Varghese	180070005
Nayan Barhate	180070037

Introduction to Audio Watermarking
Why did we chose this topic?

- Unique identifier embedded in an audio signal to identify ownership/copyright
- If the audio is copied, so is the watermark. This helps to identify files that have been illegally reproduced
- Watermarks are becoming widely popular in enabling copyright protection and ownership verification

We chose this topic because as we move towards a ‘self-reliant’ and ‘digital’ India, a huge quantity of original content is being produced and it is becoming increasingly important to ensure the protection of intellectual property rights. Audio watermarking can help in this regard. Also, the presence of a watermark can be used to check for the authenticity of audio messages, which is a very valuable tool, especially in this age of machine learning based deep fakes

DSP principles involved

1. **Discrete Fourier Transform**

A finite-length discrete signal in the time domain can be uniquely represented in the frequency domain by another finite-length discrete signal called the DFT, given by:

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}kn}$$

Being complex, the DFT has a magnitude as well as a phase. These can be efficiently implemented using numerical algorithms such as the Fast Fourier Transform (FFT).

2. Convolution

The convolution is a mathematical operation to show the spreading of two signals over each other. Given two signals *x[n]* and *y[n]*, we define *z[n]* to be their convolution:

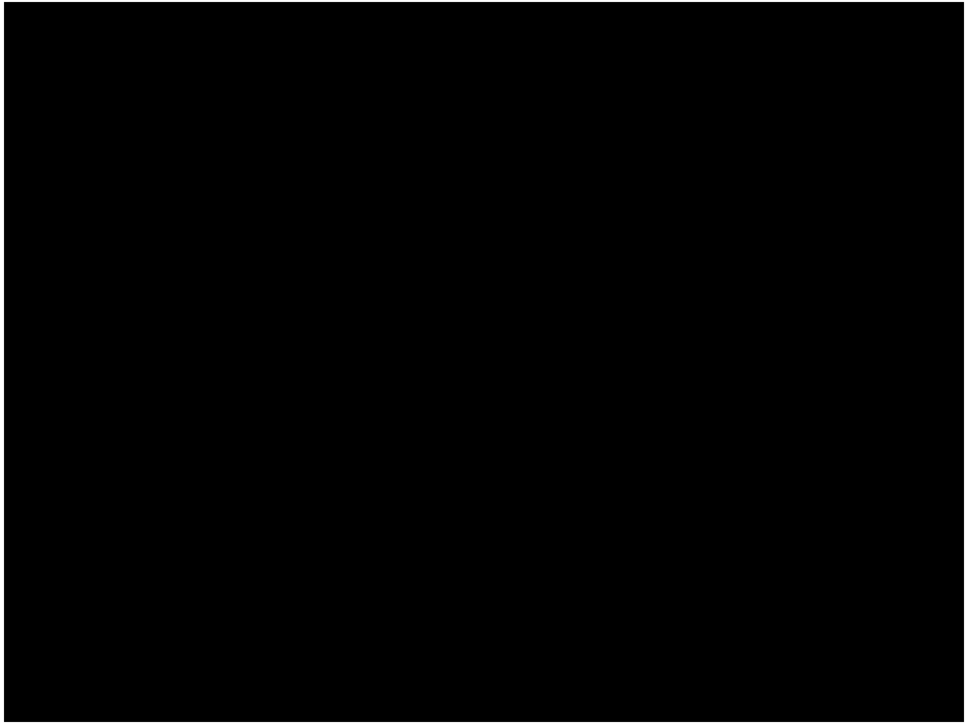
$$z[n] = x[n] * y[n] = \sum_{l=0}^{N-1} \hat{x}[l]\,\hat{y}[n-l], \text{ for } 0 \leq n \leq N-1$$

where $\hat{x}$ and $\hat{y}$ are infinite-length periodic extensions of x and y respectively

The most useful feature of the convolution is that when two signals are convolved with each other, the DFT of the resulting signal is simply the product of the DFTs of the individual signals.

$$Z[k] = X[k]Y[k]$$

This is an animation showing the convolution operation between two signals in action.



Note: These signals are continuous-time, but discrete-time convolution also works effectively the same.

3. Cepstrum

The cepstrum is a homomorphic transformation in the frequency domain, i.e., for any signal *x[n]*, we define its cepstrum $C_x[n]$, such that the DFT of C_x is the logarithm of the DFT of *x[n]*.

$$C_x[n] = \mathcal{F}^{-1}\{\log|X[k]|\}$$

(Technically this is defined to be the real cepstrum since we take the logarithm of the magnitude of *X[k]*; we also have the complex cepstrum which is obtained by taking the complex logarithm of *X[k]*.)

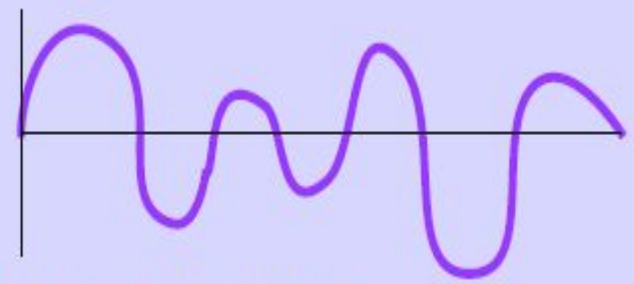
The most useful feature of this is that it converts the convolution operation to addition, i.e. if $z[n] = x[n] * y[n]$

$$C_z[n] = C_x[n] + C_y[n]$$

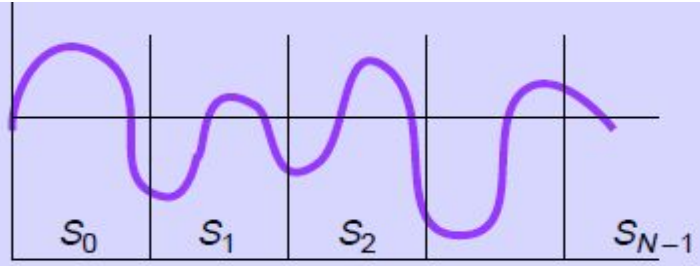
Phase Coding

- The idea here is to hide (or encode) the message to be hidden in the phase of the song.
- Say we divide a song into multiple segments, and take the DFT of these segments. Each segment has a corresponding amplitude and phase spectrum.
- It has been found that the human ear is most sensitive to the difference in phase between consecutive segments, rather than the absolute phase of any one segment.
- Using this principle, we store the message in the phase of the first segment of the song, and construct the phase for the remaining segments while maintaining the phase differences across the segments.

The song is first divided into multiple (say N) segments

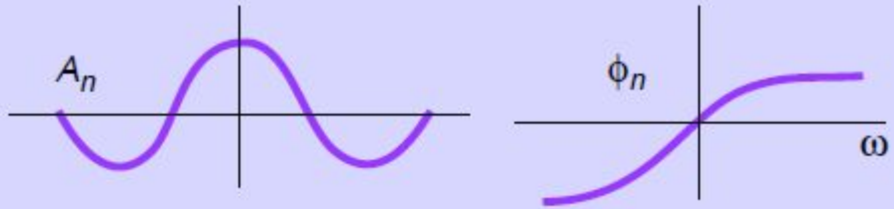


(A) ORIGINAL SOUND SIGNAL



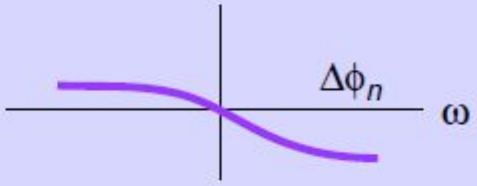
(B) BREAK INTO *N* SEGMENTS *S*

Next we obtain the fourier transform of each segment



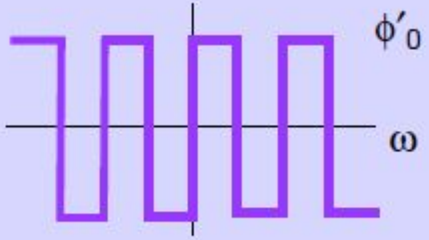
(C) TRANSFORM EACH SEGMENT INTO MAGNITUDE AND PHASE

Obtain the phase difference between consecutive segments (this is vital to ensure the ear cannot distinguish between the original and encoded signal)



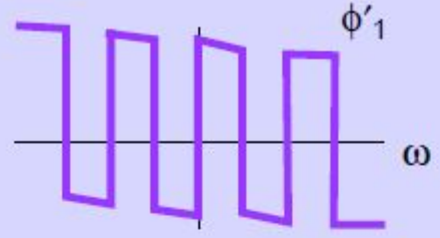
(D) CALCULATE THE PHASE DIFFERENCE BETWEEN CONSECUTIVE SEGMENTS $\phi_{n+1} - \phi_n$

Convert the bit sequence of the song to a predetermined absolute phase, which is now stored in the first segment



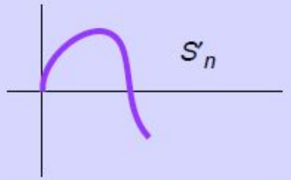
(E) FOR SEGMENT S_0 CREATE AN ARTIFICIAL ABSOLUTE PHASE ϕ'_0

Reconstruct the phase spectra of the other segments using the phase of the first segment and the previously calculated phase differences

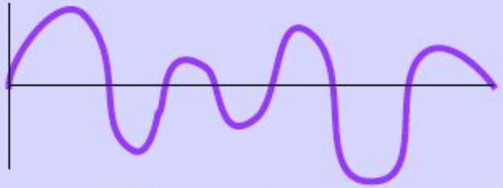


(F) FOR ALL OTHER SEGMENTS, CREATE NEW PHASE FRAMES ($\phi'_0+\Delta\phi_1$)

Finally, using the inverse fourier transform, we obtain the new segments in the time domain, and concatenate them to get the modified signal (which has the message encoded in it)



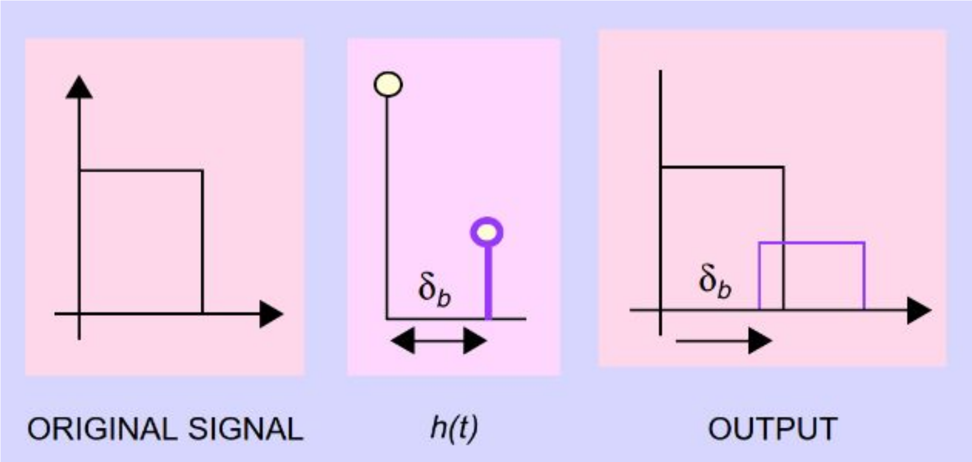
(G) COMBINE NEW PHASE AND ORIGINAL MAGNITUDE TO GET NEW SEGMENT S'_n



(H) CONCATENATE NEW SEGMENTS TO CREATE OUTPUT

Echo Data Hiding

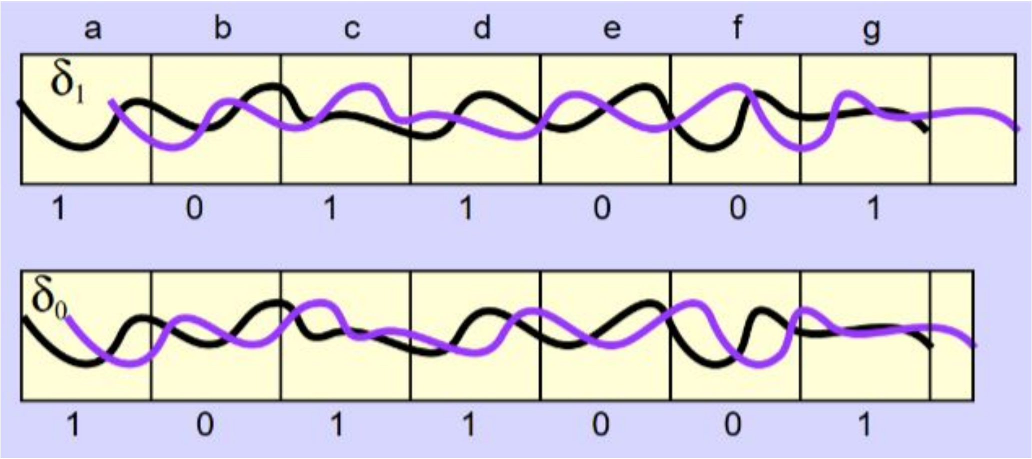
The basic philosophy in echo data hiding is to introduce an echo in the original signal with different delays to denote different bits, as shown in the figure below.



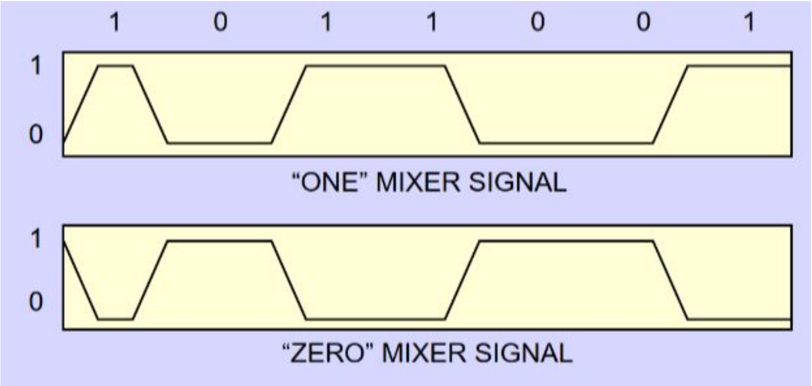
h(t) here is called the "kernel". We have different δ_b to represent 0 and 1, giving us the "zero" kernel and the "one" kernel.

The entire signal is divided into segments, each representing one bit. That segment is then filtered with (convolved with) the corresponding kernel to obtain the echoed segment. All these modified segments are then put together to obtain the final encoded signal.

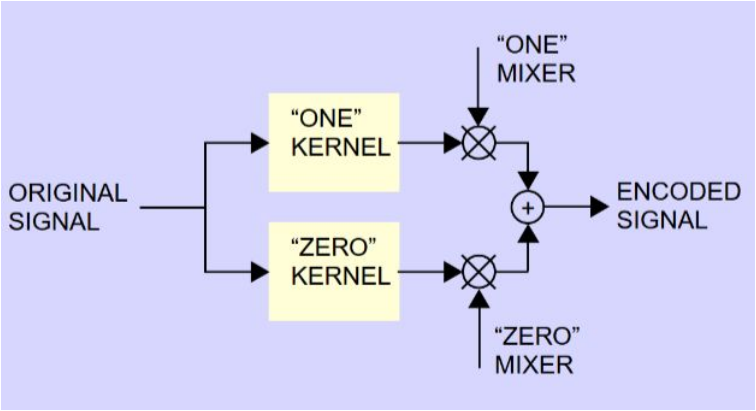
While implementing, this is done by first generating a “one” signal and a “zero” signal by filtering the entire signal with the “one” and “zero” kernels respectively.



Then based on the bit sequence to be encoded, mixer signals are created:



Finally, the mixers are mixed with the appropriate signals to get the encoded signal.



Conclusion

We observe that these techniques of audio steganography have a wide range of applications. Both the methods considered in this project cause little to no noticeable distortion in the audio quality, and can thus be used to hide data in the audio signal.
This obviously is a very powerful tool to be used in protecting the copyrights of audio digital content being produced at a rapid rate in this new, fast progressing digital India. These techniques can be used to encode some key that cannot be easily reproduced into some audio file such as an original recording of a song while it is produced. Since the encoding is embedded within the signal, it cannot be removed by any kind of filtering. To detect plagiarised copies of this original, the suspected audio file is checked for the presence of encoding and subjected to decoding. Depending on the encoded key, the decoded data can not only be used to detect plagiarism, but also identify the owner of the original copy.