Tecnologías Multimedia - Study Guide - Milestone 6: Inter-channel decorrelation in stereo audio signals

Vicente González Ruiz - Depto Informática - UAL

September 30, 2020

1. Description

Correlation is the term used in statistics for refering to the interdependency between random variables. It can be measured by the correlation coefficient [1].

In the case of InterCom, the random variables are the two channels (left L and right R) of the stereo signal. In most cases, both channels are going to be highly correlated (especially if the microphone is mono), which means that we can represent one of them (for example, the R channel) with respect to the other (the L channel). From a mathematical point of view, this process can be seen as a decorrelation process. From a physical perspective, decorrelating implies energy accumulation [2].

To perform this inter-channel decorrelation, we use an orthogonal transform

$$y = Kx = \frac{1}{2}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}x, \tag{1}$$

where x represents a frame (a tuple or L and R samples, $x[0]$ and $x[1]$), K is the 2×2 KLT (Karhunen-Loève Transform) matrix multiplied by $1/\sqrt{2}$ (which is closely related with the Haar transform [4]), and y represents the transform coefficients (in our case, a couple of coefficients $y[0]$ and $y[1]$). Notice that this transform is not orthonormal (energy preserving in the transform domain) because

$$\sum y[i]^2 = \frac{1}{\sqrt{2}} \sum x[i]^2, \tag{2}$$

although both subbands $y[0]$ and $y[1]$ have the same gain $(1/\sqrt{2})$. This will simplify the quantization of y. The described transform is similar to the so called M/S stereo coding, but in our case, the división by 2 is carried on on the forward transform, instead of the backward (inverse) transform.

This transform can be implemented *in-place* using the following algorithm:

Algorithm 1.1: $\textsc{Inter-channel\_decorrelation}(\ )$

procedure $\textsc{Analyze}(\texttt{frame})$

$$\begin{cases} \texttt{frame}[0]- = \texttt{frame}[1] \\ \texttt{frame}[1]+ = (\texttt{frame}[0] >> 1) \\ \texttt{frame}[0] >>= 1 \end{cases}$$

procedure $\textsc{Synthesize}(\texttt{frame})$

$$\begin{cases} \texttt{frame}[0] <<= 1 \\ \texttt{frame}[1]- = (\texttt{frame}[0] >> 1) \\ \texttt{frame}[0]+ = \texttt{frame}[1] \end{cases}$$

where $\texttt{a}- = \texttt{b}$ is a shorter representation of the operation $\texttt{a} = \texttt{a} - \texttt{b}$, and $>>$ and $<<$ represent the bit-wise left and right shift operations, respectively. Notice that this type of in-place computations are commonly used in the implementation of DWTs (Dicrete Wavelet Transforms) using the Lifting Scheme [3].

2. What you have to do?

1. Inherit the class `Intercom_minimal` and override the method `record_io_and_play()`. Do this in a module named `intercom_intra-channel_decorrelation.py`.

2. Implement in this method the procedures Analyze() and Synthesize(). Vectorized operations should be used.

3. Timming

You should reach this milestone at most in two weeks.

4. Deliverables

The module `intercom_inter.py`. Store it at the root directory of your `intercom`'s repo.

5. Resources

[1] Allen B. Downey. *Think Stats Probability and Statistics for Programmers*. O'Reilly, 2011.

[2] Khalid Sayood. *Introduction to data compression*. Morgan Kaufmann, 2017.

[3] Wim Sweldens. The Lifing Scheme: A Custom-Desing Constuction of Biorthogonal Wavelets. *Applied and Computational Harmonic Analysis*, 3(2):186–200, 1996.

[4] Martin Vetterli and Jelena Kovacevic. *Wavelets and Subband Coding*. Prentice-hall, 1995.