Transform coding of audio

How can we apply the DCT method to audio coding? Suppose we partition our audio signal into blocks of size $m = 512$ samples. High quality audio is sampled at $44,000$ samples per second, each block corresponds to $\frac{512}{44,000}$ seconds, which is about 12 ms per block. This is a very short time interval. There are about 80 blocks in one second. We compute the DCT of each block j. This gives us a vector

$$\vec{Y_j} = \mathbf{C}^T \vec{X_j}$$

where the elements of $\vec{Y_j}$ are indexed by $k = 0, 1, \ldots, 511$ and $\mathbf{C}$ is the $m \times m = 512 \times 512$ DCT matrix.

What do the k's signify? The k's are frequencies, namely the number of cycles of a cosine wave per 512 samples, that is, per block. The lowest frequency ($k = 1$) is $\frac{1}{2}$ cycle (per 12 ms), or about 40 cycles per second (40 Hz, called "Herz"). The next lowest ($k = 2$) is 1 cycle (80 Hz), etc, and $k = 511$ is sbout 256 cycles per 12 ms which is over 20,000 cycles per second (Hz).

[ASIDE: The human ear is not capable of measuring sound at frequencies higher than about 22,000 Hz. This is in fact why high quality audio is sampled at 44,000 samples per second. A sampling rate that is higher than this would be able to capture high frequency sounds, but the human ear would not be able to hear these components.]

The DCT represents a sampled audio signal as a sum of cosine functions. The $\vec{Y}$ values say how much of each of these pure tone sounds is present in the block $\vec{X_j}$, i.e.

$$\vec{X_j} \;=\; \mathbf{C}\; \vec{Y_j}$$

where the k^{th} column of $\mathbf{C}$ is a cosine function corresponding to frequency k in the DCT, ando so we see that $\vec{X}$ is just a sum of cosine functions of various frequencies.

We could compress the audio signal $X(t)$ by quantizing the $\vec{Y}$ vectors for each block e.g. by using a quantizer with width Δ. This would be straightforward, and would just be a 1D version of the method described last lecture (JPEG). What we will do instead (and what MP3 does) is more sophisticated and more effective than this, namely we will allow the Δ to vary. It will take me a few lectures to explain why and how.

Spectrograms

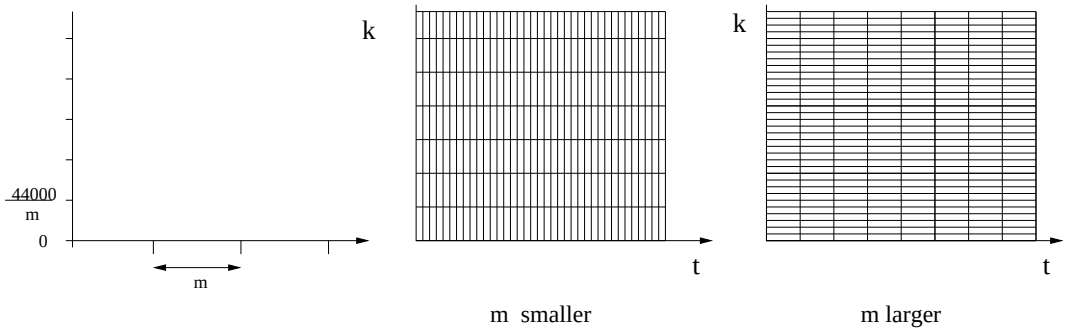
A convenient way to organize the DCT coefficients is to use spectrograms. A *spectrogram* is a function $Y_j(k)$ where j is the block number and k is the frequency. We rewrite the equation above as:

$$Y_j(k) \;=\; \mathbf{C}\vec{X_j}$$

where $\vec{Y_j} = Y_j(k)$. The sketch below shows two spectrograms, defined for different block sizes. For example, if $m = 512$ then k's represent multiples of about 40 Hz (see above). If each block were longer, say $m = 2048$, then the k's would represent multiples of about 10 Hz. If each block were very short, say $m = 32$, then the k's would represent multiples of about 700 Hz. Such spectrograms will be very important later when we get to MP3.

We can think of the spectrogram as a 2D image with a fixed number of "pixels", $Y_j(k)$, equal to the number of samples in the original sound signal. In the figure below, think of the particular (j, k) values of the spectrogram as lying on the vertical and horizontal lines. Thus, when m is larger

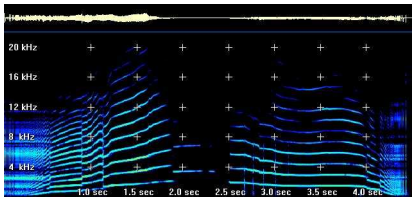
(on the right) there are fewer j values along the t axis, and more values on the frequency k axis. (If you are confused about j vs. t, recall that j indexes the block number whereas t indexes the samples. There are m times as many t values as j values.)

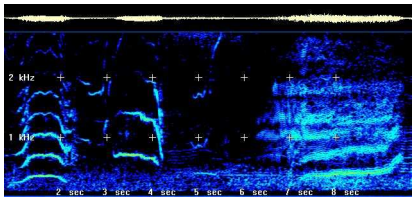


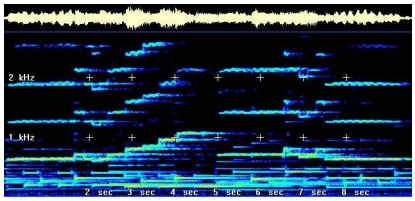
I encourage you to check out various examples of spectrograms here and listen to the corresponding sounds:

`http://www.visualizationsoftware.com/gram/examples.html`

Below are three of them[1]. Notice that in each case, the non-zero values are concentrated in particular bands of frequencies and these frequencies vary over time.


creaking door


song of a humpback whale


musical flute

Musical scores

Spectrograms are closely related to *musical scores*, namely the notation that musicians use to specify the timing of notes to be played in a song. A musical score is a 2D map, whose horizontal axis is a time line and whose vertical axis is a set of possible notes that can be played – in the case of a keyboard (e.g. piano), these notes correspond to the keys to be pressed. Higher notes correspond to higher frequencies of sound.

There are significant differences between musical scores and spectrograms, however. A musical score is basically a binary image, indicating whether a note is to be played at some time step or not. A spectrogram, by contrast, has a range of values at each (j, k) which indicate the strength of a particular frequency k during block j.

[1]The bumpy thick line at the top of the spectrogram is the function $X(t)$ namely the raw sound signal itself, prior to computation of the spectrogram.

A second difference between the notes in a musical score and the values $Y_j(k)$ of a spectrogram is that a "note" actually consists of many frequencies. For example, when you hit a key on a piano or you pluck a guitar string etc, you do not get a single frequency but rather you get a family of frequencies. (See the musical flute spectrogram above, where one "note" gives a range of k values where $Y_j(k)$ is large.)

Speech sounds

Spectrograms are often used in modelling speech. Human speech sounds have particular properties that are due to the anatomy. I don't mean that each human language has a relatively small number of words. Rather, I mean that the human anatomy itself constrains the kinds of sounds that can be produced when a person speaks. We all have roughly the same physical dimensions. Some people are smaller or bigger than average - children are smaller than adults, and have shorter vocal cords, and hence higher pitched voices. But despite these variations there are huge similarities as well.

Speech sounds include all (!) sounds produced when a person expells air from the lungs. The sound that is produced depend on several factors under the person's control. One key factor is the shape of the cavity inside your mouth. This shape is defined by the tongue, lips, and jaws, together known as the *articulators*. Consider the different vowel sounds in normal spoken English "aaaaaa", "eeeeeee", "iiiiiiii", "oooooo", "uuuuuuu". Make these sounds to yourself and notice how you need to move your tongue, lips, and jaw around (the tongue, lips, jaw are called "articulators".) Think of the space inside your mouth and throat and nasal cavity as a resonant tube, like a bottle. Changing the shape of the tube by varying the articulators produces different waveforms to be emitted from the mouth.

Voiced vs. unvoiced sounds

The throat plays a very important role in speech sounds. Inside the throat are the vocal cords. Certain speech sounds require that the vocal cords vibrate while other speech sounds require that they do not vibrate. When the vocal cords are tensed, the air rushing by them causes them to vibrate and the resulting sounds are called *voiced*. When the vocal cords are relaxed, the resulting sounds are called *unvoiced*. The standard example of an unvoiced speech sound is whispering. Normal human speech is a combination of voiced and unvoiced sounds.

Voiced sounds are formed by pulses of air emitted at a *regular* rate from the (tensed) vocal cords. There is an opening the vocal cords, called the *glottis*. When the vocal cords are tensed, the glottis opens and closes at a regular rate, as air pressure from the lungs builds up behind the glottis and is then released. A typical glottal pulse rate for spoken speech is about 10 ms. You can change this rate by providing different amounts of tension. That is what happens when you sing different notes.

Each glottal pulse has a simple shape $X(t)$. This pulse then gets reshaped into a more complicated waveform that depends on the position of the articulators. If the articulators are fixed in place over some time interval, each glottal pulse undergoes the same transformation. Some people talk very quickly but not so quickly that the position of the tongue, jaw and mouth changes much over time scales of the order of 10 ms.

When the vocal cords are relaxed, e.g. when you whisper, the resulting sounds are called *unvoiced*. It is very difficult to write a mathematical model of unvoiced sounds. When air rushes

through the relaxed vocal cords, the sound that results is like that of the wind rushing through the trees. Its very "noisy". The sound samples $(\ldots x_j x_{j+1} x_{j+2} \ldots)$ do have some structure – otherwise understanding whispered speech would be impossible! But the structure is much more difficult to capture with a mathematical model than is the case for voiced speech.

Consonants

I mentioned above that speech contains transitions from voiced to unvoiced. There are other transitions too, of course, namely when the articulators move. The resulting sound transitions also need to be encoded. Let's consider a few examples.

One way to produce meaningful sounds is to temporarily restrict the flow of air, and force it through a small opening. Most *consonants* are defined this way, namely by a partial or complete blockage of air flow. There are several classes of consonants. Let's consider a few of them. For each, you should consider what is causing the blockage (lips, tongue, specific part of palatte i.e. roof of mouth).

- fricatives (narrow constriction in vocal tract):
  - unvoiced: s, f, sh, th (as in θ)
  - voiced: z, v, zh, th (as in *the*)
- stops (temporary cessation of air flow):
  - unvoiced: p, t, k
  - voiced: b, d, g

  These are distinguished by where in the mouth the flow is cutoff. Stops are accompanied by a brief silence
- nasals (oral cavity is blocked, but nasal cavity is open)
  - voiced: m, n, ng

  If you don't believe me when I tell you that nasal sounds actually come out of your nose, then try shutting your mouth, plugging your nose with your fingers, and saying "mmmmm".

The above discussion is obviously not complete, but it should at least give you a sense of the possibilities in modelling speech sounds, and you should appreciate that such models could be used for compression of speech sounds.