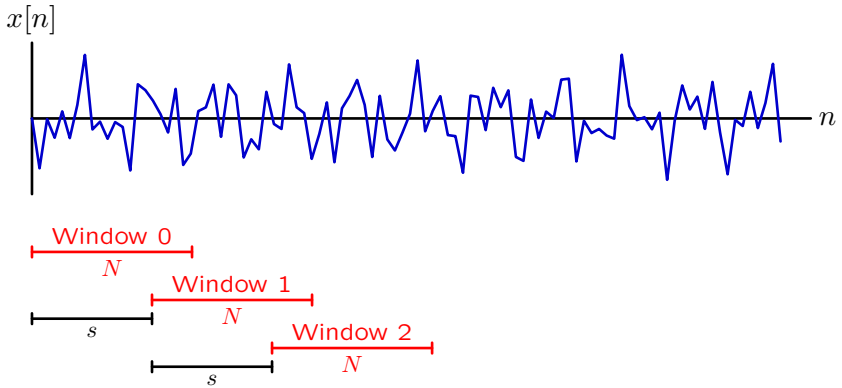
6.003: Signal Processing

Short-Time Fourier Transform and Window Functions

13 April 2021

STFT

Short-time Fourier transforms are based on the analysis of a sequence of finite-length portions of an input signal.



Motivating Example: Song Recognition

Song detection systems (like Shazam)

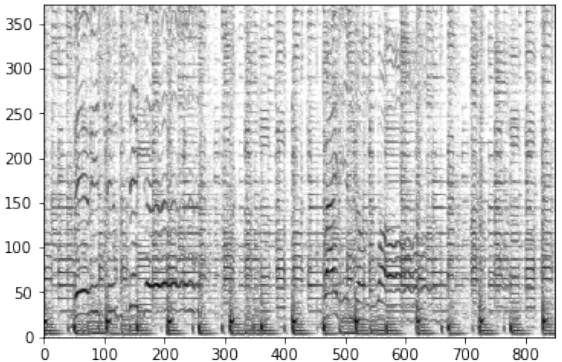
Goal: given a (potentially corrupted) recording of a portion of a song, determine what song that recording came from

Example: `noisy.wav`

How is this possible?

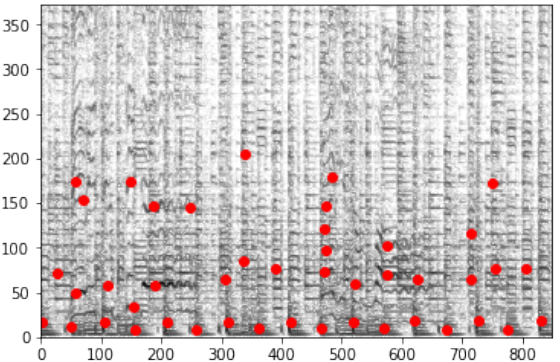
Song Recognition: Fingerprinting

Typical song recognition systems work by producing a *fingerprint* of a song. For example, we can start with a spectrogram of a song:



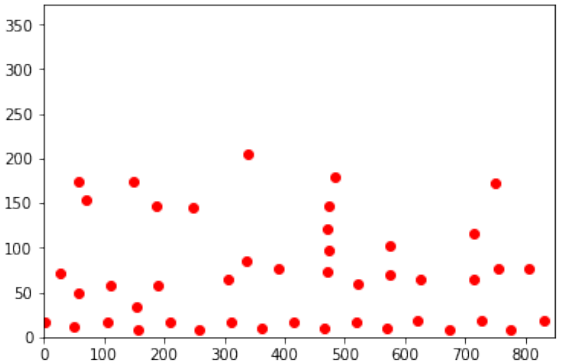
Song Recognition: Fingerprinting

Then, we can find peaks within that spectrogram to produce a sparse representation of the song:



Song Recognition: Fingerprinting

Then, we can find peaks within that spectrogram to produce a sparse representation of the song:



This representation is (surprisingly) robust against a lot of kinds of corruption we might expect to experience in the world.

Song Recognition

We can then compute a similar fingerprint for the noisy recording, and look for a match across the song (and across the other songs in our database).

This process works really well, but it is helped dramatically by having a relatively clean spectrogram of the original song to start with (so that we can find reliable information about the peak frequencies).

Spectrograms

Consider a small sound in `cos0.wav`, consisting of a single cosine alternating between two frequencies.

What do we expect its spectrogram to look like?

Windowing

To understand what we just saw, let's think just about the first "window" of the spectrogram, and how it was computed.

Characterizing Windows

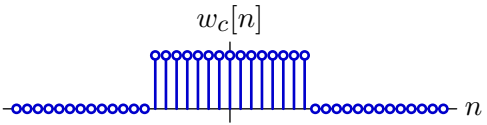
Each sequence of length N can be thought of as the product of $x[n]$ times a shifted version of a rectangular window $w[n]$ of length N:

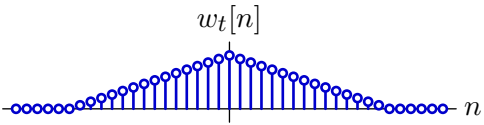
$$w[n] = \begin{cases} 1 & \text{if } 0 \leq n < N \\ 0 & \text{otherwise} \end{cases}$$

Find the DTFT of $w[n]$.

Characterizing Other Windows

What if we use a triangular window instead of a rectangular window?





Characterize the blurring properties of a triangular window.