Time-frequency Masking

EECS 352: Machine Perception of Music & Audio
The **Short-Time Fourier Transform (STFT)** is a succession of local Fourier Transforms (FT).

- **Real spectrogram**
- **Imaginary spectrogram**
- **Time signal**
- **Window**
- **Frequency**
- **Time**
- **Frame**

STFT

Zafar Rafii, Winter 2014
STFT

• If we used a window of N samples, the FT has N values, from 0 to N-1; e.g., if N = 8...

Time signal

+ j*

Real spectrum

window i

frame i

frequency

Frequency signal

window i

frame i

frequency

N frequency values

N frequency values

-3 -1 0 1 2 1 0 -1 + j*

0 1 2 3 4 5 6 7

Zafar Rafii, Winter 2014
**STFT**

- Frequency index 0 is the **DC component**; it is always real (it is the sum of the time values!)

Time signal

![Time signal](image)

**FT**

Real spectrum

![Real spectrum](image)

Imaginary spectrum

![Imaginary spectrum](image)

Window $i$

Frame $i$

![Frame $i$](image)

$$\begin{align*}
\text{Frequency frame } i & \quad \text{Real spectrum} \\
\text{Frame } i & \quad \text{Imaginary spectrum} \\
\end{align*}$$

$$\begin{align*}
\text{Time signal} & \quad \text{FT} \\
\text{Window } i & \quad \text{Time} \\
\end{align*}$$

$$\begin{align*}
\text{Window } i & \quad \text{Time} \\
\text{Frame } i & \quad \text{Frequency} \\
\end{align*}$$
STFT

• Frequency indices from 1 to floor(N/2) are the “unique” complex values \((a + j*b)\)

Time signal → FT → Real spectrum + j* Imaginary spectrum

Window \(i\) → Time → Frequency \(i\) → Frequency \(i\) → Complex Values

Zafar Rafii, Winter 2014
STFT

• Frequency indices from floor(N/2) to N-1 are the “mirrored” complex conjugates \((a - j*b)\)
• If $N$ is even, there is a **pivot component** at frequency index $N/2$; it is always real!
STFT

• Summary of the frequency indices and values in the STFT (in colors!)

N frequency values = frequency 0 to N-1

Frequency 0 = DC component (always real)

Frequency 1 to floor(N/2) = “unique” complex values

Frequency N/2 = “pivot” component (always real)

Frequency floor(N/2) to N-1 = “mirrored” complex conjugates
The (magnitude) **spectrogram** is the magnitude (absolute value) of the STFT.
Spectrogram

- For a complex number $a + j \cdot b$, the absolute value is $|a + j \cdot b| = \sqrt{a^2 + b^2}$
• All the $N$ frequency values (frequency indices from 0 to $N-1$) are **real and positive** (abs!)
Spectrogram

- Frequency indices from 0 to floor(N/2) are the unique frequency values (with DC and pivot)
Spectrogram

• Frequency indices from floor(N/2)+1 to N-1 are the **mirrored frequency values**

\[ \text{Real spectrum} + j* \text{Imaginary spectrum} \rightarrow \text{abs} \rightarrow \text{Magnitude spectrum} \]

-3 -1 0 1 2 1 0 -1 + j* 0 -1 0 1 0 -1 0 1 = 3 1.4 0 1.4 2 1.4 0 1.4

Zafar Rafii, Winter 2014
Since they are redundant, we can discard the frequency values from \( \text{floor}(N/2)+1 \) to \( N-1 \).
• The spectrogram has therefore $\text{floor}(N/2)+1$ unique frequency values (with DC and pivot)
Spectrogram

• Why the magnitude spectrogram?
  – Easy to visualize (compare with the STFT)
  – Magnitude information more important
  – Human ear less sensitive to phase
Spectrogram

• When you display a spectrogram in Matlab...
  – `imagesc`: data is scaled to use the full colormap
  – `10*log10(V)`: magnitude spectrogram in dB
  – `set(gca,'YDir','normal')`: y-axis from bottom to top
Spectrogram

• The signal **cannot be reconstructed** from the spectrogram (phase information is missing!)
Time-frequency Masking

• Suppose we have a mixture of two sources: a music signal and a voice signal
We assume that the sources are sparse = most of the time-frequency bins have null energy.

Time-frequency Masking

Music signal + Voice signal = Mixture signal

Music spectrogram + Voice spectrogram = Mixture spectrogram

Zafar Rafii, Winter 2014
We assume that the sources are \textbf{sparse} = most of the time-frequency bins have null energy.
We assume that the sources are disjoint = their time-frequency bins do not overlap.
Time-frequency Masking

- We assume that the sources are \textbf{disjoint} = their time-frequency bins do not overlap

![Music signal](image1) + ![Voice signal](image2) = ![Mixture signal](image3)

Not a lot of overlapping
Time-frequency Masking

• Assuming sparseness and disjointness, we can **discriminate** the bins between mixed sources.
Time-frequency Masking

• Assuming sparseness and disjointness, we can **discriminate** the bins between mixed sources.

![Music signal](image1.png) + ![Voice signal](image2.png) = ![Mixture signal](image3.png)

Source 1 = bright
Source 2 = dark
Time-frequency Masking

• Bins that are likely to belong to one source are assigned to 1, the rest to 0 = binary masking!
Time-frequency Masking

- By multiplying the binary mask to the mixture spectrogram, we can “preview” the estimate.
However, we cannot derive the estimate itself because we cannot invert a spectrogram!
Time-frequency Masking

- We mirror the redundant frequencies from the unique frequencies (without DC and pivot)
Time-frequency Masking

• We then apply this full binary mask to the STFT using a element-wise multiplication
The estimate signal can now be reconstructed via inverse STFT.

Time-frequency Masking

- The estimate signal can now be reconstructed via inverse STFT.
Time-frequency Masking

- Sources are not really sparse or disjoint in time-frequency in the mixture
Time-frequency Masking

• Bins that are likely to belong to one source are close to 1, the rest close to 0 = soft masking!
Time-frequency Masking

• Let’s listen to the results!
Question

• How can we efficiently model a binary/soft time-frequency mask for source separation?...

• To be continued...