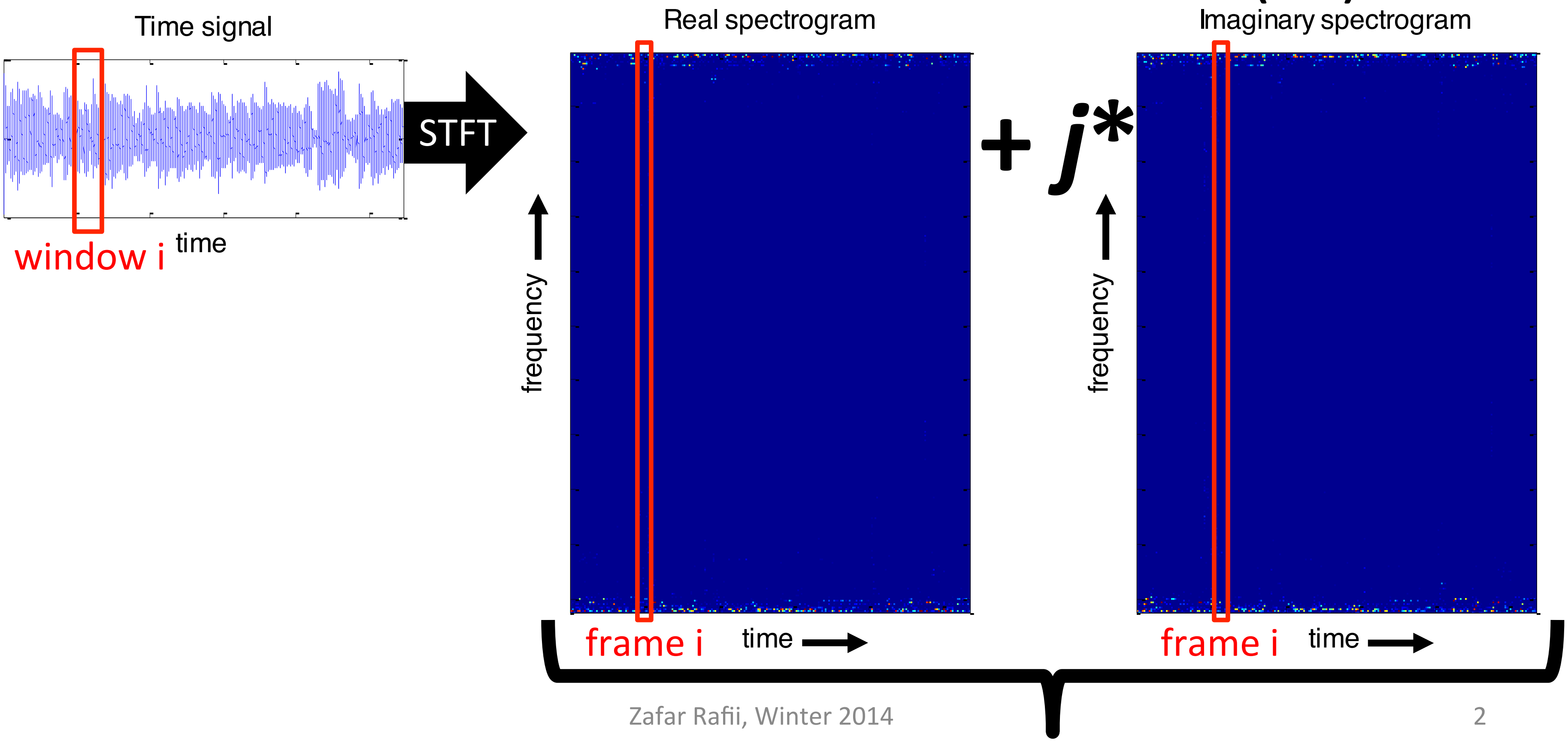
Time-frequency Masking

EECS 352: Machine Perception of Music & Audio

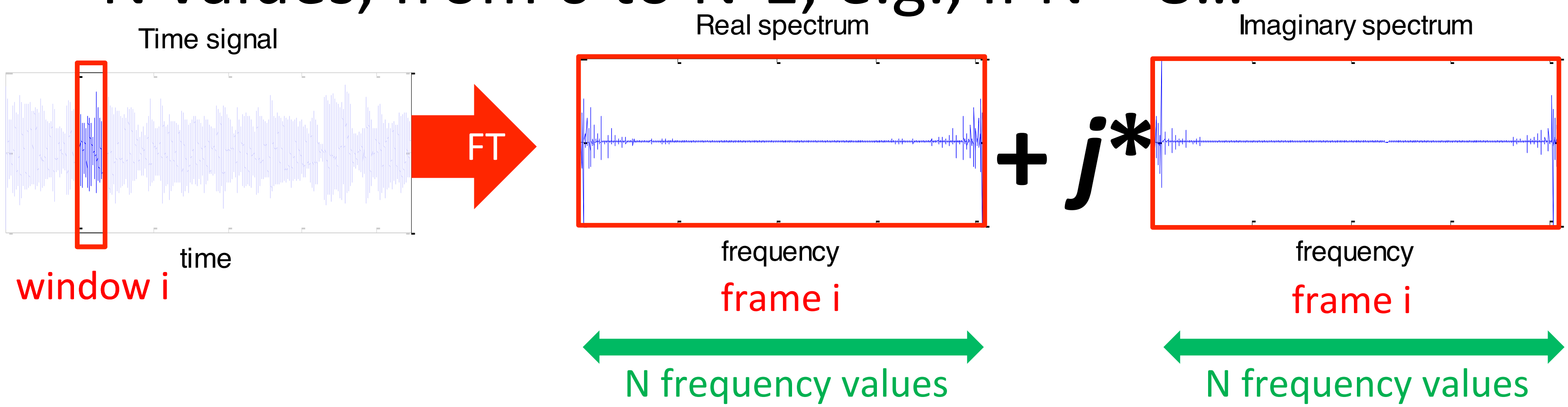
STFT

- The **Short-Time Fourier Transform** (STFT) is a succession of local Fourier Transforms (FT)



STFT

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



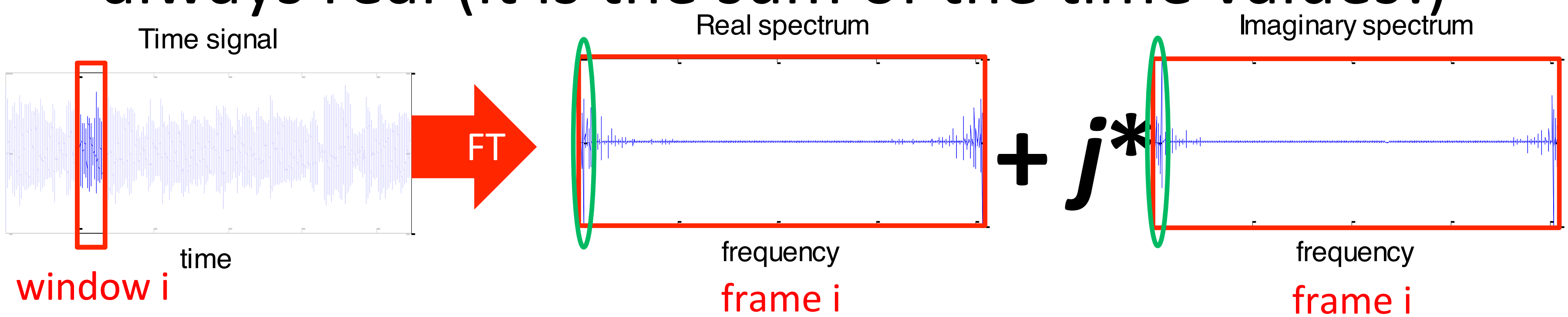
-3	**-1**	**0**	**1**	**2**	**1**	**0**	**-1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

+ *j**

0	**-1**	**0**	**1**	**0**	**-1**	**0**	**1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

STFT

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



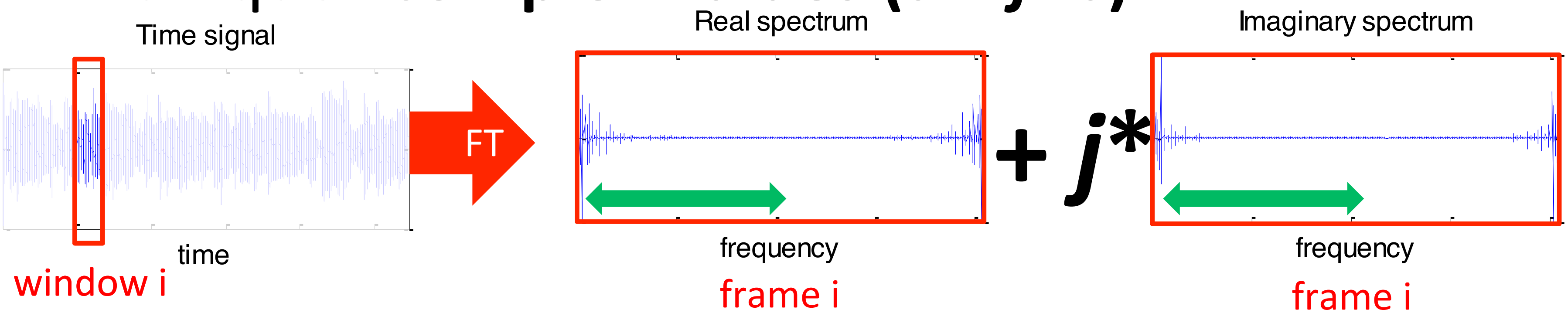
-3	**-1**	**0**	**1**	**2**	**1**	**0**	**-1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

+ *j**

0	**-1**	**0**	**1**	**0**	**-1**	**0**	**1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

STFT

- Frequency indices from 1 to floor(N/2) are the "unique" **complex values (a + *j**b)**



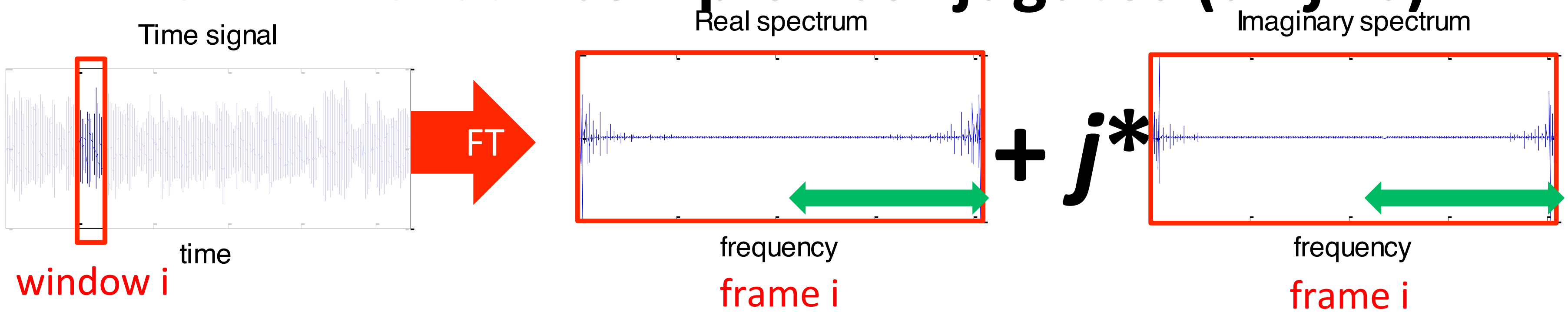
0	1	2	3	4	5	6	7
-3	-1	0	1	2	1	0	-1

+ *j**

0	1	2	3	4	5	6	7
0	-1	0	1	0	-1	0	1

STFT

- Frequency indices from floor(N/2) to N-1 are the “mirrored” **complex conjugates (a - *j**b)**



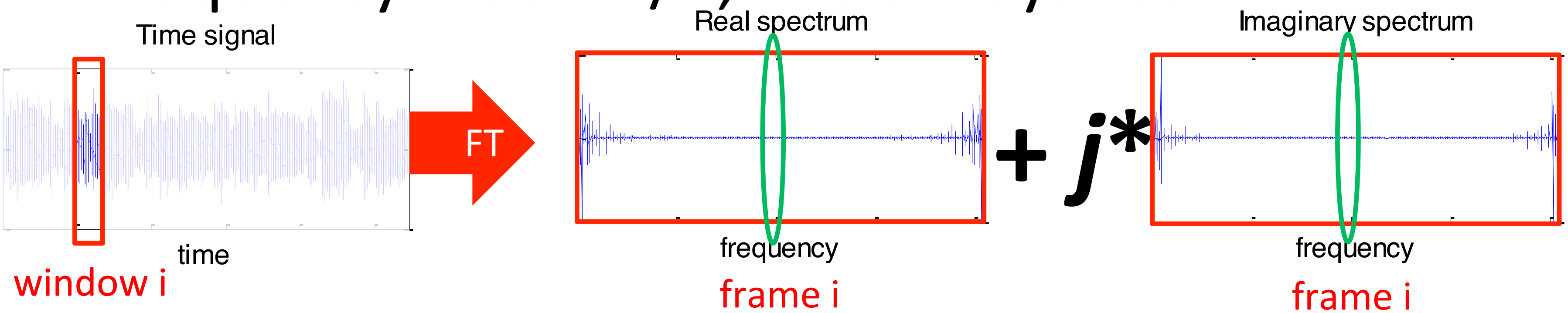
-3	**-1**	**0**	**1**	**2**	**1**	**0**	**-1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

+ *j**

0	**-1**	**0**	**1**	**0**	**-1**	**0**	**1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

STFT

- If N is even, there is a **pivot component** at frequency index N/2; it is always real!



Real	**-3**	**-1**	**0**	**1**	**2**	**1**	**0**	**-1**
index	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*

+ *j**

Imaginary	**0**	**-1**	**0**	**1**	**0**	**-1**	**0**	**1**
index	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*

STFT

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

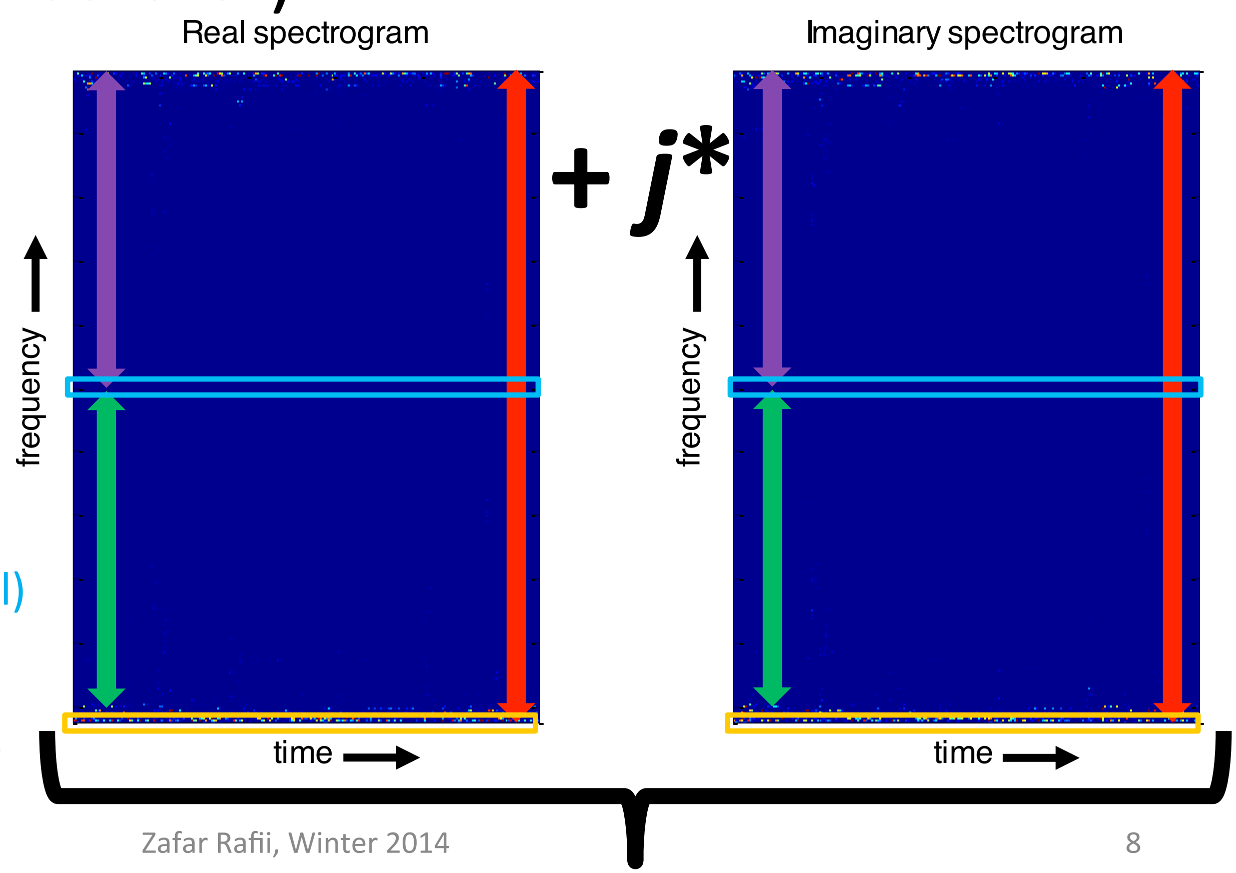
N frequency values =
frequency 0 to N-1

Frequency 0 =
DC component (always real)

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

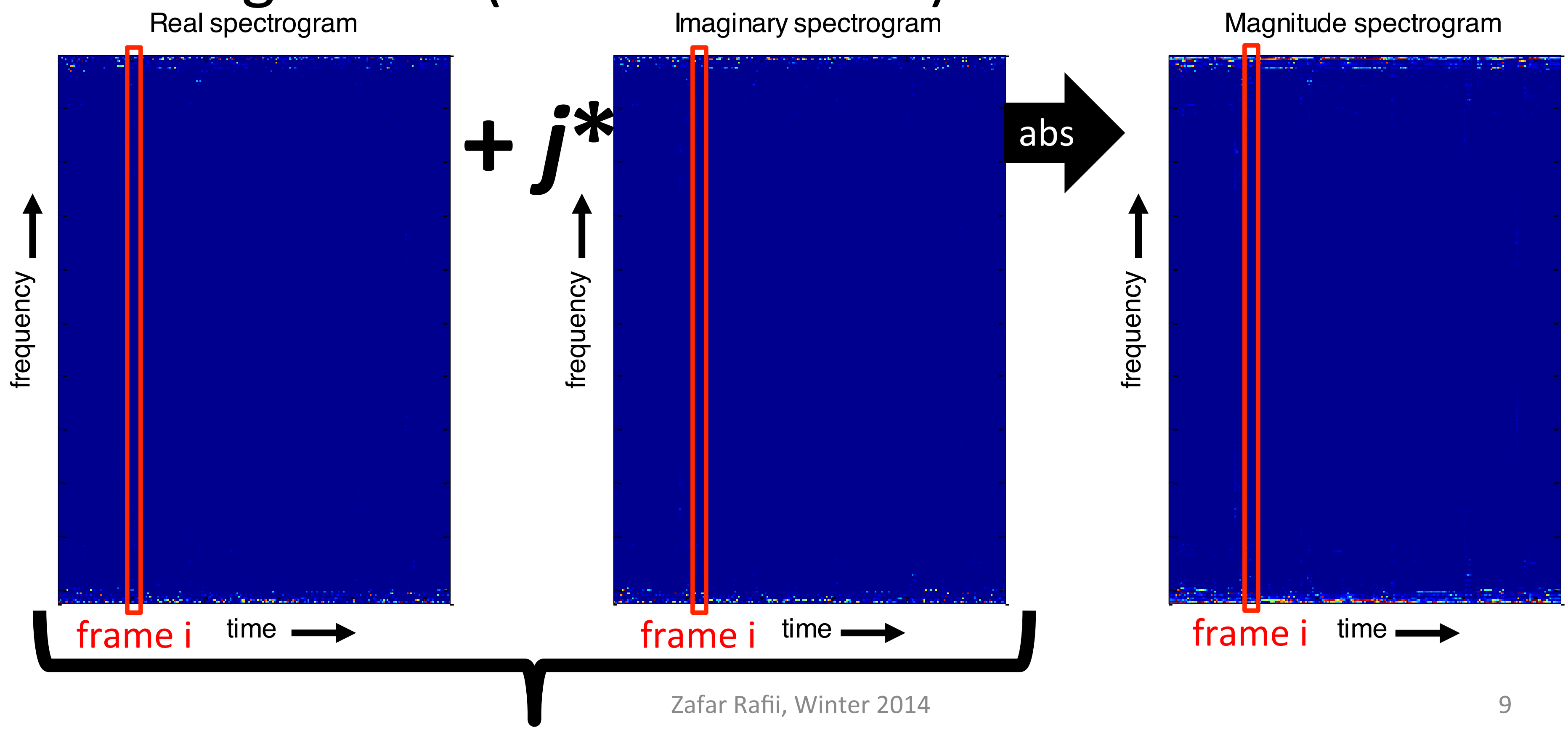
Frequency N/2 =
“pivot” component (always real)

Frequency floor(N/2) to N-1 =
“mirrored” complex conjugates



Spectrogram

- The (magnitude) **spectrogram** is the magnitude (absolute value) of the STFT



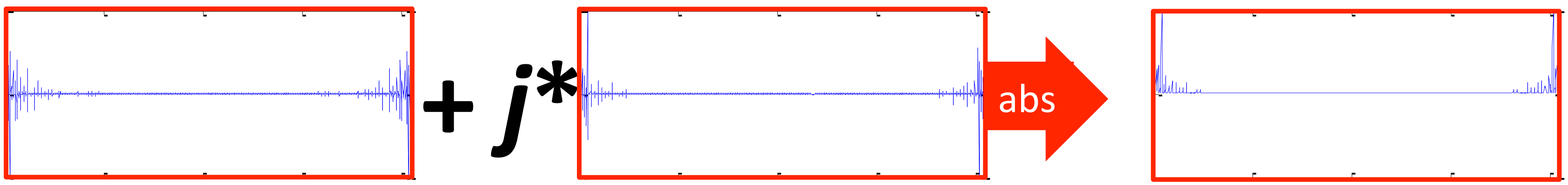
Spectrogram

- For a complex number $a+j*b$, the absolute value is $|a+j*b|=\sqrt{a^2+b^2}$

Real spectrum

Imaginary spectrum

Magnitude spectrum



frequency

frame i

frequency

frame i

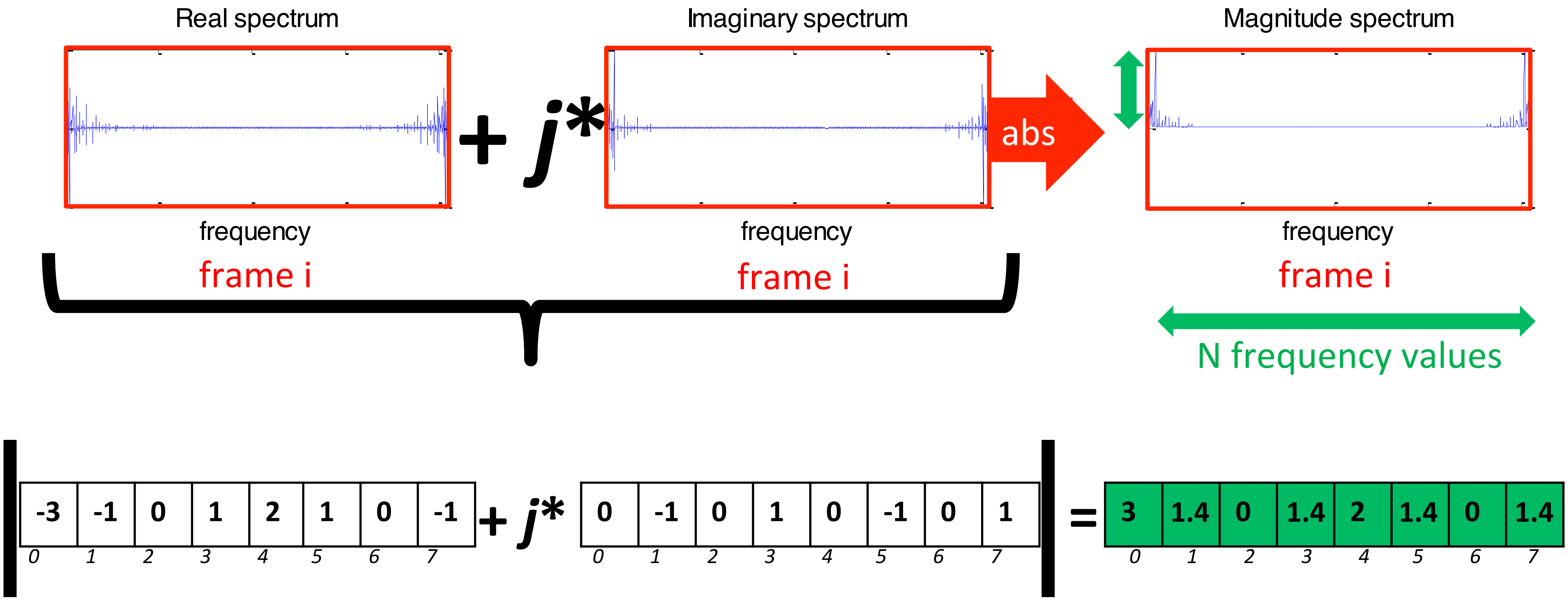
frequency

frame i

$$\left|\begin{array}{|c|c|c|c|c|c|c|c|}\hline -3 & -1 & 0 & 1 & 2 & 1 & 0 & -1 \\ \hline \multicolumn{1}{c}{\scriptstyle 0} & \multicolumn{1}{c}{\scriptstyle 1} & \multicolumn{1}{c}{\scriptstyle 2} & \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 4} & \multicolumn{1}{c}{\scriptstyle 5} & \multicolumn{1}{c}{\scriptstyle 6} & \multicolumn{1}{c}{\scriptstyle 7}\end{array} + j* \begin{array}{|c|c|c|c|c|c|c|c|}\hline 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 \\ \hline \multicolumn{1}{c}{\scriptstyle 0} & \multicolumn{1}{c}{\scriptstyle 1} & \multicolumn{1}{c}{\scriptstyle 2} & \multicolumn{1}{c}{\scriptstyle 3} & \multicolumn{1}{c}{\scriptstyle 4} & \multicolumn{1}{c}{\scriptstyle 5} & \multicolumn{1}{c}{\scriptstyle 6} & \multicolumn{1}{c}{\scriptstyle 7}\end{array}\right| = \begin{array}{|c|c|c|c|c|c|c|c|}\hline 3 & 1.4 & 0 & 1.4 & 2 & 1.4 & 0 & 1.4 \\ \hline\end{array}$$

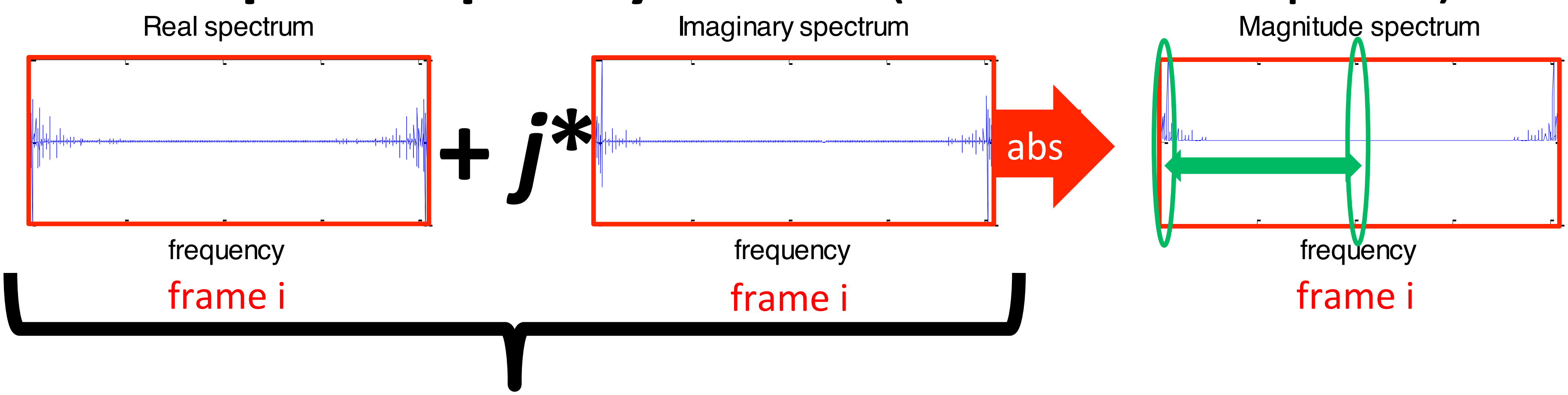
Spectrogram

- 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)



Real part:

-3	**-1**	**0**	**1**	**2**	**1**	**0**	**-1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

$+\, j*$ (imaginary part):

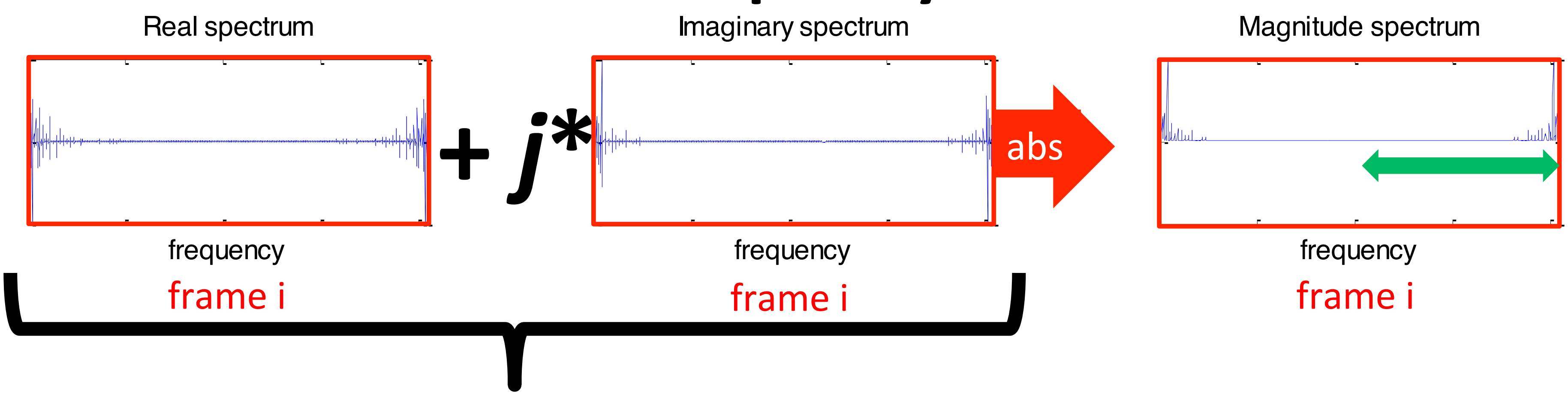
0	**-1**	**0**	**1**	**0**	**-1**	**0**	**1**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

$=$ (absolute value):

3	**1.4**	**0**	**1.4**	**2**	**1.4**	**0**	**1.4**
0	*1*	*2*	*3*	*4*	*5*	*6*	*7*

Spectrogram

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



value	-3	-1	0	1	2	1	0	-1
index	0	1	2	3	4	5	6	7

+ *j**

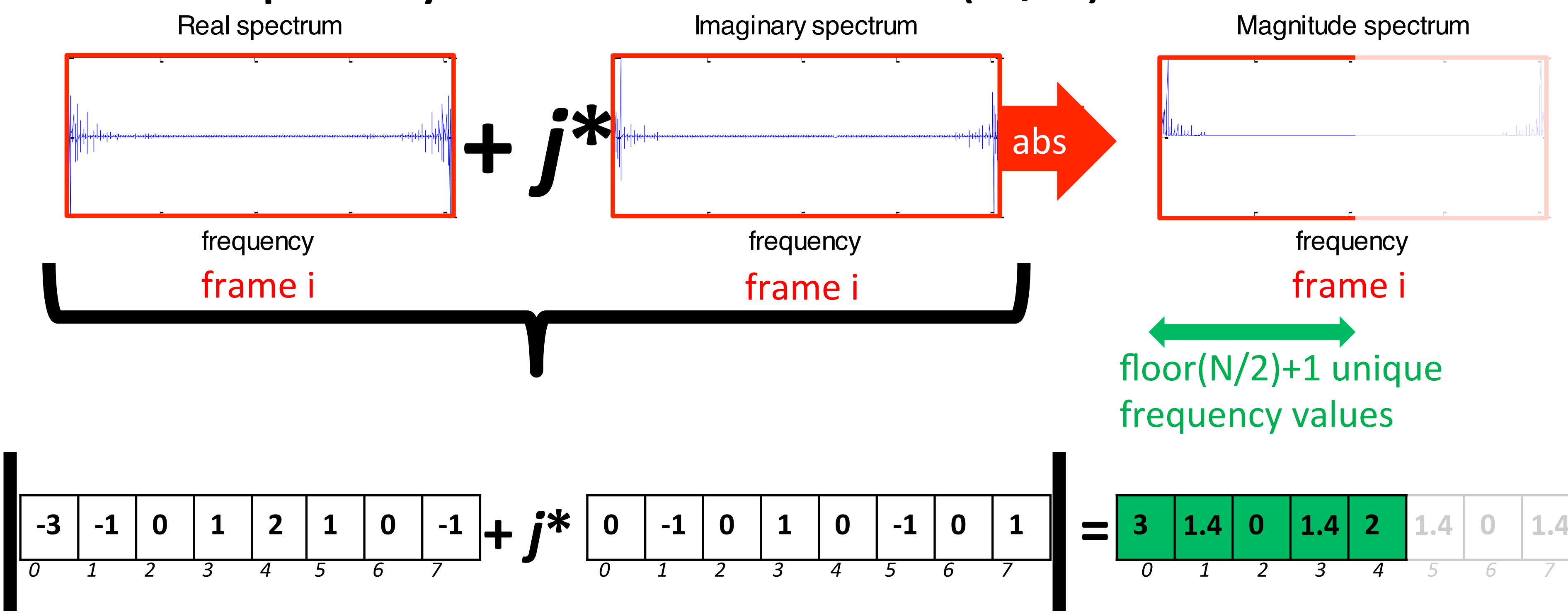
value	0	-1	0	1	0	-1	0	1
index	0	1	2	3	4	5	6	7

=

value	3	1.4	0	1.4	2	1.4	0	1.4
index	0	1	2	3	4	5	6	7

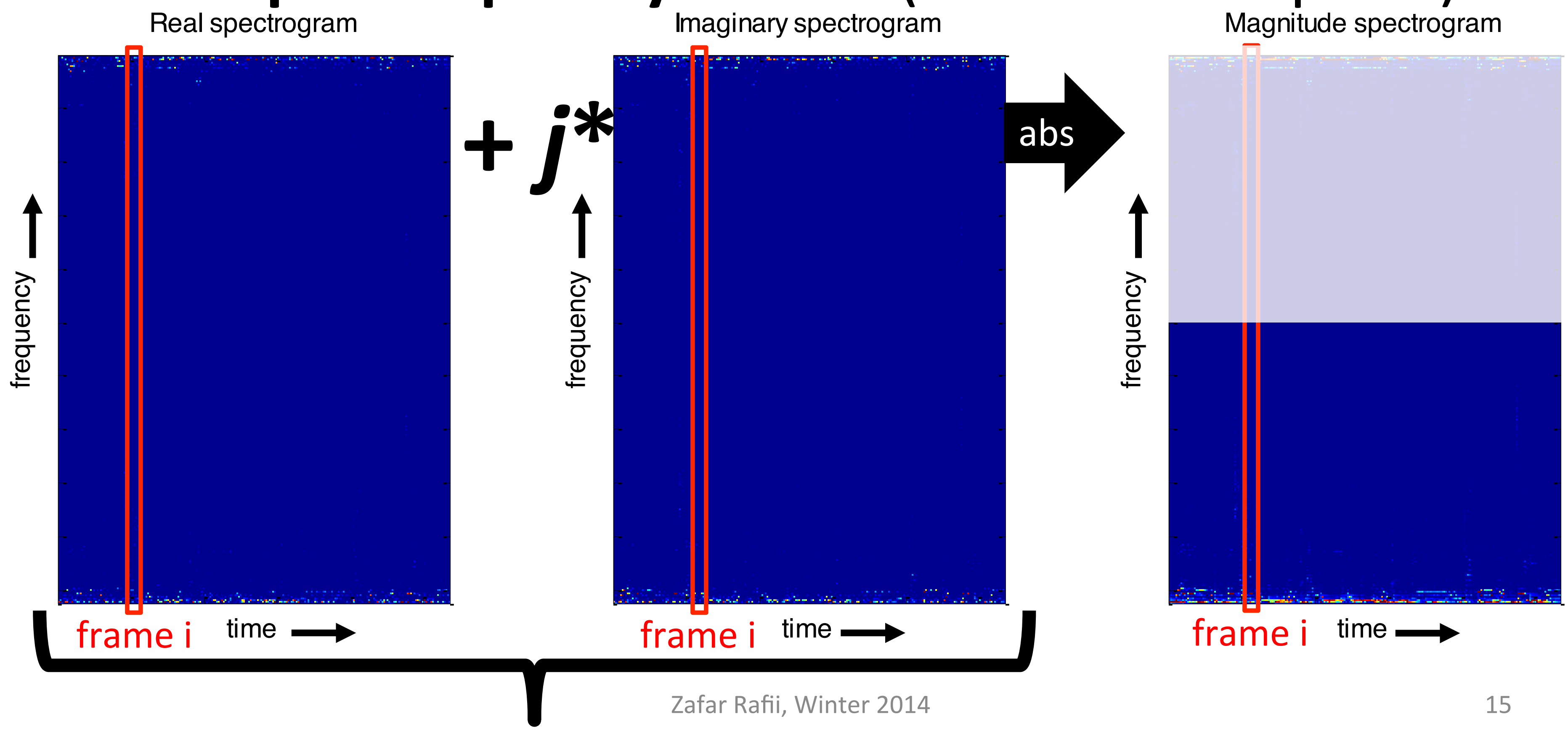
Spectrogram

- Since they are redundant, we can discard the frequency values from floor(N/2)+1 to N-1



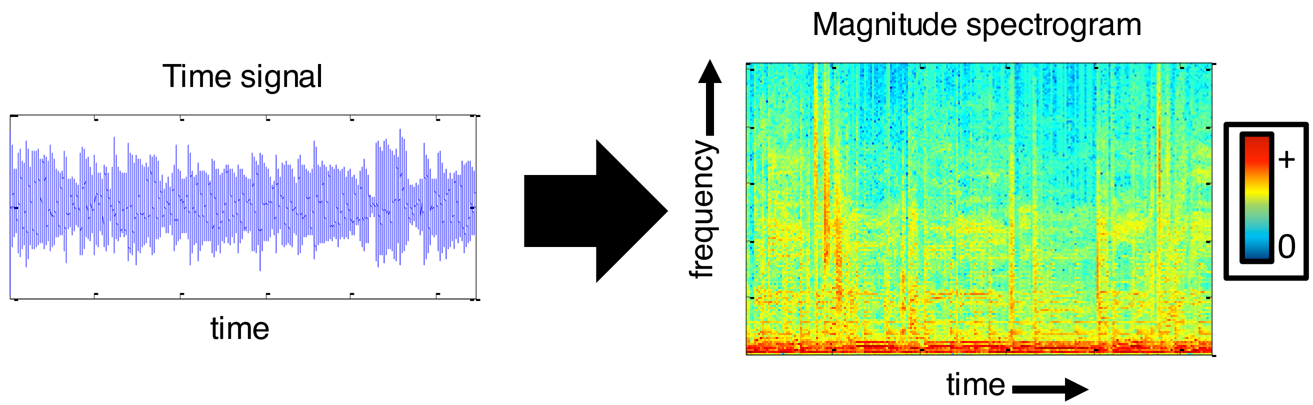
Spectrogram

- The spectrogram has therefore **floor(N/2)+1 unique frequency values** (with DC and pivot)



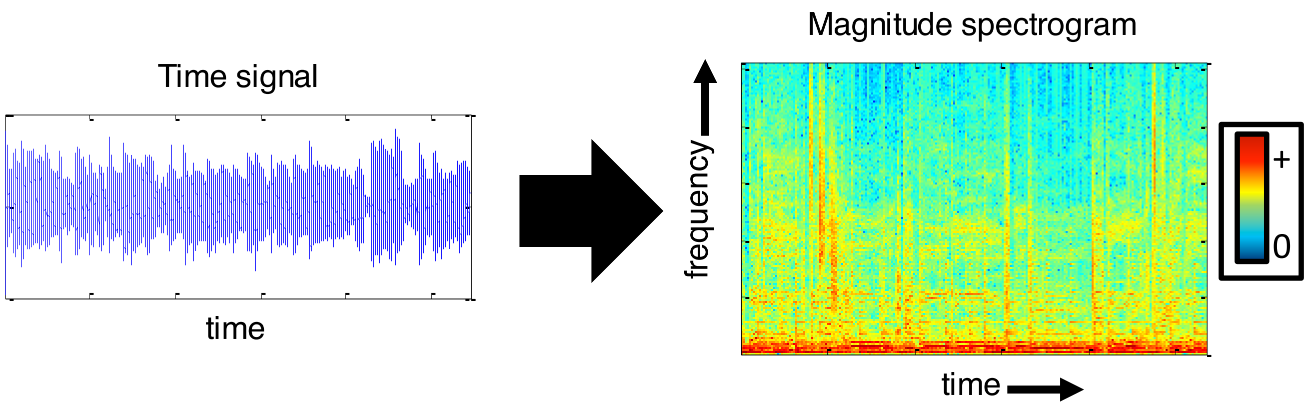
Spectrogram

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



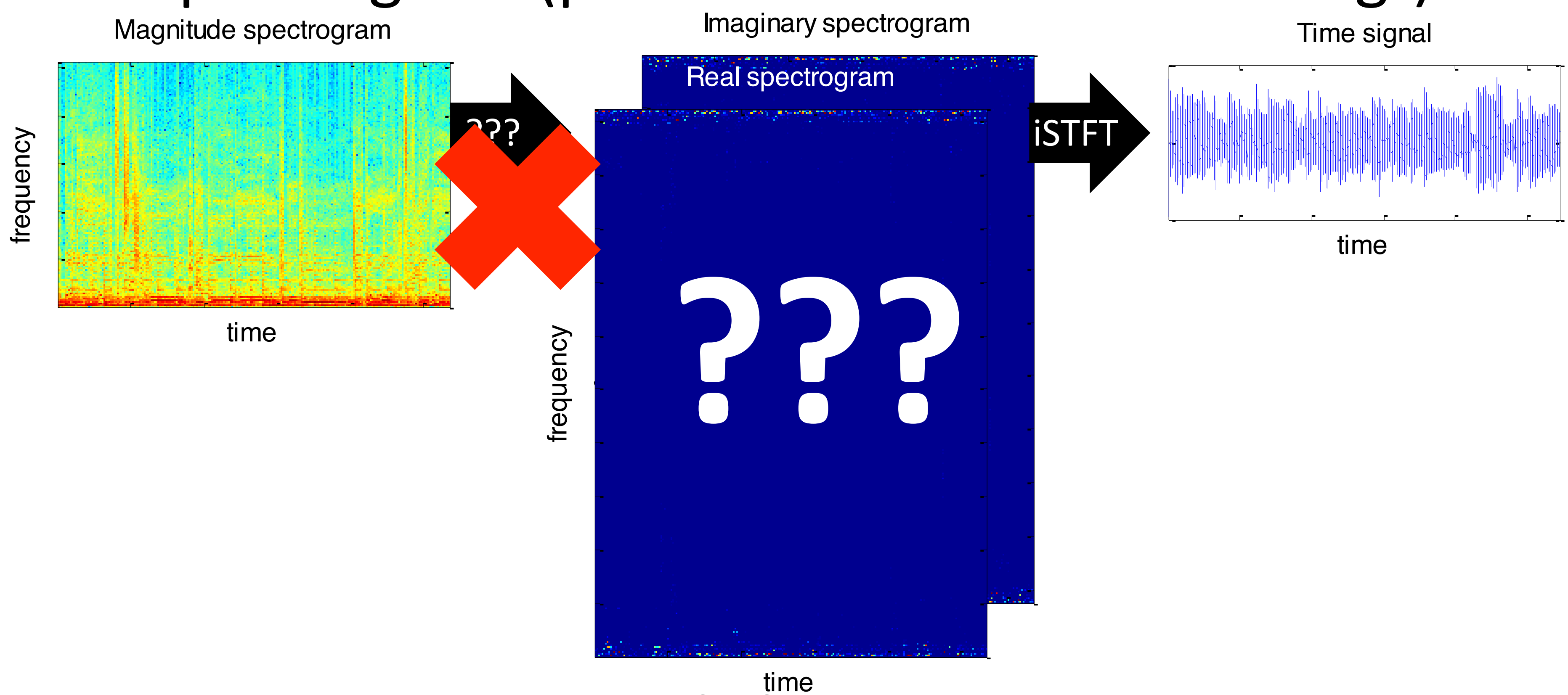
Spectrogram

- 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



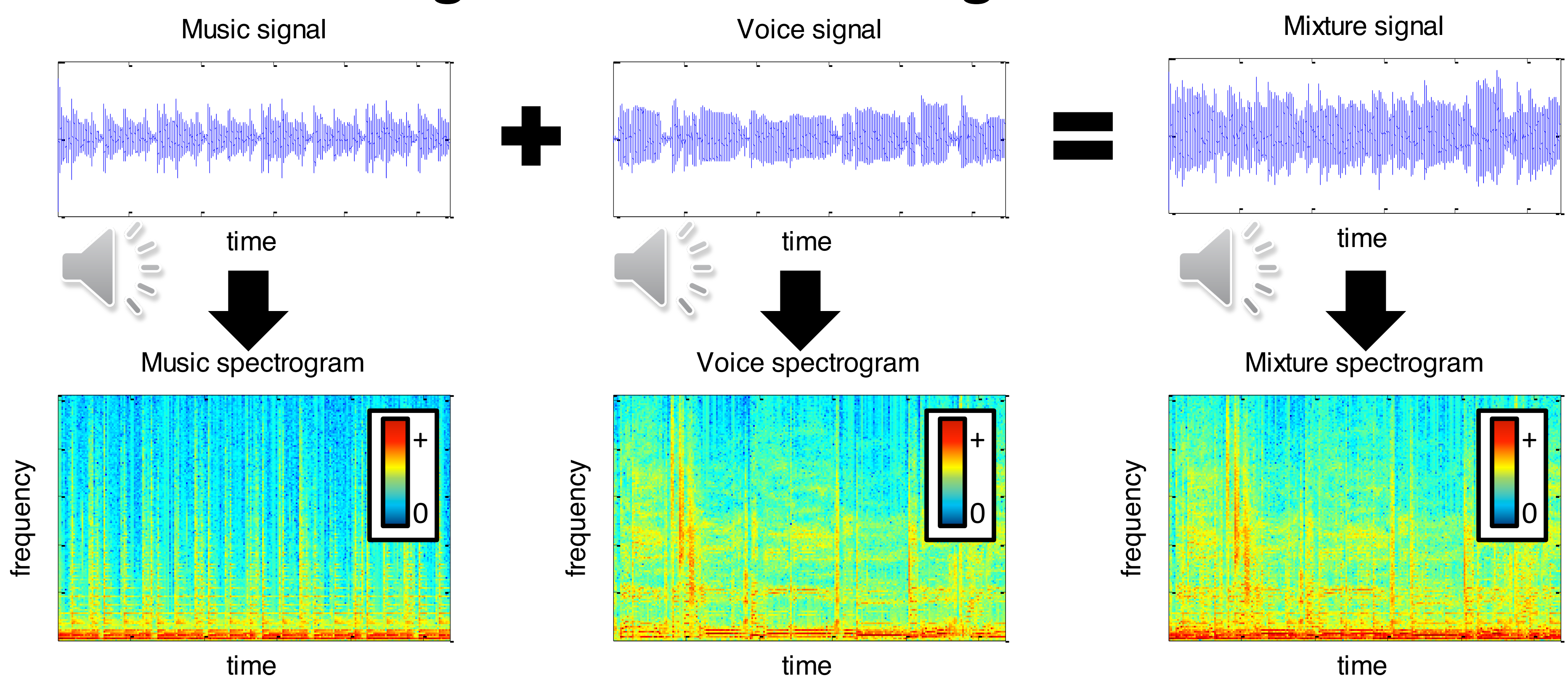
Spectrogram

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



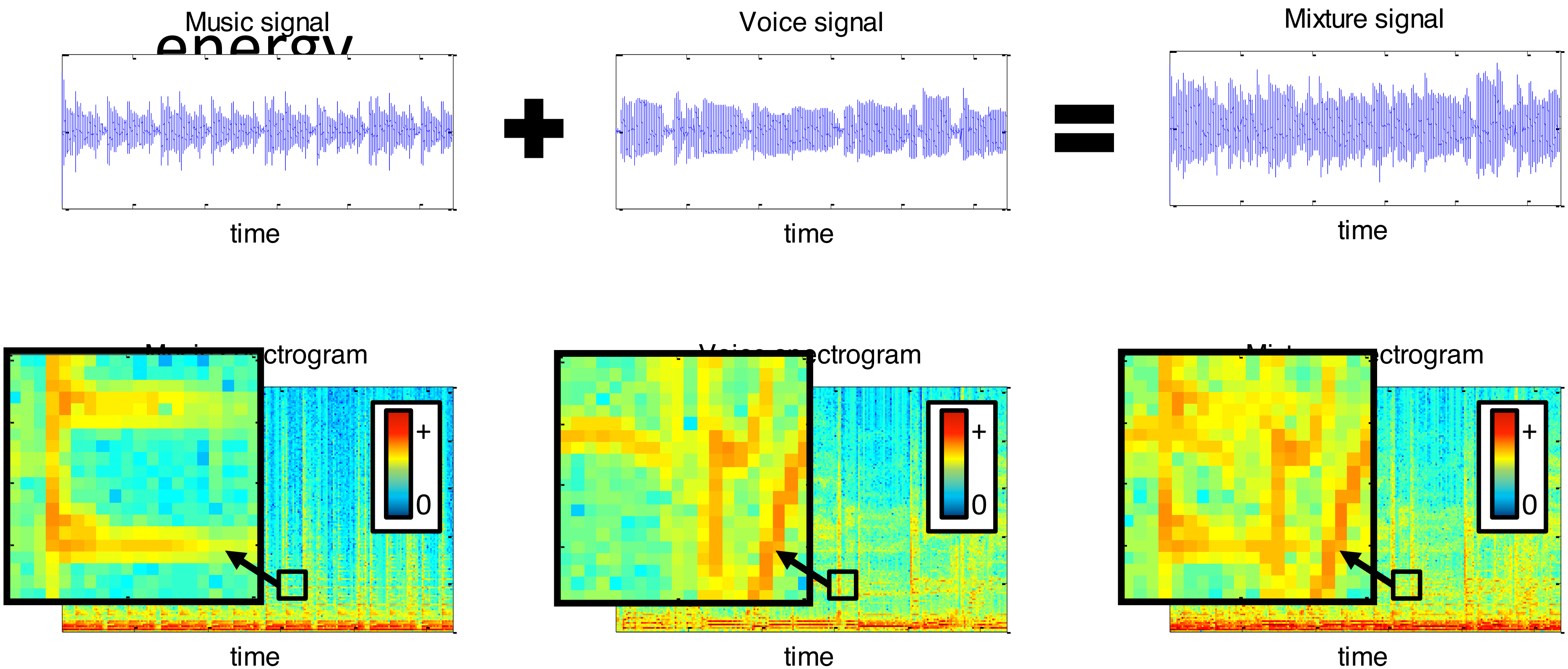
Time-frequency Masking

- Suppose we have a mixture of two sources: a music signal and a voice signal



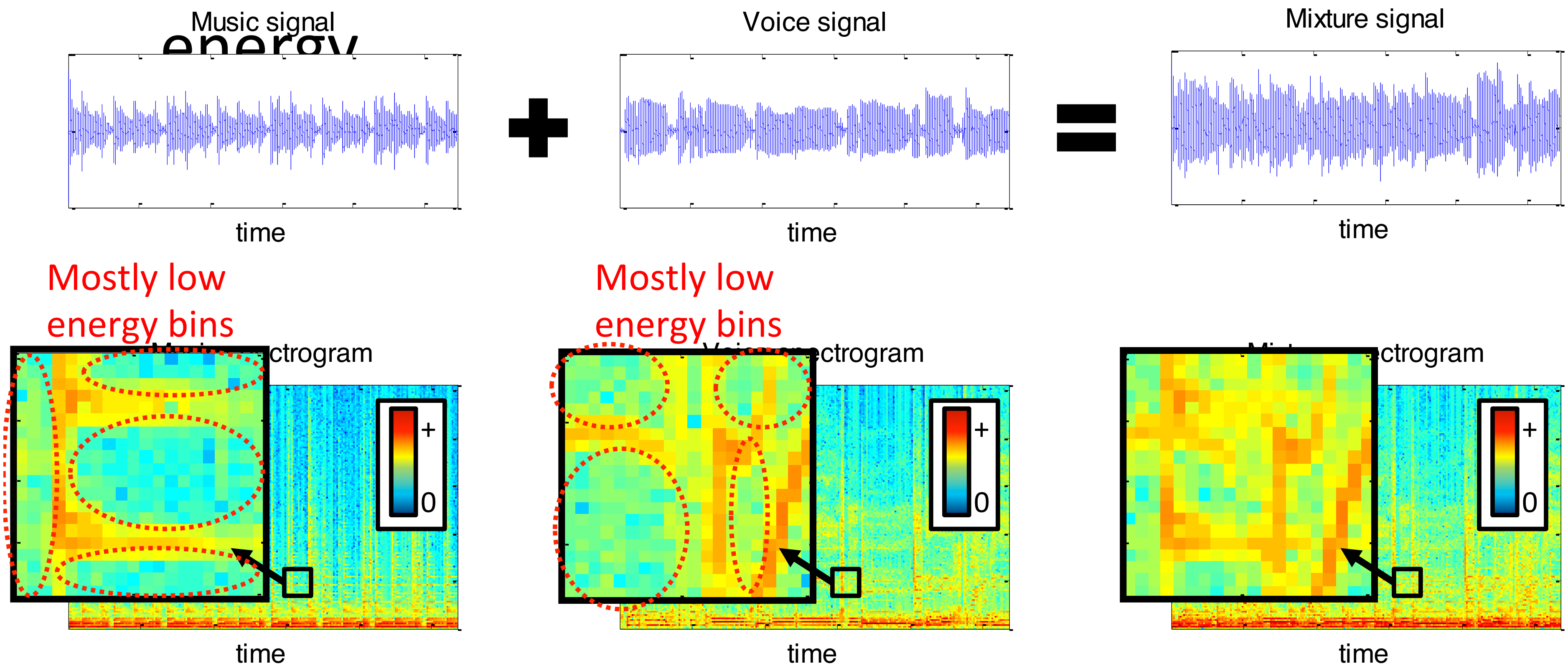
Time-frequency Masking

- We assume that the sources are **sparse** = most of the time-frequency bins have null energy



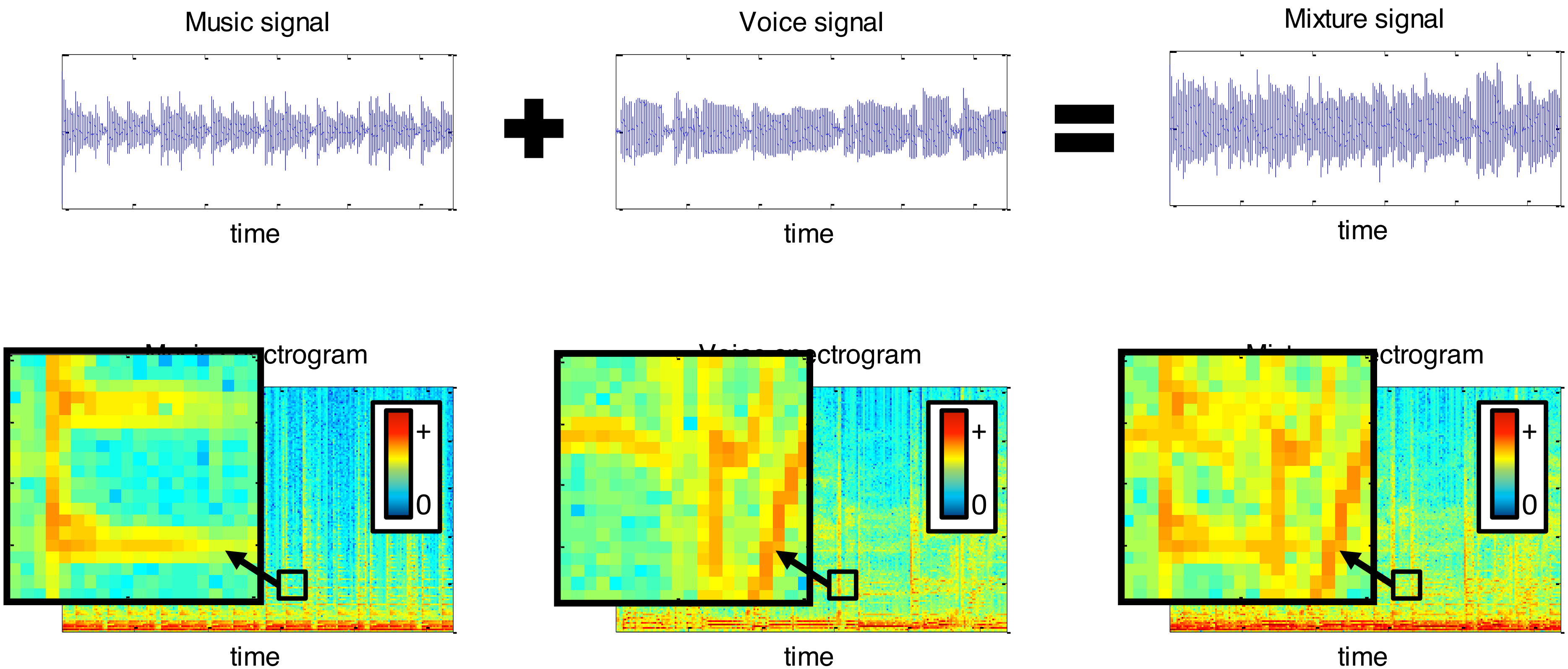
Time-frequency Masking

- We assume that the sources are **sparse** = most of the time-frequency bins have null energy



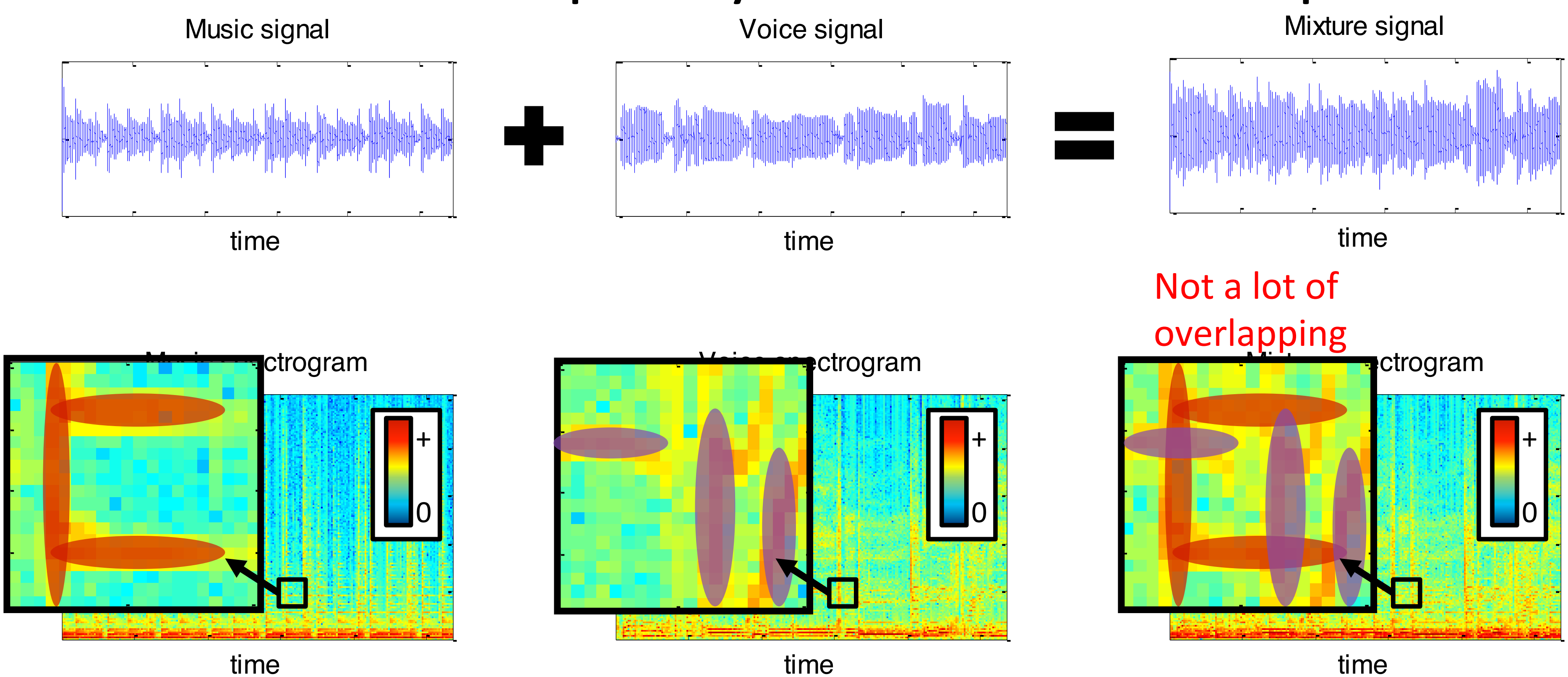
Time-frequency Masking

- We assume that the sources are **disjoint** = their time-frequency bins do not overlap



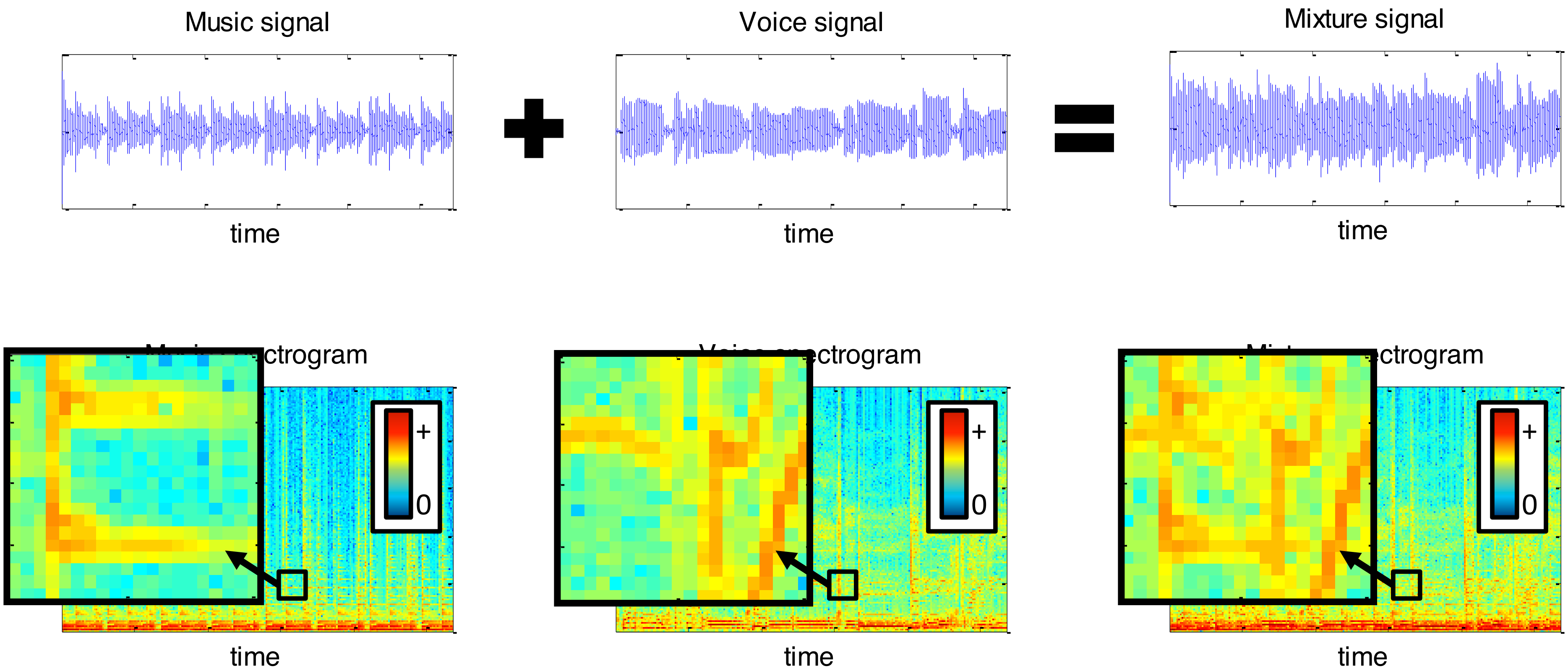
Time-frequency Masking

- We assume that the sources are **disjoint** = their time-frequency bins do not overlap



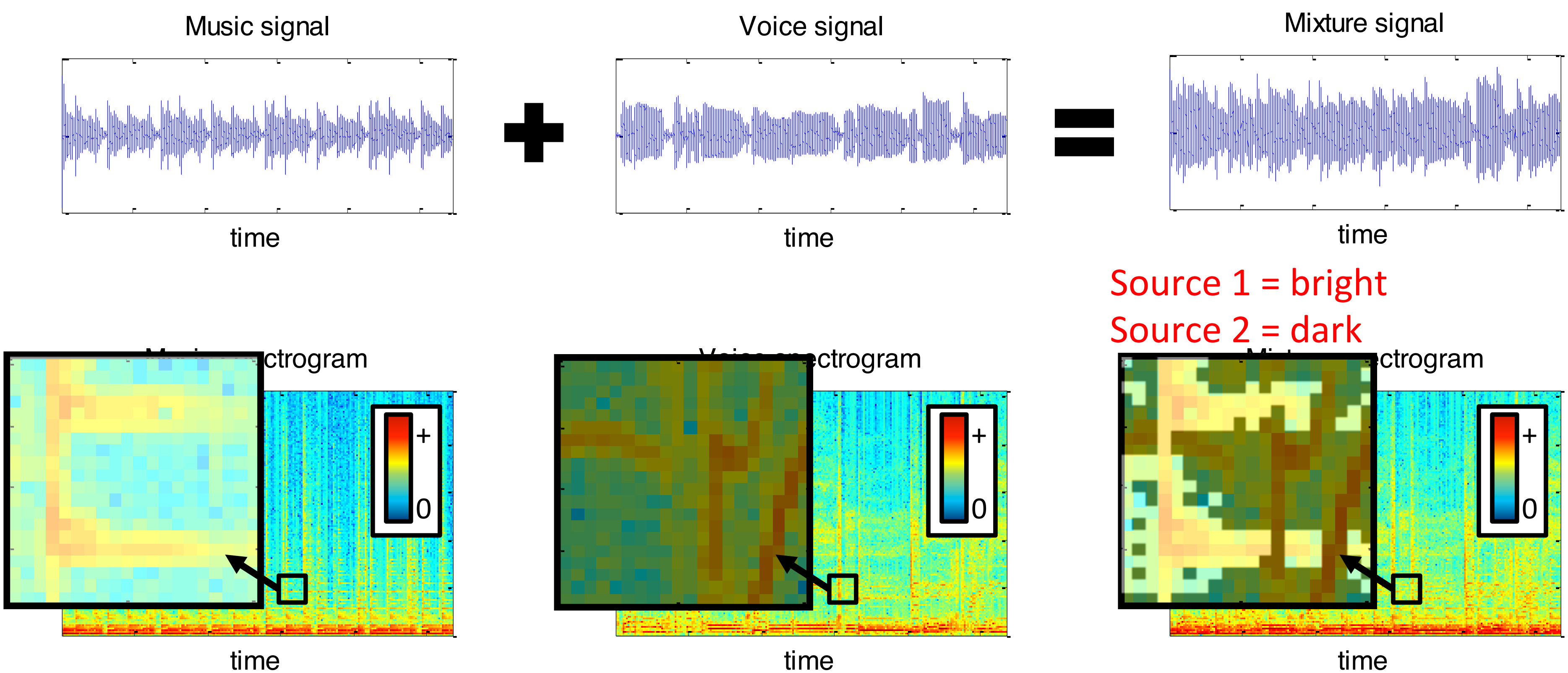
Time-frequency Masking

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



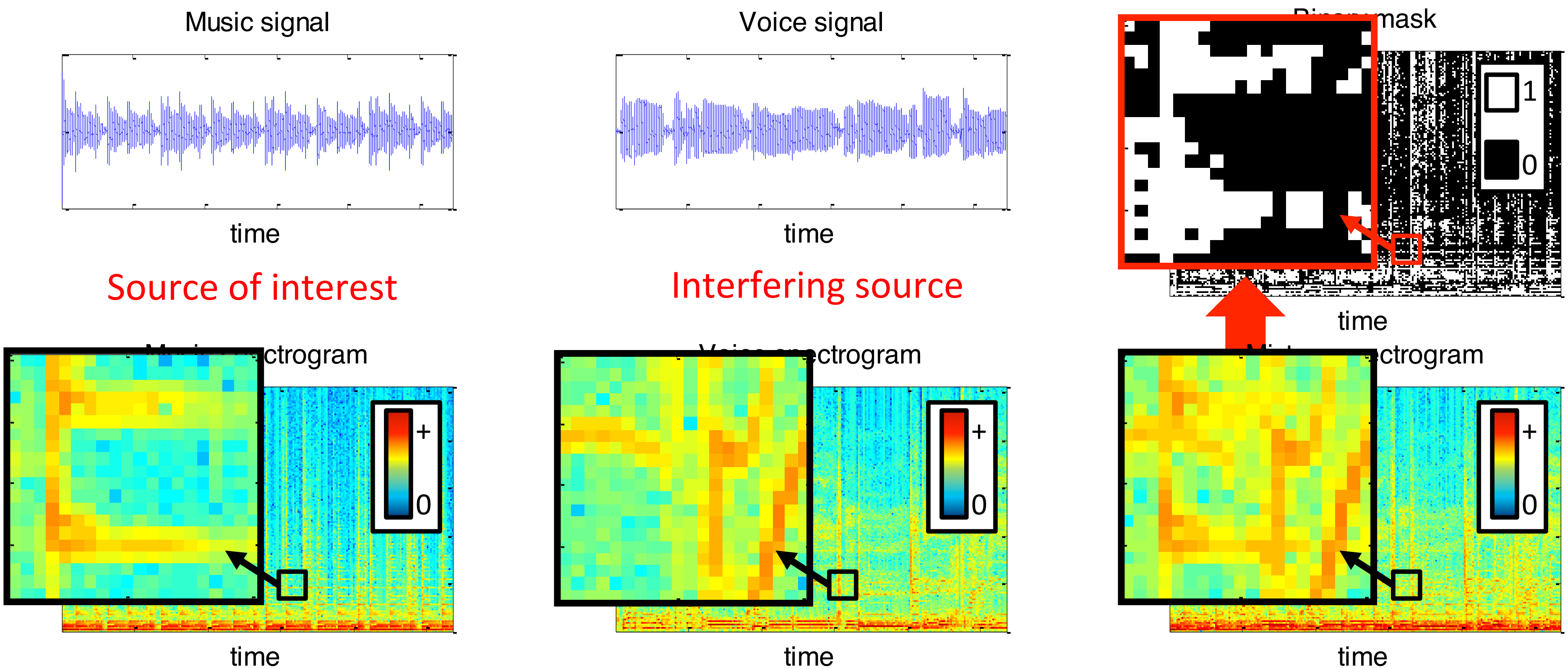
Time-frequency Masking

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



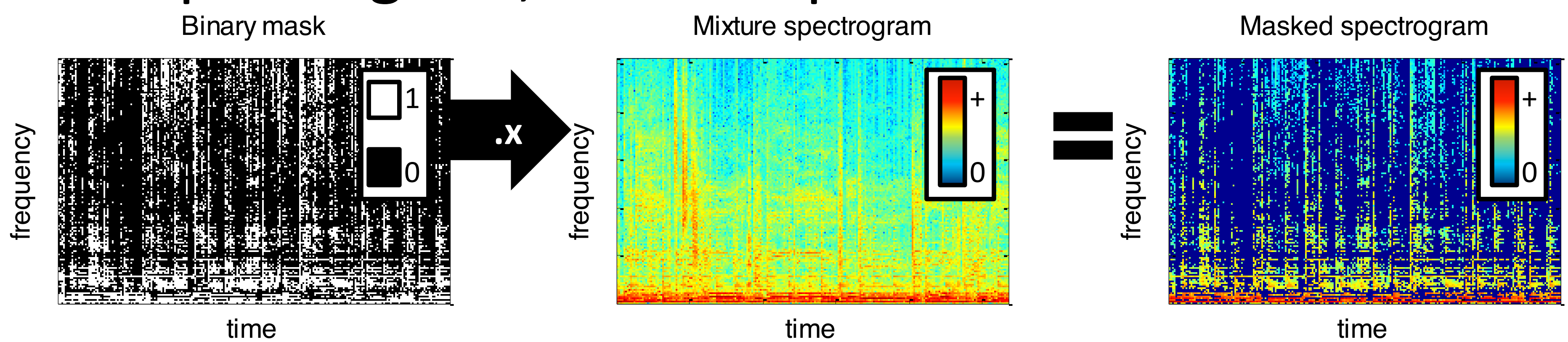
Time-frequency Masking

- Bins that are likely to belong to one source are assigned to 1, the rest to 0 = **binary masking**!



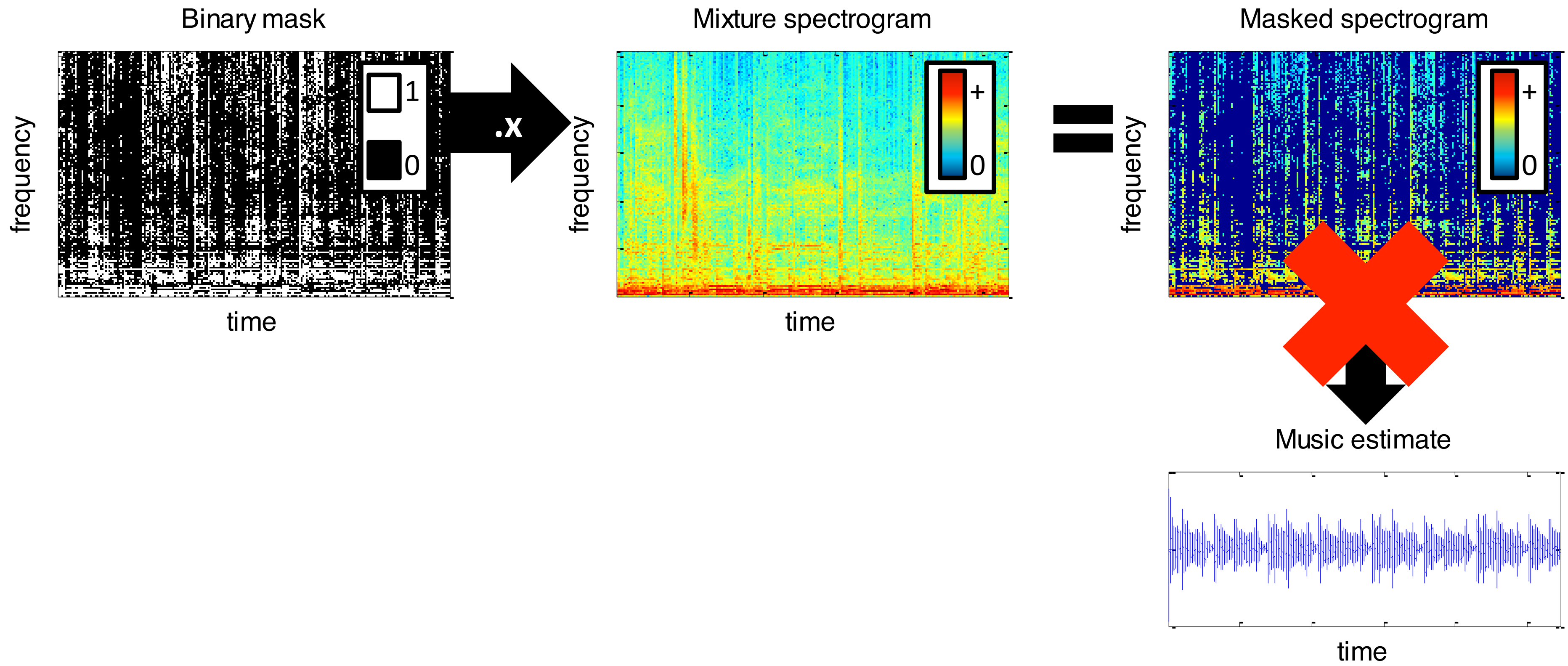
Time-frequency Masking

- By multiplying the binary mask to the mixture spectrogram, we can “preview” the estimate



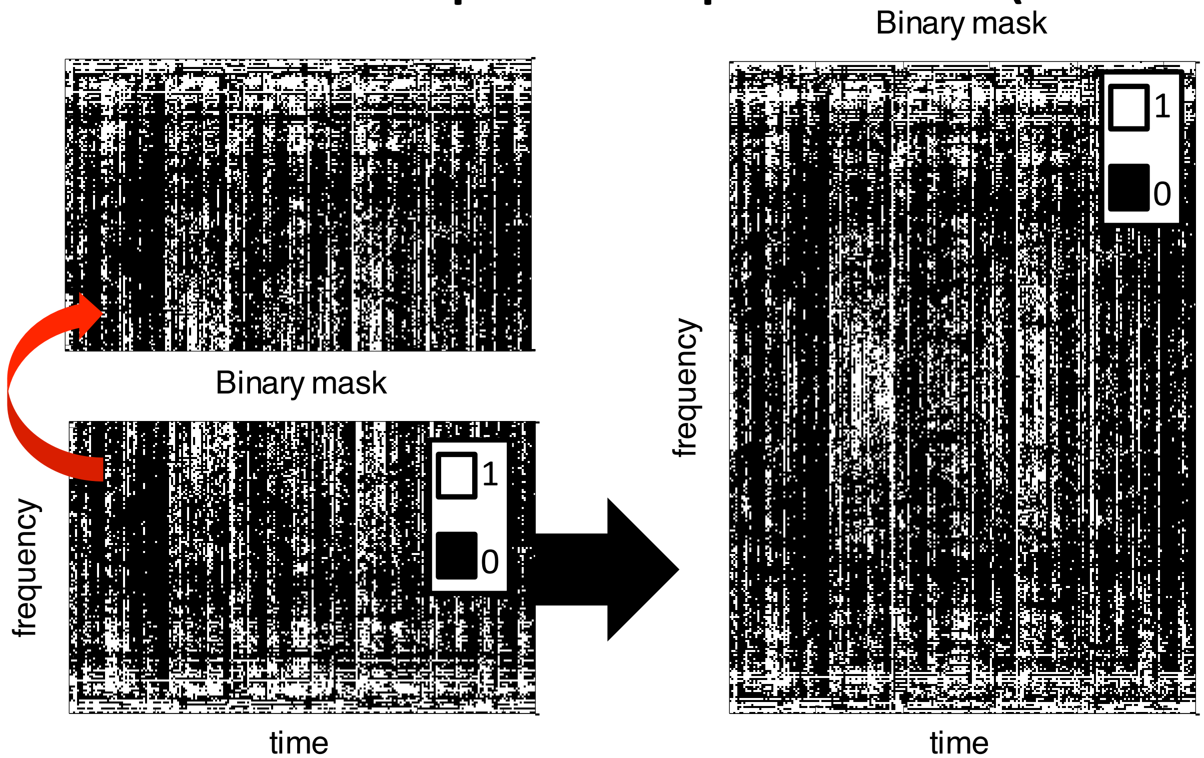
Time-frequency Masking

- However, we cannot derive the estimate itself because we cannot invert a spectrogram!



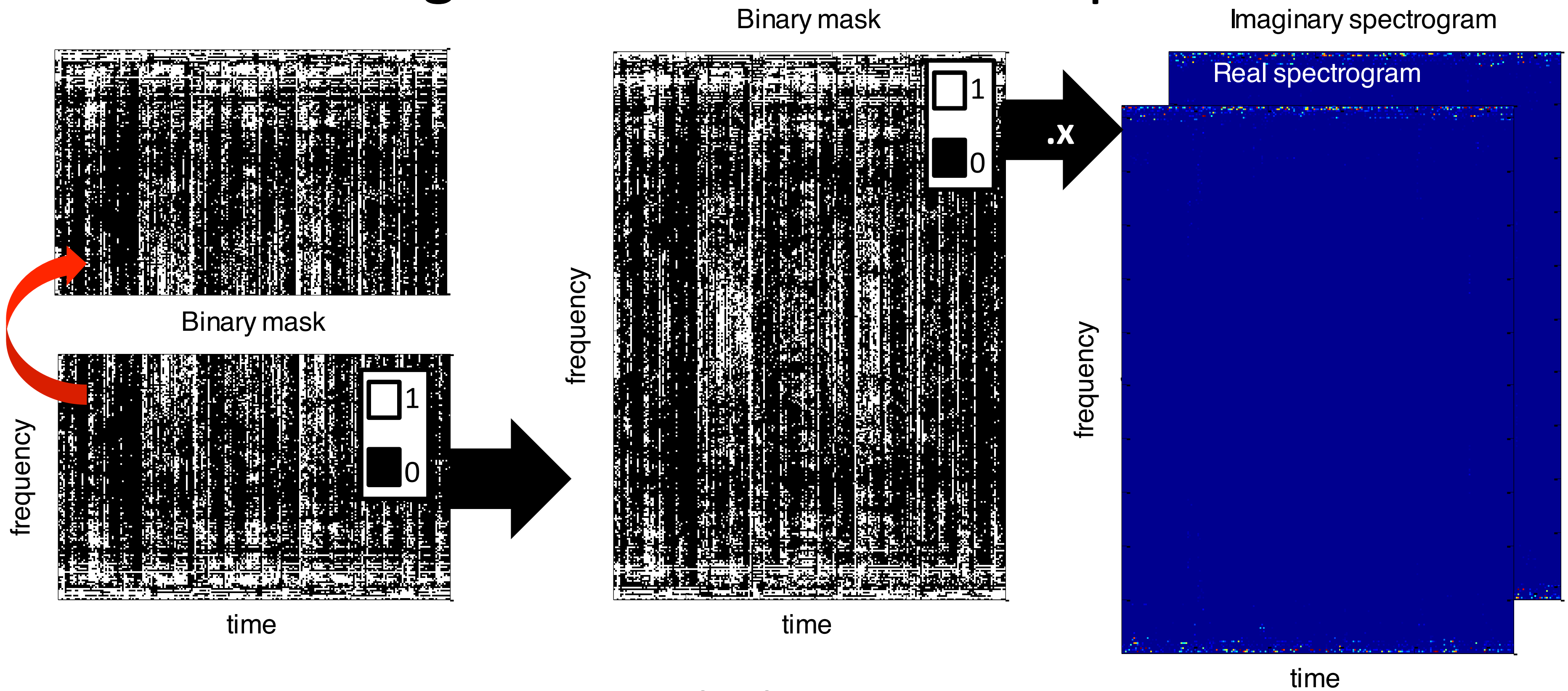
Time-frequency Masking

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



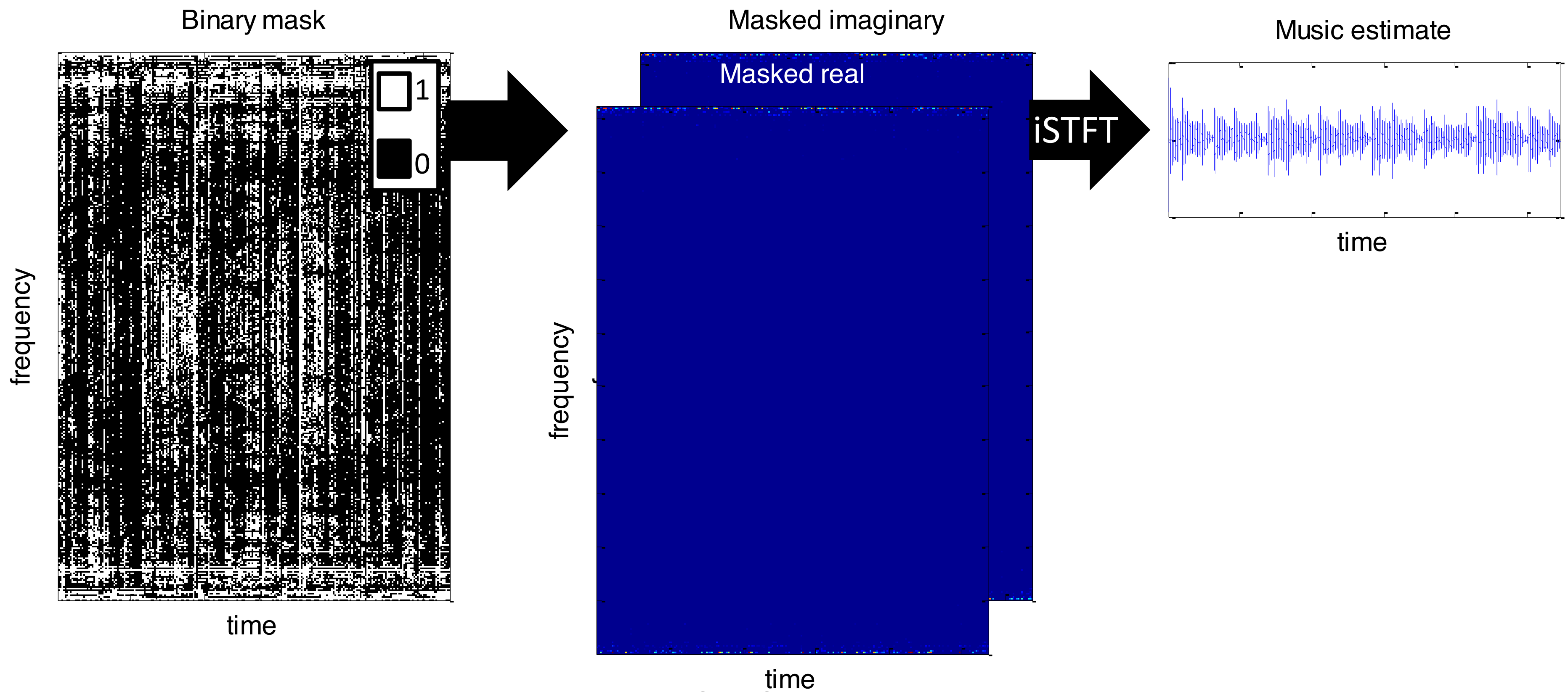
Time-frequency Masking

- We then apply this full binary mask to the STFT using a element-wise multiplication



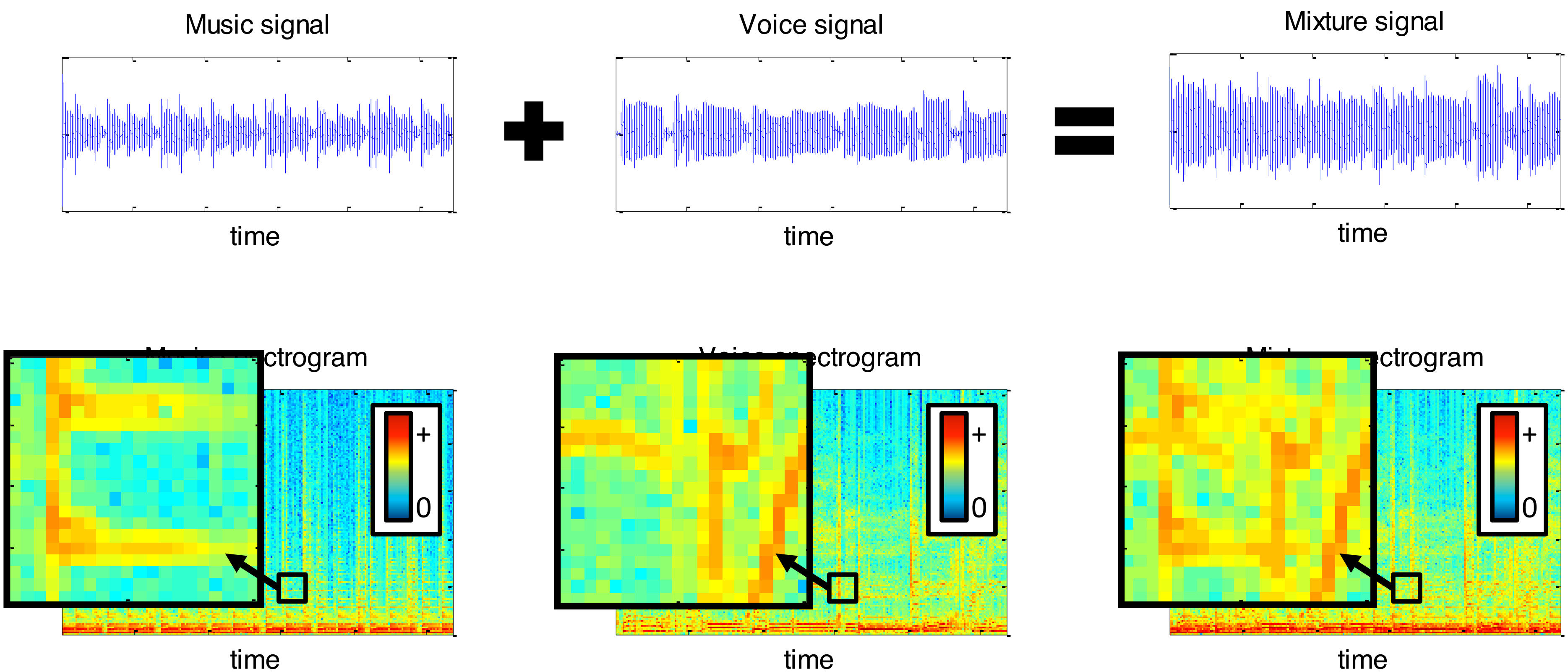
Time-frequency Masking

- The estimate signal can now be reconstructed via inverse STFT



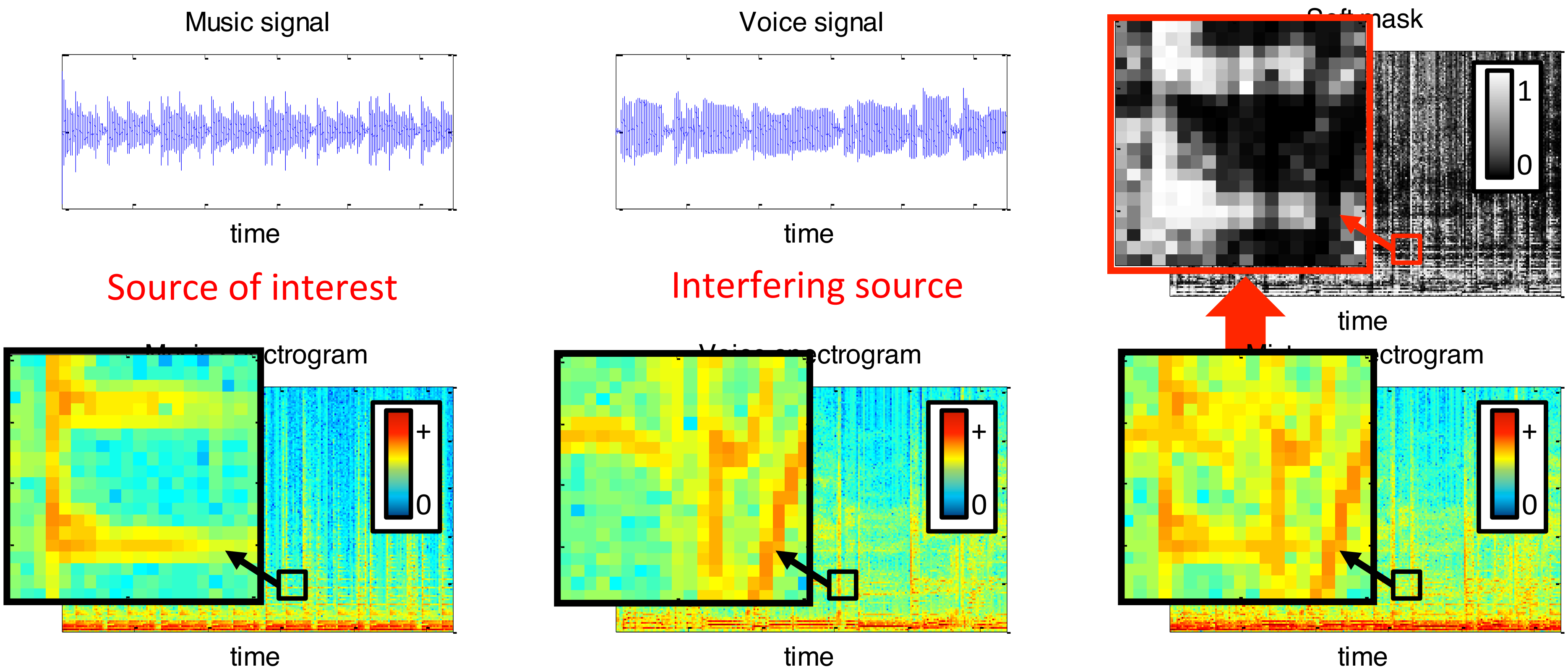
Time-frequency Masking

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



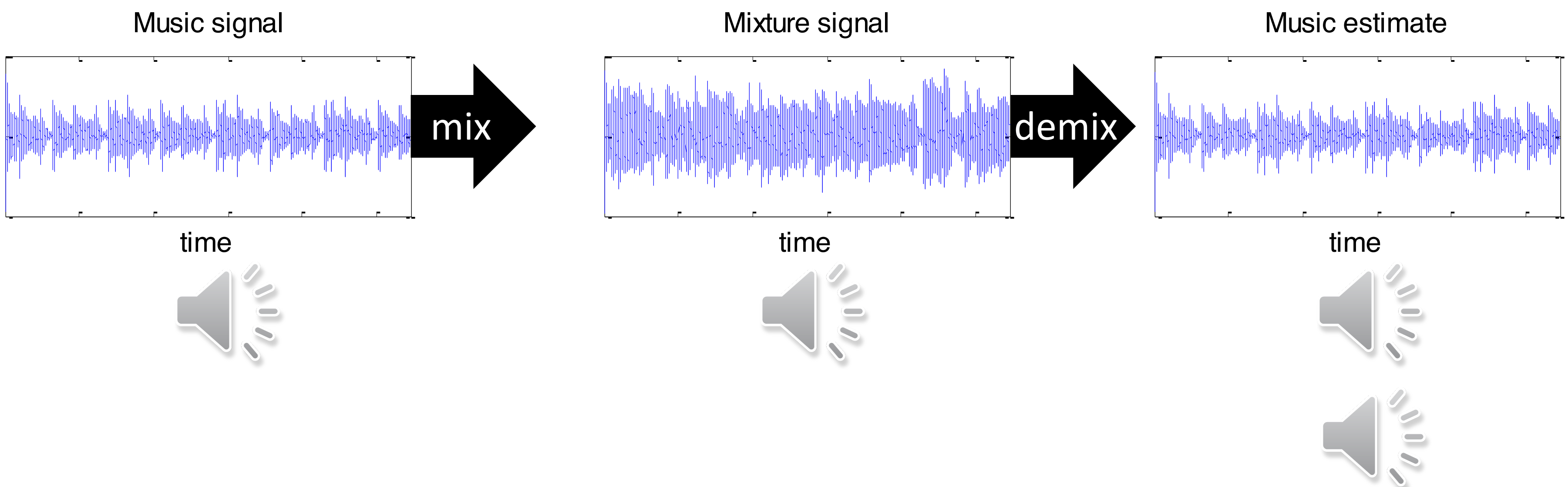
Time-frequency Masking

- 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...