SIGNAL PROCESSING

Understanding overlapping in STFT

Asked 3 years, 11 months ago Active 1 year, 2 months ago Viewed 7k times

4

2

I understand the concept of the STFT. In order to avoid spectral leakage, you use a Hann window that overlaps by 50%. However, the problem that I have, is that the example code online does not account for this.

What I mean is, if the signal is overlapping by 50%, why doesn't anyone add the frequency content of the overlapping sections?

https://kevinsprojects.wordpress.com/2014/12/13/short-time-fourier-transform-using-python-and-numpy/

This is an example of what I mean. It looks like he is grabbing the overlap, but he is not adding the overlap pieces together. I don't see how the frequency content is being saved. He windows a function, does the FFT, then moves over and does the FFT again with the overlapping piece. I feel like he should be adding the overlapping pieces together. Is there something I am missing?

Thank you: John

Edit: I see the first answer, but it is still not hitting home. Hanning windows make the left and right information approach zero. So when it is overlapped, this removes this problem. However, when the FFT is taken, the information of the transform is put side-by-side. Wouldn't this mean, First Window=> upper time series information lower amplitude, Second window =>lower time series information lower amplitude.... This does not seem to solve the issue. You are still losing information with each window... is this not the case?

fft python spectrogram short-time-ft window

Share Improve this question Follow

edited Jul 16 '17 at 10:03
jojek ♦
10.1k 6 30 63

asked Jul 15 '17 at 2:59
John Menoso
51 1 3

Adding 50% of the preceding and 50% of the following window would be equivalent to a form of triangular blur filter. You could do a blur filter, but it's more common not to blur spectrographs, and just plot each spectrograph column as a point sample representing the center of each STFT window. – hotpaw2 Jul 15 '17 at 3:55

I have the same doubt you were having, I do understand how windowing reduces spectral leakage and how overlap is needed then to avoid obtaining a lossy signal. However, as John Menoso was pointing I do not understand why some stft implementations concatenate the overlapped segments side-by-side while others overlap zero-padded portions or even add the overlapped segments. If overlapped segments are concatenated the number of time windows in the stft matrix increases and the stft looks smoother (Time axis is compressed to original duration) What's the difference in between the different methods – Pedro Martinez Lopez Nov 16 '17 at 12:01

@PedroMartinezLopez please do not post comments as answers to questions. Actual answers to the question should only be posted answers. – Peter K. ♦ Nov 16 '17 at 15:41

"spectral leakage" in the frequency domain is not the core reason for overlap-adding segments of the time-domain signal in STFT. the core reason is that the processing one might be doing with the STFT will usually become time-variant and you do not want those parameters varying in time to change suddenly and cause some kinda discontinuity or *glitch* in your output. if spectral leakage was your sole concern, a far better window would be the Kaiser window. but the Kaiser window is not complementary as the Hann window is. – robert bristow-johnson Nov 16 '17 at 20:32

1 and *"Hanning"* is a misnomer. there is no *Dr. Hanning* to name a window after. – robert bristow-johnson Nov 16 '17 at 20:33

3 Answers

Active Oldest Votes

5

> I understand the concept of the STFT. In order to avoid spectral leakage, you use a hann window that overlaps by 50%.

I'm sorry but you have a misunderstanding of spectral leakage in addition to how a **spectogram** should be computed. To be exact you **cannot avoid** spectral leakage completely; all you can do is to make a compromise between the spectral resolution and the spectral leakage: A window that yields less spectral leakage should have smaller side lobes (in its window Fourier transform), but that window will unavoidably smear the input signal spectrum because of the increased main lobe width (of the window Fourier transform) and therefore limit the spectral resolution per given window length.

So you should optimize between spectral leakage and spectral resolution. In order to increase spectral resolution for a given accepted leakage level you can increase the window length. But then this will decrease the time localization of the STFT. And in order to accomadate for this latter side effect, you can use overlapping windows so that the next window does not start from the end of the current window, but begins somewhere in the middle; an accepted choice is 50% overlapping. But you can also use other percentages according to your application as @StanleyPawlukiewicz states in his answer.

Note that 50% overlapp is also an optimal choice to reduce the variance of the estimate of the power density spectrum computed by the **periodogram** averaging applied in overlapping windowed analysis segments (the Welch's method)

Share Improve this answer Follow

answered Jul 15 '17 at 14:41

Fat32
25k 3 19 43

I am trying to understand from this answer whether STFT, Hanning and Welch are compatible. I have read holometer.fnal.gov/GH_FFT.pdf but now it seems that is only applicable to stationary signals - what happens if my signal is not stationary? Can I use Welch? If so what do I average? – Andy Piper Aug 21 '19 at 16:57

2

Your overlap is computed right here, in the `hop_size` variable.

```
hop_size = np.int32(np.floor(fft_size * (1-overlap_fac)))
```

Let's make a small example. You use a 1024 sample fft to compute the STFT of a 8192 long recording. Without overlap, you will get 8 different spectrums all spaced by 1024 sample in time (at fs=100Hz, that would mean 1.024 sec between each spectrum).

With a 50% overlap, you will find 15 spectrums separated by 512 samples (or 0.512 sec at fs=1000Hz).

We could put it this way. As time goes by, you will consider a sample more than once for the spectrum amplitude, so their energy will be considered more than once. Since applying a windows, reduces the amplitudes of the samples, it means that the spectrum energy will be reduced as well. By considering your samples more than once, you compensate for that reduction of energy in the spectrum.

In other words, you take the energy developped by your signal during 1.024 seconds, compresses it to a time period of 0.512 sec. That makes a greater power density.

Here's another example, note that I avoid any mathematics rigors as my goal is simply to put a picture in your mind, not make a proof. Consider 2 seconds of data of a 1W signal with power distributed equally in each frequency bin. Without overlap, you'll find a total energy of:

1. Interval [0s-1s] : 1J
2. Interval [1s-2s] : 1J

With 50% overlap:

1. Interval [0s-0.5s] : 1J

2. Interval [0.5s-1s] : 1J
3. Interval [1s-1.5s] : 1J

You see that we have 2 different spectrograms of the same data, but the total amount of energy in it is different. We indeed added energy to our spectrogram by overlapping the FFTs. This increase of energy in the spectrogram is compensated by the reduction of amplitude of the samples by the windowing operation.

The optimum overlap depends on the windows used, and for Hanning window, it happens to be a perfect 50%.

Hope I could make it a little clearer for you.

Share Improve this answer Follow

edited Nov 16 '17 at 19:55

answered Nov 16 '17 at 19:22

Pier-Yves Lessard
656 4 14

2

From my answer [Short-time Fourier transform Tradeoffs]

> Since the frequency content of overlapping windows are displayed sequentially rather than averaged, I find talking about overlapping windows confusing. I would rather divide the time domain into Frames for which we try to obtain a picture of the frequencies in their range as accurately as possible. This may involve extending the window function past the range of the frames to minimize the scalloping noise/windowing loss tradeoff. However, this will introduce leakage of the frequencies between neighboring frames (reducing time resolution).

Share Improve this answer Follow

answered Apr 14 '20 at 1:20

Tom Huntington
190 13