

Separation of singing voice from music [closed]

Asked 15 years, 10 months ago Modified 10 years, 2 months ago

Viewed 7k times



4



Closed. This question needs to be more [focused](#). It is not currently accepting answers.



Want to improve this question? Update the question so it focuses on one problem only by [editing this post](#).

Closed 5 years ago.

[Improve this question](#)

I want to know how to perform "spectral change detection" for the classification of vocal & non-vocal segments of a song. We need to find the spectral changes from a spectrogram. Any elaborate information about this, particularly involving MATLAB?

matlab

fft

Share

edited Feb 2, 2009 at 17:54

Improve this question

Follow

community wiki
3 revs, 2 users 67%
VIGNESH

5 Answers

Sorted by:

Highest score (default)



4

Separating out distinct signals from audio is a very active area of research, and it is a very hard problem. This is often called [Blind Signal Separation](#) in the literature. (There is some MATLAB demo code in the previous link.



Of course, if you know that there is vocal in the music, you can use one of the many [vocal separation](#) algorithms.



Share Improve this answer

answered Feb 2, 2009 at 21:29

Follow

community wiki
[Nick Haddad](#)



2

As others have noted, solving this problem using only raw spectrum analysis is a dauntingly hard problem, and you're unlikely to find a good solution to it. At best, you might be able to extract some of the vocals and a few extra crossover frequencies from the mix.



However, if you can be more specific about the nature of the audio material you are working with here, you might be able to get a little bit further.



In the worst case, your material would be normal mp3's of regular songs -- ie, a full band + vocalist. I have a feeling that this is the case you are probably looking at given the nature of your question.

In the best case, you have access to the multitrack studio recordings and have at least a full mixdown and an instrumental track, in which case you could extract the vocal frequencies from the mix. You would do this by generating an impulse response from one of the tracks and applying it to the other.

In the middle case, you are dealing with simple music which you could apply some sort of algorithm tuned to the parameters of the music to. For instance, if you are dealing with electronic music, you can use to your advantage the stereo width of the track to eliminate all mono elements (ie, basslines + kicks) to extract the vocals + other panned instruments, and then apply some type of filtering and spectrum analysis from there.

In short, if you are planning on making an all-purpose algorithm to generate clean acapella cuts from arbitrary source material, you're probably biting off more than you can chew here. If you can specifically limit your source material, then you have a number of algorithms at your disposal depending on the nature of those sources.

Share Improve this answer

answered [Feb 2, 2009 at 22:23](#)

Follow



1



This is hard. If you can do this reliably you will be an accomplished computer scientist. The most promising method I read about used the lyrics to generate a voice only track for comparison. Again, if you can do this and write a paper about it you will be famous (amongst computer scientists). Plus you could make a lot of money by automatically generating timings for karaoke.

Share Improve this answer

answered [Feb 2, 2009 at 8:43](#)

Follow



0



If you just want to decide whether a block of music is clean a-capella or with instrumental background, you could probably do that by comparing the bandwidth of the signal to a normal human singer bandwidth. Also, you could check for the base frequency, which can only be in a pretty limited frequency range for human voices.

Still, it probably won't be easy. However, hearing aids do this all the time, so it is clearly doable. (Though they typically search for speech, not singing)

answered [Feb 17, 2009 at 13:21](#)

Share Improve this answer

Follow

community wiki
[bastibe](#)



-1

first sync the instrumental with the original, make sure they are the same length and bitrate and start and end at the exact time and convert them to .wav



then do something like



```
I = wavread(instrumental.wav);  
N = wavread(normal.wav);  
i = inv(I);  
A = (N - i); // it could be A = (N * i) or A = (N + i)  
around  
wavwrite(A, acapella.wav)
```

that should do it.. a little linear algebra goes a long way.

Share Improve this answer

edited Sep 20, 2011 at 16:29

Follow

community wiki

2 revs, 2 users 60%

zakbot

I don't think you understood the question. He is asking about "spectral change detection"; your taking the difference of two wavs does not answer his question. – [Jonas Heidelberg](#) Feb 21, 2012 at 17:00
