

QM-DSP - Feature #761

phase vocoder

2013-10-01 03:45 AM - Matthias Mauch

Status:	Closed	Start date:	2013-10-01
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			

Description

It appears that the phaseVocoder implementation does not actually calculate the phase and amplitude according to the well-known phase vocoder method, which is curious.

The equations that it should implement are, I think, (12) and (13) in

http://www.dafx.ca/proceedings/papers/p_247.pdf

Matthias

History

#1 - 2013-10-01 09:04 AM - Chris Cannam

blimey. Am I reading this wrong, or does this mean the phase & complex domain detection functions don't implement at all the methods described in the literature?!

I didn't implement this particular code (and I do know what a phase vocoder is, he says defensively!). My first assumption when I read this was that the "phase vocoder" in the code was just misnamed, and it was used in a case where the magnitude and instantaneous phase calculation were all that was needed. But I don't think that's true, having checked the original papers from Bello.

Funnily enough I was just starting the process of getting some test coverage across this library -- looks like it really needs it.

#2 - 2013-10-01 09:08 AM - Matthias Mauch

Well, that's certainly what I wanted to say. However, my knowledge of PV is patchy, and -- like you -- I'm not sure what was intended with this code.

#3 - 2013-10-02 02:20 PM - Chris Cannam

It may be that the sums work out the same either way, given what the onset detector does with the phase information, or at the very least that it makes little difference. I'll have to check the figures more carefully, but I am also going to drop in a true phase vocoder (with tests!) to compare against.

I notice that the equivalent MATLAB code (http://code.soundsoftware.ac.uk/projects/davies-beat-tracker/repository/entry/onset_detection_function.m) also doesn't do phase unwrapping.

#4 - 2013-10-02 02:42 PM - Matthias Mauch

I also made my own little implementation.

<http://code.soundsoftware.ac.uk/projects/predomino/repository/entry/PhaseVocoder.cpp>

Works ok for predomino, but

- might still be buggy
- does not actually offer any pure PV implementation, includes some application-specific things, for example A Rating multiplication in <http://code.soundsoftware.ac.uk/projects/predomino/repository/entry/PhaseVocoder.cpp#L184>

#5 - 2013-10-02 03:02 PM - Chris Cannam

I wonder how many implementations there are on this site -- and of other common things like basic FFTs of course.

(e.g. every build of Sonic Visualiser has at least two different phase vocoder implementations in it -- one for calculating spectral peak locations in the peak-frequency spectrogram and one in the timestretcher library.)

#6 - 2013-10-02 03:36 PM - Matthias Mauch

Interesting.

I would have liked to use an existing implementation. In fact, that's why I looked at the qm-dsp code.

Does the SV one have amplitude correction as well (I haven't implemented that yet)? (Or should I google that ?:)

#7 - 2013-10-03 01:14 PM - Chris Cannam

I've pushed this stuff to the pvoc branch.

The answer to the original question, then, is that the PhaseVocoder class does now implement a phase vocoder. At least in that branch.

The answer to the original implied question is that, while it's nice that the class called PhaseVocoder now implements a phase vocoder, it doesn't actually make any difference in this use -- the phase deviation modulo 2π which is used in the onset detector is the same regardless of whether the phase is unwrapped. If you want to test it in practice, change `m_thetaAngle` to `m_unwrapped` in `DetectionFunction.cpp` lines 167 and 171.

#8 - 2019-05-15 08:59 AM - Chris Cannam

- *Status changed from New to Closed*