Final Year Project Presentation
2016-2017 Spring - KY1602

Automatic piano reduction (backend)
Chord identification

Supervisors:Professor Yip Yuk Lap, Kevin
Professor Lucas Wong
Students:Wong Cheuk Bun (1155048570)
Yip Wai Man (1155047486)

©YalorɿoIɐY

Automatic Piano Reduction
Chord Identification

Quick Revision

Reduction is a process that **transforms complicated score into simpler**
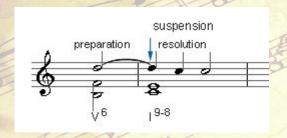
score

Chord Identification

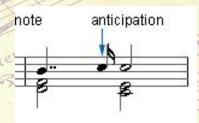Improve performance by running Harmonic Analysis

1. Supplementary Music Theory
2. Selectable Features
3. Experimental Result
4. Limitation
5. Future Development

Supplementary Music Theory

Non-harmonic Note

- notes that not belongs to the chord
- may happens on both on beat and off beat
- increase the complexity of chord identification

Chord Function

- 3 types of function: Tonic(T), Subdominant(S), Dominant(D)

Tonic
- Most stable
- Can express the tonality most

Dominant
- Most unstable
- Tendency to T

Subdominant
- A bit unstable
- Bridge between T and D

- Progression follows rules, e.g. D->T
- Great hint in doing progression recognition

©YalorɿolɒY

Selectable Features

Selectable Features

Three modes of target IntervalType

- OnBeat
- AllIntervalType
- ChangedBaseline

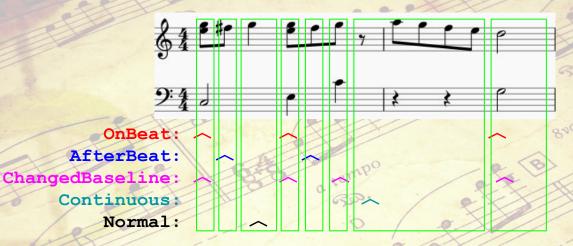Three choices of ProgressionFeature

- VtoIProgression
- ChordFunction
- FirstComeFirstServe

Four combinations for each modes

- ChordFunction
- VtoI + ChordFunction
- FirstComeFirstServe
- VtoI + FirstComeFirstServe

IntervalType

ChordInterval are divided by event, concurrently playing notes

OnBeat:

AfterBeat:

ChangedBaseline:

Continuous:

Normal:

Three modes of target IntervalType

OnBeat

    Select only the ChordInterval that type is OnBeat

    Music usually put emphasis on Strong Beat

ChangedBaseline

    Select only the ChordInterval that type is ChangedBaseline

    Baseline is commonly used to build sense of chord

AllIntervalType

    Select all ChordInterval except type of Continuous

Three choices of ProgressionFeature

VtoIProgression

Special emphasis is put on V-I Chord progression in classical music

ChordFunction

Select chord that make good sense in chord function to have

higher priority

FirstComeFirstServe

Select chords that make sense in the progression table

	A	B	C	D	E	F	
1	KEY	I	I6	I64	bII	bII6	II
2	I		Yes	Yes	Yes	Yes	Y
3	I6	Yes		Yes	No	Yes	Y
4	I64	Yes	Yes		No	Yes	Y
5	bII	No	No	No		Yes	N
6	bII6	No	Yes	Yes	Yes		N
7	II	No	Yes	Yes	No	No	

©Yalorıoɪɒʏ

Selectable Features

Why ?

Not all musics are applicable to One Single Rule of progression

Selectable Features

Experimental Results

Canon in D

- Excerpted bar 3-6
- Chord progression (in D major):

  I-V-VI-III-IV-I-IV-V (repeat till the end)

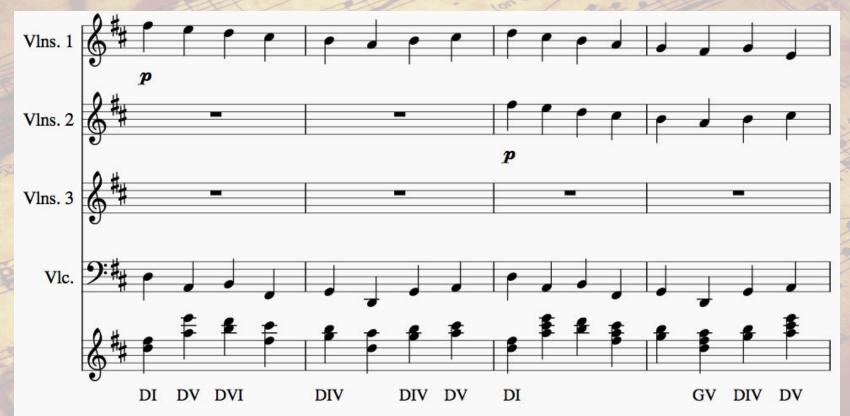| IntervalChoice	ProgressionFeature	Total Correct	Partial Correct
ChangedBaseline	ChordFunction	12/16 (75%)	0/16 (0%)
ChangedBaseline	FirstComeFirstServe	16/16 (100%)	0/16 (0%)
ChangedBaseline	VtoIProgression, ChordFunction	9/16 (56.3%)	1/16 (6.3%)
ChangedBaseline	VtoIProgression, FirstComeFirstServe	11/16 (68.8 %)	1/16 (6.3%)

Chord Function

First Come First Serve
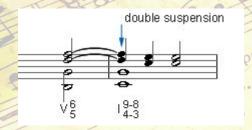
Chord Function, V to I Progression

First Come First Serve, V to I Progression

Limitation

Sampling range of Interval

- First Term choose between by beat and by measure
- Only one single event is used to calculate chords at present
  - On beat non-harmonic notes are not considered
- Time complexity ↑↑ if consider all events consider at the same time

Choosing suitable Features

Greedy Algorithm is applied in designing progression

    turn out that it is not suitable

    running music sequentially may not have the best answer

Example

    V - I progression vs Subdominant,

    Subdominant can be used to release tension

Tonality Change

- Tonality change is not considerd at the moment
- Can be easily achieved
- Time complexity ↑↑↑↑

```
Measure:  0  Interval:  0
       Start at  0.0 End at  0.5
       Type: OnBeat ChangedBaseline

       Note List:  ['C', 'E', 'G']

       Total match chords are:
Dm :  ['VII']
Bm :  ['bII']
Fm :  ['V+']
C :  ['I']
B :  ['bII']
E :  ['bVI']
G :  ['IV']
F :  ['V']
Em :  ['VI']
Cm :  ['I+']
Am :  ['III']
Gm :  ['IV+']

       Exact match chords are:

       Possible match chords are:

       Equivalent Chord:
0 : ['Dm VII', 'Bm bII', 'Fm V+', 'C I', 'B bII', 'E bVI', 'G IV', 'F V', 'Em VI', 'Cm I+', 'Am III', 'Gm IV+']
```

Future Development

- Better algorithm to sort out non-harmonic notes
  - Efficiently
  - Maintain accuracy
- Algorithm to detect music type to choose suitable features
- Algorithm to sort out possible tonality change
- HarmTrace
  - Non sequential (functional programming)
  - http://dreixel.net/research/pdf/fmmh.pdf