Chordination

Jacqui De Sa | CK Ong | Zeo Liu

© TemplatesWise.com

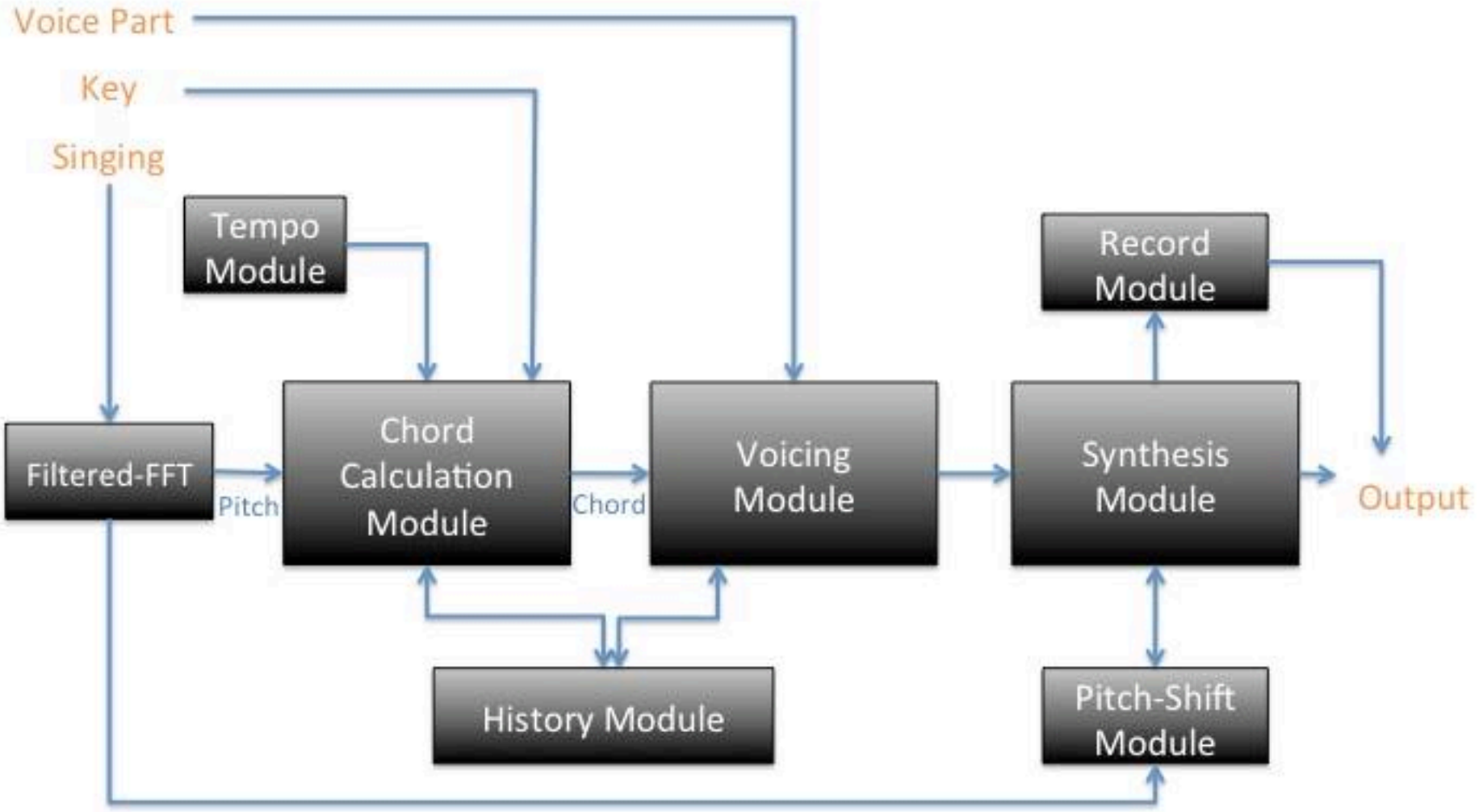
Real Chords. Real Time.

Chordination calculates a harmony and plays it with you in real time.

Motivation

- Make arranging more accessible – no music theory knowledge needed!
- User can focus on melody while harmonization is machine-driven
- No need to pre-record a song before generation chords
- Real-time inspiration

Chordination Block Diagram



Filtered FFT Module

Pitch From Voice

- Reduce Noise (Low-Pass Filter)
- Get Pitches (Fourier Transform)

Tempo Module

Time between Chords

- Takes user input about how regular interval indicating how often chords should change
- MVP implementation uses switches to select pre-defined values
- Extended implementation would allow users to clap in or sing in a beat indicating the tempo.

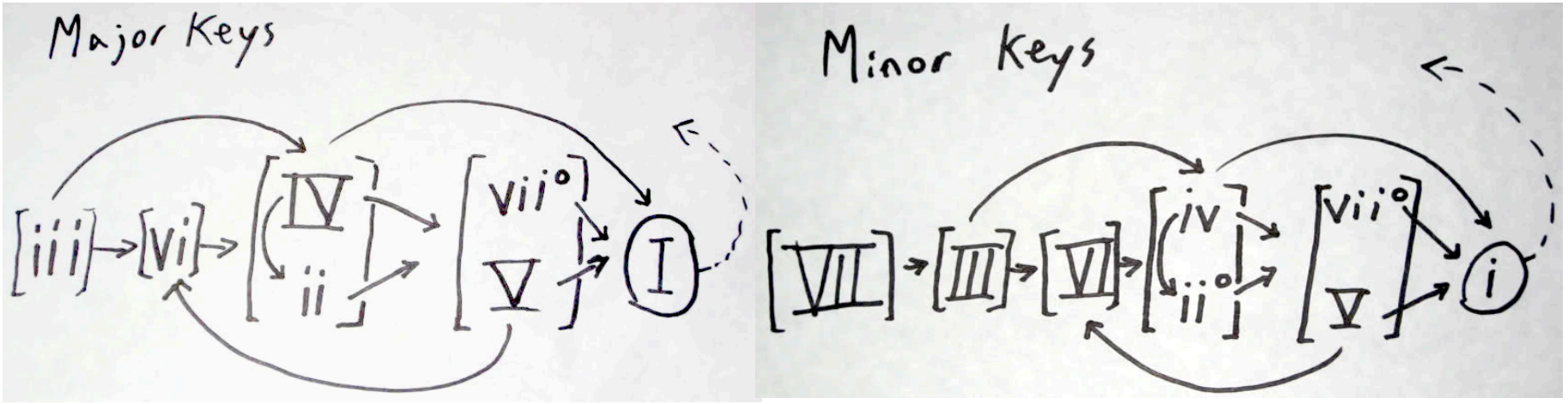
Chord Calculator

Possible Chords

- Calculates next chord based on note, key, tempo, and chord history
- Calculates matching note from pitch (Filtered FFT Module)
- Only calculates chords on chord change time (from Tempo Module)

Chord Calculator: Possible Chords

- Model chord progression as a state machine that follows traditional music theory transition rules.



- Passes chord onto Voicing Module to convert from chord into playable notes

Voicing Module

Placing Notes in a Chord

- Finds out which exact notes to play
- Input/Output
- Organization of internal modules (validity check)
- Uses established theory
- Important considerations
  - Voice part
  - Chord priority

Synthesis Module

Integrate & Output to Speaker

- Shift
- Integrate
- Record

Evaluation Matricies

- Error
- Lag Time
- Correctness of Output Chord

Timeline

- **Week 1:** Finalize ideas & plan for module details
- **Week 2:** Start coding & implement test benches
- **Week 3:** Debugging, test the modules
- **Week 4:** Finish testing and connect modules. Implement extended functionality.

Questions?