live coding music

– robotic pianos –

vitor guerra rolla
postdoctoral fellow

impa

VisgrafLab
summary

- introduction
- chuck programming language
- MIDI protocol
- pre-loaded code
- robotic pianos => instruments
- live coding
live coding

- performing arts
- on the fly or real-time
- musician(s) + computer(s)
- video animation or image
- pedagogical / learning purpose
chuck

• Ge Wang

• real-time sound synthesis

• time-based programming => now

• operators: =>, %

• MIDI compatible
musical instrument digital interface (MIDI)

Std.mtof()
pre-loaded code

- root key \rightarrow (key.ck)
  \hspace{.5cm} changeRootNote();

- musical scales \rightarrow (scales_lib.ck)
  \hspace{.5cm} quantize();

- monitoring \rightarrow (monitor.ck)
  \hspace{.5cm} monitor();
pre-loaded code

- (scales_lib.ck) => function quantize(C, aeolian)

octave

C aeolian scale
robotic pianos

- composed by Andrew Sorensen
- impromptu language $\Rightarrow$ $\lambda$
- two piano hands + guitar + hi-hat
løft-hand – step 1

\[ 2 \times G + A + B \]

(4 beats)

0.5 sec → time interval between notes
mutable

key
changeRootNote()

harmonic progression

lefthand – step 2

0.5 sec → time interval
0.25 sec offset → step 1

One note per measure
(8 beats)

C# D# E F# G# A#

C D E F G A B

48 50 52
righthand

\[(3 \times \cos (x / 2) + 5) \times (\cos ((7 \times x)/3) + \text{key.root} + 24)\]

\[x = 90^\circ\]

quantize(wave, scale, key.root);
guitar

- triangle wave form
- ADSR envelope

volume

max

0,7

On

0,05

0,3

Off

0,5
time (seconds)
hi-hat

- noise
- ADSR envelope
- high pass filter

```
random();
```
live coding – robotic pianos
thank you