

Computer und Musik

Jan Dreier, Philipp Kuinke, Peter Rossmanith

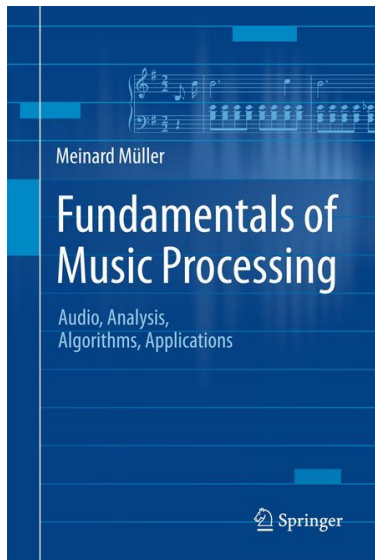
13. September 2017

Die Thematik

Musik ist ein alltäglicher Bestandteil des Lebens, der Menschen seit tausenden von Jahren begleitet. Computer hingegen sind im Gegensatz dazu vergleichbar jung. Wie können wir Computer dazu benutzen um Musik zu erkennen, verändern oder erzeugen?

– *Philipp Kuinke*

Material



Erhältlich bei uns oder in
der Informatik Bibliothek

Organisatorisches

- ▶ Recherche
 - ▶ Startpunkt: Kapitel aus Buch oder Quelle von uns.
 - ▶ Es wird erwartet, dass ihr weitere Quellen bei Bedarf selbstständig sucht.
 - ▶ Quellen von uns könnt, müsst ihr aber nicht verwenden
- ▶ Vorträge
 - ▶ Eine Person: 25 Minuten
 - ▶ Zwei Personen: 35 Minuten
 - ▶ Danach kurze Diskussion
 - ▶ Deutsch oder Englisch
- ▶ Ausarbeitung
 - ▶ Maximal 10 Seiten
 - ▶ Deutsch oder Englisch

Wer Feedback vor dem Vortrag/Ausarbeitung möchte,
rechtzeitig eine Email schicken!

Zeitplan

Circa 2 Vorträge pro Termin.

- ▶ 16.10. (Einführungsvorträge)
- ▶ ...
- ▶ 04.12
- ▶ 11.12
- ▶ 18.12
- ▶ 08.01
- ▶ 15.01
- ▶ 22.01
- ▶ 29.01

Terminfindung

Mögliche Termine:

- ▶ Donnerstags 15:00 Uhr
- ▶ Dienstags 15:00 Uhr
- ▶ Dienstags 12:30 Uhr

Struktur eines Vortrags

Ablauf:

- ▶ Beschreibe das Thema
- ▶ Warum ist das Thema interessant? Warum hast du dieses Thema gewählt? Was sind typische Anwendungen?
- ▶ Erkläre kurz nötiges Hintergrundwissen
- ▶ Im Hauptteil Thema vorstellen
- ▶ Live-Demo (wenn möglich)
- ▶ Bonus Punkte: Live-Demo von eigenem Code

Dabei auf folgendes achten:

- ▶ Nicht zu sehr ins Detail gehen (Keiner hat sich mit dem Thema so stark beschäftigt wie ihr)
- ▶ Folien übersichtlich halten (maximal halb so viel Text wie auf dieser Folie)

Struktur eines Ausarbeitung

- ▶ Ähnlich wie der Vortrag
- ▶ Alles etwas länger und ausführlicher
- ▶ \LaTeX ist Pflicht!
- ▶ <https://www.latex-tutorial.com/tutorials/>
- ▶ Template stellen wir bereit

Dabei auf folgendes achten:

- ▶ Ihr sollt nicht das Buch oder die Paper nacherzählen!

Deadlines und Ablauf

- ▶ Man kann sich ab jetzt 2 Wochen lang ohne Fehlversuch abmelden
- ▶ Ein Monat nach dem Vortrag: Abgabe der Ausarbeitung
- ▶ Nach Feedback von uns: Letzte Überarbeitung der Ausarbeitung

Die Themen

1. Einführung in Audio (1 Person)

- ▶ *Erster Einführungsvortrag!*
- ▶ Fragestellungen:
 - ▶ Was ist Audio? Was ist Klang? Wie stellt man ihn da?
 - ▶ Sheet Music, MIDI, Waves and Waveforms, Frequency and Pitch, Dynamics, Intensity, and Loudness, Timbre
- ▶ Quelle: Buch, Seite 1-30

2. Fourier Transformation (1-2 Personen)

- ▶ *Zweiter Einführungsvortrag!*
- ▶ *Achtung: sehr mathematisch und schwierig*
- ▶ Fragestellungen:
 - ▶ Was sind Zeit- und Frequenz Domäne von Audio Signalen?
 - ▶ Was ist ein Spektrogramm und welche Informationen enthält es?
 - ▶ Wie kann man mittels Fourier Transformation zwischen Zeit- und Frequenz Domäne wechseln?
 - ▶ Was ist die Diskrete Fourier Transformation?
 - ▶ Wie kann man z.B. einfache Tiefpassfilter Mittels Fourier Transformation interpretieren?
- ▶ Quelle: Buch, Seite 39-105

3. Analogue to Digital (1 Person)

- ▶ Wie kommt man von einem akustischen Signal zu einer digitalen Representation?
- ▶ Fragestellungen:
 - ▶ Was ist Pulse Code Modulation?
 - ▶ $\text{sampling depth} \cdot \text{sampling frequency} \cdot \text{channels} = \text{bitrate}$
 - ▶ Alle Terme und ihren Zusammenhang erklären
 - ▶ wie ist ein A/D Wandler aufgebaut?
- ▶ Überblick:
https://en.wikipedia.org/wiki/Digital_audio
- ▶ Quelle: Black, Edson: Pulse Code Modulation

4. Beat Detection (1-2 Personen)

- ▶ Fragestellungen:
 - ▶ Wie wird es angewand (z.B. in DJ-Software)?
 - ▶ Onset Detection (Erkennen neuer Events)
 - ▶ Tempoerkennung
 - ▶ Beat Tracking (Erster Note eines Takts folgen)
- ▶ Quelle: Buch, Seite 303-347

5. Audio Coding formats (1 Person)

- ▶ Survey mit pro/contra
- ▶ lossy vs lossless
- ▶ Eigenständige Recherche
- ▶ Fragestellungen:
 - ▶ Kann man die Codecs gruppieren?
 - ▶ Für welche Anwendungen eignet sich welcher Codec?
(z.B. Streaming vs. Download)
 - ▶ Was sind die Ideen hinter den Codecs?
- ▶ Startpunkt: https://en.wikipedia.org/wiki/Comparison_of_audio_coding_formats

6. MP3 u. AAC (1-2 Personen)

- ▶ Genaue Betrachtung von MP3 und AAC
- ▶ Fragestellungen:
 - ▶ Wie funktioniert der Algorithmus
 - ▶ Wie wird Audio File Compression realisiert?
 - ▶ Welche Trade-offs wurden gemacht?
- ▶ Quellen:
 - ▶ Brandenburg: MP3 and AAC explained
 - ▶ Church: On beer and audio coding

7. JACK (1-2 Personen)

- ▶ Fragestellungen:
 - ▶ Was ist Echtzeit Audio?
 - ▶ Was ist ein Sound Server?
 - ▶ Was sind die Anforderungen?
 - ▶ Wie wird JACK diesen Anforderungen gerecht?
 - ▶ Vergleich mit PulseAudio oder ALSA
 - ▶ Implementations Details, wie Memory, Buffer, etc
 - ▶ Jack Live Presentation
- ▶ Quellen
 - ▶ <http://jackaudio.org/files/refman.pdf>
 - ▶ <http://www.alsa-project.org/~tiwai/soundsystems.pdf>
 - ▶ <http://libremusicproduction.com/articles/demystifying-jack-%E2%80%93-93-beginners-guide-getting-started-jack>

8. Soundchip des Commodore 64 (1 Personen)

- ▶ Fragestellungen:
 - ▶ Was ist der historische Kontext?
 - ▶ Was kann der Chip?
 - ▶ Wie programmiert man den Chip?
 - ▶ Klangbeispiele
- ▶ Quellen
 - ▶ https://en.wikipedia.org/wiki/MOS_Technology_SID
 - ▶ Official SID documentation

9. Synthesizer (1-2 Personen)

- ▶ Fragestellungen:
 - ▶ Beschreibe Techniken zur Sounderzeugung (Additive synthesis, Subtractive synthesis, FM synthesis, ...)
 - ▶ Beschreibe welche Parameter den Sound beeinflussen (Attack/Decay/Sustain/Release, Low/Highpass, ...)
 - ▶ Beschreibe, wie der Spieler die Parameter beeinflussen kann (LFOs, Velocity, Ribbon Control, ...)
 - ▶ Live Demo der beschriebenen Phänomene
- ▶ Quellen:
 - ▶ http://evanxmerz.com/soundsynthjava/Sound_Synth_Java.html
 - ▶ http://exellon.net/book/The_Complete_Synthesizer.pdf
 - ▶ <http://kotisivut.fonet.fi/~jkleimol/Jari%20Kleimola%20MST.pdf>
 - ▶ <https://www2.ak.tu-berlin.de/~fhein/Alias/Studio/Miranda/Csound/Lecture2/SynthLecture2.pdf>

10. Pitch and Chord Recognition (1-2 Personen)

- ▶ Fragestellungen:
 - ▶ Anwendungen?
 - ▶ Wie erkennt man Pitch?
 - ▶ Wie erkennt man Akkorde?
 - ▶ Was sind Hidden Markov Models?
- ▶ Quellen:
 - ▶ Kuhn: A Real-Time Pitch Recognition Algorithm for Music Applications
 - ▶ Buch, Seite 237-297

11. Music Structure Analysis (1-2 Personen)

- ▶ Analyse von Songstrukturen z.B. für:
 - ▶ Segmentation (Intro, Chorus, Bridge etc.)
 - ▶ Audio Thumbnailing
- ▶ Fragestellungen:
 - ▶ Was ist Self-Similarity?
 - ▶ Was sind Fitness Measures?
 - ▶ Weitere Anwendungsgebiete?
- ▶ Quelle: Buch, Seite 167-230

12. Audio-Based Music Classification and Annotation (1-2 Personen)

- ▶ Klassisches Machine-Learning Problem
- ▶ Teilprobleme:
 - ▶ Genre Classification
 - ▶ Mood Classification
 - ▶ Artist Identification
 - ▶ Instrument Recognition
 - ▶ Music Annotation
- ▶ Startpunkt: Fu et al.: A survey of Audio-Based Music Classification and Annotation

13. Shazam (1-2 Personen)

- ▶ Wie funktioniert Shazam?
- ▶ Fragestellungen:
 - ▶ Was ist Audio Fingerprinting?
 - ▶ Wie muss eine Datenbank aufgebaut sein?
 - ▶ Was sind die größten Herausforderungen?
- ▶ Quellen:
 - ▶ <http://coding-geek.com/how-shazam-works/>
 - ▶ Buch, Seite 355-408

14. Computer Composing (1-2 Personen)

- ▶ 3 Paper: 1971, 1991, und 2003
- ▶ Fragestellungen:
 - ▶ Welche verschiedene Ansätze gibt es?
 - ▶ Was hat sich im Laufe der Zeit verändert, z.b. durch Machine Learning?
 - ▶ Und im Jahre 2017?
- ▶ Quellen:
 - ▶ Moorer: Music and Computer Composition
 - ▶ Horner: Genetic Algorithms and Computer-Assisted Music Composition
 - ▶ Dubnov et al.: Using Machine-Learning Methods for Style Modeling
 - ▶ Papadopolus, Wiggins: AI Methods for Algorithmic Composition: A survey, a Critical View and Future Prospects

15. Personal orchestra (1 Person)

- ▶ Gestenbasiertes dirigieren einer Aufnahme eines echten Orchesters
- ▶ Fragestellungen:
 - ▶ Gestenerkennung
 - ▶ Tempo anpassung
 - ▶ Echtzeit
- ▶ Quellen:
 - ▶ Borchers et. al: Personal orchestra: a real-time audio/video system for interactive conducting
 - ▶ Schertenleib et. al.; Conducting a Virtual Orchestra

16. Eigener Vorschlag?

