

ARTISTIC TIME-STRETCHING

Realtime Time-Stretching Effekte als Auditory Virtual
Environment in einer Theaterproduktion von Kafkas
'Der Prozess'

MANUEL PLANTON

Bachelorarbeit

*Institut für Elektronische Musik und Akustik
Universität für Musik und darstellende Kunst Graz*

Betreuer:
WINFRIED RITSCH Ao.Univ.Prof. DI

Graz, im Februar 2018



(Name in Blockbuchstaben)

(Matrikelnummer)

Erklärung

Hiermit bestätige ich, dass mir der *Leitfaden für schriftliche Arbeiten an der KUG* bekannt ist und ich diese Richtlinien eingehalten habe.

Graz, den

.....
Unterschrift der Verfasserin/des Verfassers

Inhaltsverzeichnis

Abstract	3
Kurzfassung	4
1 Einleitung	5
1.1 Überblick über den Aufbau der Arbeit	5
1.2 Das Projekt	6
1.3 Kafkas ‘Der Prozess’	7
1.4 AVE - Auditory Virtual Environment	9
2 Time-Stretching	11
2.1 Definition	11
2.2 Klassische Anwendungen	11
2.3 TSM Algorithmen	12
2.3.1 Grundsätzliche Vorgangsweise von TSM Algorithmen	13
2.3.2 Artefakte von TSM Algorithmen	14
2.3.3 TSM basierend auf Overlap-Add (OLA)	16
2.3.4 TSM basierend auf Waveform Similarity Overlap-Add (WSOLA)	17
2.3.5 TSM basierend auf dem Phase Vocoder (PV-TSM)	18
2.3.6 TSM basierend auf Harmonic-Percussive Separation	19
2.3.7 TSM basierend auf dem Sinusoidal plus Residual Model	20
3 Artistic Time-Stretching	24
3.1 Das Konzept	24
3.2 Artistic Time-Stretching als Auditory Virtual Environment	24
3.3 Mögliche Effekte	25
4 Softwareentwicklung	27
4.1 Anforderungen und Konkretisierung	27
4.2 Implementierung	28
4.2.1 Pd-Abstraction: Realtime TSM basierend auf OLA	29
4.2.2 Pd-External: Realtime TSM mittels librubberband	31
4.2.3 Integration in ave-absurdum	33

4.2.4	Haupt-GUI	33
5	ave-absurdum	36
5.1	Das Audiosystem	36
5.2	Aufgabenbereiche	38
5.3	Proben und Live-Performance	40
6	Fazit	44
7	Links	45
	Literaturverzeichnis	46

Abstract

The generation of an Auditory Virtual Environment after Jens Blauert, in short AVE, is the creation of the auditory part of an interactive virtual environment. In this virtual environment humans have perceptions, that do not correspond to their physical environment. Time-Stretching refers to a time prolongation of an audio signal without effecting its pitch. This can be achieved with various algorithms, which produce different artifacts. The concept “Artistic Time-Stretching” has been developed, that uses these artifacts as a musical effect to create an AVE. To use this concept in a theatre production of Kafka’s “The Trial”, the needed realtime capable software has been developed. With the help of Ambisonics, a three-dimensional surround sound technique, a live performance originated in cooperation with directors and actors, which was successfully used in the public performances.

Kurzfassung

Die Erzeugung einer Auditory Virtual Environment nach Jens Blauert, kurz AVE genannt, ist die Erschaffung des auditiven Teils einer interaktiven virtuellen Umgebung. In dieser virtuellen Umgebung haben Menschen Empfindungen, die nicht ihrer physikalischen Umgebung entsprechen. Time-Stretching bezeichnet eine zeitliche Dehnung eines Audiosignals ohne Tonhöhenänderung. Dies kann mit verschiedenen Algorithmen erfolgen, die jedoch unterschiedliche Artefakte erzeugen. Es wurde das Konzept “Artistic Time-Stretching” entwickelt, welches diese Artefakte bewusst als musikalischen Effekt nutzt, um eine AVE zu kreieren. Um dieses Konzept in einer Theaterproduktion von Kafkas “Der Prozess” praktisch einzusetzen, wurde die benötigte echtzeitfähige Software entwickelt. Mit Hilfe von Ambisonics, einer Form von dreidimensionaler Beschallung, entstand eine Live-Performance in Kooperation mit Regisseuren und Schauspielern, welche als Teil des Theaterstücks erfolgreich in den Aufführungen Verwendung fand.

1 Einleitung

In dieser Bachelorarbeit befindet sich die Dokumentation meines Bachelorprojektes und die inhaltliche Auseinandersetzung der darin vorkommenden Themen.

Am Anfang dieses Kapitels steht der *Überblick über den Aufbau der Arbeit*, in dem die Gestaltung dieses Dokuments beschrieben wird. Im Folgenden wird die Theaterproduktion vorgestellt, in deren Rahmen das Bachelorprojekt absolviert wurde. Darauf folgt eine Zusammenfassung von Kafkas Roman ‘Der Prozess’, um einen inhaltlichen Einblick in die Theaterproduktion zu geben. Weiters wird der Begriff *Auditory Virtual Environment* nach Jens Blauert definiert, der die Theorie kennzeichnet, welche die Eigenschaften des in der Theaterproduktion erzeugten Schallfeldes beschreibt.

Englische Fachausdrücke werden in der Originalsprache beibehalten, wenn es kein gängiges deutsches Äquivalent gibt. Externe Links werden blau dargestellt und sind für gedruckte Exemplare dieses Dokuments am Ende des Dokuments aufgeführt. Interne Links sind Verweise innerhalb dieses Dokuments und werden rot dargestellt.

1.1 Überblick über den Aufbau der Arbeit

Im Kapitel **Time-Stretching** werden nach der Definition des Begriffes klassische Anwendungen dieser Technik vorgestellt. Es folgt eine Erklärung der grundsätzlichen Vorgangsweise solcher Algorithmen, die Time-Stretching implementieren und der dabei entstehenden Artefakte. Danach werden die wichtigsten Algorithmen vorgestellt und nach deren Qualitätsmerkmale und Artefakte kategorisiert.

Das Kapitel **Artistic Time-Stretching** beschreibt das grundlegende Konzept, welches für das Bachelorprojekt entwickelt wurde. Es erklärt, wie Time-Stretching als AVE¹ verwendet werden kann, indem man unter anderem die Artefakte der vorgestellten Algorithmen künstlerisch nutzt.

Der Softwareentwicklungsprozess der Software, die im Theaterprojekt zur Anwendung

¹siehe **AVE - Auditory Virtual Environment**

kam, wird im Kapitel [Softwareentwicklung](#) dargestellt.

Im Kapitel [ave-absurdum](#) wird dieses Teilprojekt der Theaterproduktion vorgestellt. Beschrieben wird das verfügbare Audiosystem im Györgyi-Ligeti-Saal und die Aufgabenbereiche im Team. Es folgt eine Beschreibung der Proben mit Schauspielern und Regisseuren und der Live-Performance, die in den Aufführungen durchgeführt wurde.

1.2 Das Projekt

Die Theaterproduktion² von Kafkas Roman ‘Der Prozess’ war ein institutsübergreifendes Projekt der Kunst Universität Graz aus den Bereichen Bühnengestaltung, Schauspiel und Gesang sowie Elektronische Musik und Akustik im Herbst 2016 („Franz Kafka: Der Prozess“ [2016](#)). Diese Produktion wurde nach einer Bearbeitung der Romanvorlage mit Schauspiel, Chor, Gesangssoli, Sounddesign und elektronischen Live-Effekten aufgeführt. Die Aufführungsstandorte waren der Györgyi-Ligeti-Saal des [MUMUTH](#)³, das [TiP](#)⁴ und ein aufgebauter Pavillon - die Peepshow - auf dessen Vorplatz. Die Vorstellung fand in drei gleichen, zeitlich aufeinander folgenden, Durchgängen an den drei Schauplätzen gleichzeitig statt und endete mit dem Finale im Györgyi-Ligeti-Saal. So konnte das Publikum in drei Gruppen geteilt dem Stück folgen und nach jedem Durchgang die Aufführungsstandorte wechseln. Abschließend folgte das Finale, bei dem das gesamte Publikum anwesend war.

Von Seiten des IEM⁵ wurde das Projekt unter der Leitung von Winfried Ritsch in folgende Teilprojekte und Arbeitsverzeichnisse (Repositories) mit den zugehörigen Gruppen von Studierenden aufgeteilt:

- [ave-absurdum](#): Erzeugung einer AVE mit unterschiedlichen Effekten im Györgyi-Ligeti-Saal
- [ave-streamer](#): Versorgung der Aufführungsstandorte mit den Audio-Streams der jeweils anderen

²In dieser Arbeit werden die Begriffe Theaterproduktion und Theaterprojekt synonym verwendet.

³Haus für Musik und Musiktheater

⁴Theater im Palais

⁵Institut für Elektronische Musik und Akustik

- [ave-tetrahedron](#): Erzeugung einer AVE mittels Tetrahedron-Lautsprechern als Instrument für den Vorhof der aufgebauten Peepshow
- [ave-kafka](#): Gemeinsames Repository zur Zusammenarbeit im Projekt mit [Projektdokumentation](#)

Meine Aufgabe war es, im Team des Teilprojektes ave-absurdum im Györgyi-Ligeti-Saal, eine AVE mit dem Konzept [Artistic Time-Stretching](#) zu erzeugen. Dieses Konzept wurde im Vorfeld entwickelt und wird an einer späteren Stelle in dieser Arbeit vorgestellt.

1.3 Kafkas ‘Der Prozess’

Zusammenfassung:

Josef K. wird am Morgen seines 30. Geburtstags verhaftet. Er ist sich keiner Schuld bewusst und das anklagende Gericht und dessen Recht ist ihm vollkommen unbekannt. Selbst den Inhalt der Anklage darf er nicht wissen. Er wird jedoch nicht festgenommen und kann seinem Beruf als Prokurist einer Bank weiter nachgehen.

Bei der ersten Untersuchung hält K. eine Rede über die Sinnlosigkeit des Verfahrens und über die Korruption dieser Beamtenschaft. Er erkennt, dass alle Anwesenden Beamte sind; selbst die in zwei Parteien aufgeteilte Menschenmenge, die dem Prozess verfolgt. Der Untersuchungsrichter spricht kaum und er gibt keine Informationen über K.s Prozess, bis er schlussendlich sagt, dass K. sich den Vorteil eines Verhörs beraubt hatte.

Die Gerichtsräume sind so verwinkelt, undurchsichtig und erdrückend, wie auch das ihm unbekannte geltende Recht bzw. die Rechtsausübung dieser Organisation. Entgegen den Informationen des Gerichts kommt es zu keiner weiteren Verhandlung.

Wie auch K.s Onkel, erfahren immer mehr Menschen in seiner Umgebung von seinem Prozess. Dieser stellt ihm den Advokaten Huld vor, der ihn vor dem Gericht vertreten wird. Der Advokat ist geschwächt, da er an einer Herzkrankheit leidet. In dessen Haus besprechen der Onkel, der Advokat und der Kanzleidirektor K.s Verhandlung.

Doch er ist wenig interessiert und geht mit Leni, der Pflegerin des Advokaten, in das Arbeitszimmer, wo sie sich näher kommen. Danach verurteilt der Onkel K.s Abwesenheit.

K. wird immer mehr von seinem Prozess vereinnahmt, sodass sein Zustand seine Arbeit beeinflusst. Das Unwissen über die Anklage und die Vorgänge im Gericht setzen ihm zu. Er beschließt, die Verteidigung doch selbst in die Hand zu nehmen und die erste Einlage für das Gericht zu verfassen, die der Advokat immer wieder hinausgezögert hatte. In ihr müsste er alle Entscheidungen seines Lebens rechtfertigen, da er die Anklage nicht kennt.

Ein Gerichtsmaler bietet K. an, ihm mit seinen Beziehungen zu den eiteln Richtern zu helfen. Von ihm erfährt er, dass ein wirklicher Freispruch vor diesem Gericht noch nie vorgekommen ist. Er rät ihm, dass eine ständige Verzögerung des Urteils die beste Wahl wäre.

K. kommt zum Haus des Advokaten und trifft dort Kaufmann Block, der auch bei einem Prozess vom Advokaten vertreten wird. Der Kaufmann beschäftigt auch mehrere Winkeladvokaten und verwendet seine gesamten Ressourcen für seinen Prozess, der ihn schon total vereinnahmt hat. Beim Advokaten angelangt, sagt K., dass er ihn nicht mehr weiter beschäftigen wird. Der Advokat will K. umstimmen und lässt den Kaufmann herein. Er wird vom Advokaten beleidigend und sklavenartig, wie ein Hund, behandelt, während er ihm von den Neuigkeiten seines Prozesses berichtet. Der Kaufmann scheint vollkommen abhängig von allen zu sein, die mit seinem Prozess zu tun haben. Umgekehrt wird er von diesen immer schlechter behandelt. K. lässt sich dadurch erst recht nicht umstimmen.

Gestresst und überarbeitet von den zusätzlichen Mühen, die ihm der Prozess bereitet, fällt es K. sehr schwer, seiner Arbeit nachzugehen. Er soll einem italienischen Geschäftsfreund der Bank die Sehenswürdigkeiten der Stadt zeigen. Nach einer Besprechung mit ihm und dem Bankdirektor, vereinbaren sie ein Treffen um 10 Uhr beim Dom. K. trifft beim Dom ein, doch der Italiener erscheint nicht. Ein Geistlicher erklimmt eine Kanzel und spricht zu ihm. Er sei der Gefängniskaplan und habe ihn schicken lassen. Sie unterhalten sich über den Prozess, dass er wohl schlecht ausgehen

werde und reden über die Bedeutung eines Gleichnisses eines Türhüters, das aus dem Gesetz des anklagenden Gerichts stammt. Schließlich erklärt der Geistliche, dass er zum Gericht gehört und nichts von K. will.

Am Vorabend von K.s 31. Geburtstag wird er von zwei Männern aus der Stadt abgeführt und bei einem Steinbruch durch einen Messerstich ins Herz hingerichtet. (Kafka 2006)

Die Szenen und die Atmosphäre im Roman werden immer absurder und erdrückender. K. ist einem anonymen Gerichtsapparat ausgeliefert, der scheinbar willkürlich anklagt und alles von den Angeklagten abverlangt. Die Beamten des Gerichts sind korrupt, aber auch extrem penibel in ihrer Pflichtausübung. Je mehr K. versucht, den Prozess zu seinem Vorteil zu beeinflussen, desto schlechter steht es um ihn. Schließlich wird er ohne Urteil hingerichtet.

1.4 AVE - Auditory Virtual Environment

“Eine virtuelle Umgebung⁶ zielt darauf ab, Situationen zu erzeugen, in denen Menschen Empfindungen haben, die nicht ihrer physikalischen Umgebung, sondern einer virtuellen Umgebung entsprechen”⁷ (Blauert 2005, 277). Der auditive Teil einer solchen Umgebung wird nach Blauert (2005) Auditory Virtual Environment (AVE) genannt. Die AVE wird in einem ComputermodeLL geformt und erzeugt die entsprechenden Signale zur klanglichen Darbietung. Dieser Begriff ist jedoch von einer einfachen Simulation abzugrenzen. Der entscheidende Unterschied zwischen einer einfachen Simulation und einer AVE ist, dass das System zur Erzeugung der AVE interaktiv gestaltet ist. (Blauert 2005)

Dies kann in den folgenden zwei Beispielen erläutert werden.

Im Falle einer Darbietung mit Kopfhörern kann mit einem Halleffekt eine binaurale Raumsimulation erzeugt werden. Bewegt die beschallte Person nun ihren Kopf, dreht sich das simulierte Schallfeld mit der Kopfbewegung mit. Wird jedoch z.B.

⁶virtual environment

⁷A virtual environment aims at creating situations in which humans have perceptions that do not correspond to their physical environment but to a virtual one.

mittels Headtracking und mit der Verwendung von HRTF⁸s das virtuelle Schallfeld von der physikalischen Umgebung entkoppelt, so kann die Kopfdrehung interaktiv ausgeglichen werden und der Raumeindruck entspricht einer virtuellen Umgebung, was als AVE bezeichnet wird.

Bei der Darbietung mit Lautsprechern kann mittels Stereophonie eine Simulation geschaffen werden, in der Schallquellen auf einer horizontalen Ebene positioniert werden. Einerseits sind hier die möglichen Positionen für virtuelle Schallquellen zur Erzeugung einer virtuellen räumlichen Umgebung stark innerhalb der Horizontalebene begrenzt und andererseits kann sich die beschallte Person nur innerhalb eines kleinen Bereichs⁹ bewegen, da sonst das Stereobild verzerrt wird. Die Erzeugung einer AVE mit Lautsprechern ist z.B. mittels Ambisonics¹⁰ möglich. Ist die Ordnung des Systems hoch genug und der verwendete Raum schallarm, so lässt sich eine AVE schaffen, in der sich die Zuhörerinnen und Zuhörer relativ frei bewegen können, ohne den Eindruck des virtuellen Schallfeldes zu verlieren.

Blauert unterteilt AVEs in 3 Kategorien:

1. *Authentic*: Reproduktion von real existierenden Umgebungen
2. *Plausible*: Erzeugung einer Umgebung, die alle relevanten Schallereignisse für eine bestimmte Anwendung aufweist, als ob sie in einer realen Umgebung stattgefunden hätten
3. *Creational*: Erzeugung einer Umgebung, die keine Ansprüche an Authentizität oder Plausibilität stellt

(Blauert 2005, 278)

⁸Head Related Transfer Function

⁹“sweet-spot” bei Verwendung von zwei Lautsprechern

¹⁰siehe Malham und Myatt (1995)

2 Time-Stretching

Die entwickelte Software für das Bachelorprojekt basiert auf Time-Stretching. Daher wird in diesem Kapitel eine begriffliche Abgrenzung definiert, gefolgt von klassischen Anwendungen und schließlich eine Erklärung der wichtigsten Algorithmen. Die einzelnen Algorithmen werden bezüglich ihrer *Qualitätsmerkmale* und *Artefakte* bewertet, um ihre Eignung für das Konzept **Artistic Time-Stretching** zu erkennen.

2.1 Definition

Time-Scale Modification (TSM) ist der Vorgang, ein Audiosignal zeitlich zu verlängern oder zu verkürzen, ohne dessen Tonhöhe zu ändern (Driedger und Müller 2016, 1). Time-Stretching bedeutet eine zeitliche Dehnung des Audiosignals ohne Tonhöhenänderung, wird jedoch meist synonym zu TSM verwendet (Zölzer und Amatriain 2002, 205), obwohl TSM der allgemeinere Begriff ist. Time-Compression hingegen bezeichnet die zeitliche Verkürzung (zeitliche Kompression¹¹) eines Audiosignals ohne Tonhöhenänderung.

2.2 Klassische Anwendungen

Die klassischen Anwendungen von TSM reichen von der Integration¹² in den meisten DAWs¹³ zur Audibearbeitung über atomatisches Anpassen des Song-Tempos aufeinanderfolgender Musikstücke¹⁴ bis zur Synchronisation der Audiospur des Videomaterials bei der Erstellung von Slow Motion Videos¹⁵.

Weiters kann Pitch Shifting, also eine Tonhöhenänderung ohne Änderung der Signaldauer, eines Audiosignals mittels TSM und Resampling realisiert werden.¹⁶ Hiermit kann beispielsweise eine Tonhöhenkorrektur einer Audiospur erfolgen. Dafür

¹¹nicht zu verwechseln mit der dynamischen Kompression eines Signals

¹²siehe Cockos Inc. (2017), 183; („The Ardour Manual“ 2017)

¹³Digital Audio Workstation

¹⁴siehe Cliff (2000)

¹⁵siehe Moinet, Dutoit, und Latour (2013)

¹⁶siehe Zölzer und Amatriain (2002), 217-219

gibt es kommerzielle¹⁷ und auch quelloffene¹⁸ Produkte in Form von Audio-Plugins¹⁹, Audio-Editoren²⁰ oder wie bei TSM in die DAW integriert²¹. Umgekehrt kann TSM mittels Pitch-Shifting und Resampling realisiert werden.

Diese klassischen Anwendungen bedingen eine möglichst hohe Audioqualität der TSM Algorithmen. Also sollen durch den Algorithmus möglichst wenige Artefakte entstehen. Ein wesentlicher Teil dieses Projekts war es jedoch, diese Artefakte als künstlerischen Effekt zu nutzen. Somit werden im restlichen Kapitel entstehende Artefakte von TSM Algorithmen nicht als prinzipiell schlecht angesehen, wie dies meist im technischen Sinne geschieht, sondern als künstlerisch gleichwertig zu Qualitätsmerkmalen der Algorithmen.

2.3 TSM Algorithmen

Musiksignale sind komplexe Mischungen aus verschiedensten Quellen mit sehr unterschiedlichen Charakteristika. Es kommen harmonische Klänge, perkussive transiente Geräusche, Mischformen daraus und noch viele andere Signalarten vor. Dies stellt einen hohen Anspruch an einzelne TSM Routinen, die ein qualitativ hochwertiges Ergebnis erzielen sollen, da diese meist nur für eine Signalart gut funktionieren. Bei anderen Signalen entstehen zum Teil starke Artefakte. Weil viele verschiedene Signalformen möglich sind und mehrere Algorithmen zur Auswahl stehen, kann durch eine Trennung des Quellsignals nach Signaltypen und einer Kombination von mehreren TSM Algorithmen, die diese Signaltypen mit guter Qualität verarbeiten können, eine gute Gesamtqualität erreicht werden. Die Vorteile der jeweiligen Algorithmen können so besser genutzt werden, als bei der Verwendung von nur einem Algorithmus. (Driedger und Müller 2016, 1–2)

Die TSM Routine kann jedoch auch so gewählt werden, dass starke Artefakte für einen künstlerischen Einsatz entstehen. Es hat sich im Verlauf des Projektes herausgestellt, dass das Erzielen eines interessanten Effekts gut gelingt, wenn ein

¹⁷siehe („Antares Auto-Tune“ 2017)

¹⁸siehe („Zeta-AT1 - Quick Guide“ 2017); Baran (2017)

¹⁹siehe („LV2“ 2017); („Virtual Studio Technology“ 2016)

²⁰siehe („Sliding Time Scale / Pitch Shift - Audacity Manual“ 2017)

²¹siehe Cockos Inc. (2017), 179

bestimmter Algorithmus implementiert wird und dessen Parametereinstellungen mittels Experimentieren intuitiv ermittelt werden. Die Auswahl des Algorithmus erfolgt bei dieser Herangehensweise über die Artefakte, die von ihm produziert werden. Anschließend können moderate aber auch oft extreme Parametereinstellungen verwendet werden, um das künstlerische Potential des Algorithmus zu erkunden. Anschließend legt man sich für bestimmte Parameterintervalle fest und versieht diese Intervalle mit den gewünschten Skalierungen, um diese schließlich mit dem vorhandenen Interface komfortabel bedienen zu können.

2.3.1 Grundsätzliche Vorgangsweise von TSM Algorithmen

Viele TSM Algorithmen nutzen die folgende grundlegende Strategie, um eine zeitliche Änderung eines Audiosignals zu ermöglichen, ohne dessen Tonhöhe zu beeinflussen.

1. Analyse: Das Eingangssignal wird in Analyse-Frames mit fixer Länge im Millisekunden-Bereich zerlegt. Jeder Frame erfasst den lokalen Tonhöhengehalt des Gesamtsignals. Die Analyse-Frames können sich überlappen, oder auch Lücken zwischeneinander haben.
2. Relokation: Die Frames werden mit einem meist konstanten Zeitintervall relativ zueinander verschoben. Hier geschieht die zeitliche Änderung des Signals.
3. Synthese: Das modifizierte TSM Signal wird durch Zusammenführen der verschobenen Frames gebildet. In diesem Schritt handelt es sich um Synthese-Frames, die allgemein in irgendeiner Art und Weise aus den verschobenen Analyse-Frames gebildet werden.

Das Verhältnis zwischen dem fixen zeitlichen Abstand der Synthese-Frames H_s und dem variablen Abstand der Analyse-Frames H_a ergibt den Time-Stretching-Faktor α .

$$\alpha = \frac{H_s}{H_a}$$

Time-Stretching liegt vor, wenn das Zeitintervall zwischen den Analyse-Frames H_a kleiner als das Zeitintervall zwischen den Synthese-Frames H_s ist. Daraus folgt ein

$\alpha > 1$.

Umgekehrt entsteht Time-Compression bei einem $\alpha < 1$.

Bei $\alpha = 1$ wird ein Ausgangssignal in Originalgeschwindigkeit produziert.

(Driedger und Müller 2016, 2–3)

2.3.2 Artefakte von TSM Algorithmen

Artefakte bei Algorithmen im Zeitbereich:

- Phase jumping: Wird die lokale Periodizität des Originalsignals bzw. die Periodizität der benachbarten Synthese- oder Analyse-Frames nicht berücksichtigt, so entstehen Phasensprünge im Synthesesignal. Diese Phasensprünge klingen ähnlich wie eine Art Frequenzmodulation. Dieser Artefakt macht sich vor allem bei harmonischen Klängen bemerkbar, da die Wahrnehmung eines Klanges (Klangfarbe, Tonhöhe) maßgeblich von dessen lokaler periodischer Struktur abhängt. (Driedger und Müller 2016, 5)
- Transient doubling: Da sich die Analyse-Frames bei Time-Stretching überlappen, können kurze transiente Signalanteile des Originalsignals in zwei oder mehreren Analyse-Frames enthalten sein. Nach der Relokation und der Synthese ist das transiente Signal mehrmals im Synthesesignal enthalten. Dieser Artefakt entsteht häufig bei langer Frame-Länge und hohem Time-Stretching-Faktor.
- Transient skipping: Da die Analyse-Frames bei Time-Compression Lücken zueinander haben, kann es dazu kommen, dass kurze transiente Signalanteile in diesen Lücken liegen und so nicht in den Analyse-Frames enthalten sind. Somit fehlen diese transienten Signalanteile im Synthese-Signal. Dieser Artefakt entsteht häufig bei kurzer Frame-Länge und niedrigem Time-Stretching-Faktor.

(Driedger und Müller 2016, 9)

Artefakte bei Algorithmen im Frequenzbereich:

In “A Review of Time-Scale Modification of Music Signals” (Driedger und Müller 2016, 10–15) wird die Entstehung von Artefakten bei TSM-Algorithmen im Frequenzbereich folgendermaßen beschrieben:

TSM-Algorithmen im Frequenzbereich benutzen eine Frequenzbereichsdarstellung des Analyse-Signals, die mit einer geeigneten Transformation²² vom Zeitbereich in den Frequenzbereich gebildet wird. Damit sind die Frequenzkomponenten des Analyse-Signals zur Verarbeitung zugänglich. Die Frequenzkomponenten können als Amplitude und Phase je Frequenz interpretiert werden. Werden Veränderungen im Frequenzbereich durchgeführt, ergeben sich diese zwei charakteristischen Eigenschaften in Bezug auf entstehende Artefakte:

1. Horizontale Phasenkohärenz: kontinuierliche Phase aller Frequenzkomponenten eines Signals entlang der Zeitachse
2. Vertikale Phasenkohärenz: fixes Verhältnis der Phasen aller Frequenzkomponenten eines Signals innerhalb eines Frames

Die Verbesserung der horizontalen Phasenkohärenz wird meist zur Vermeidung von Phasensprüngen (phase jumping) auf Kosten der vertikalen Phasenkohärenz erlangt. Der Verlust der vertikalen Phasenkohärenz führt zu den folgenden Artefakten:

- Transient smearing: Der Verlust der vertikalen Phasenkohärenz verschlechtert die zeitliche Lokalisation von Klangereignissen. Dies führt zur zeitlichen Verschmierung von transienten Signalanteilen. Perkussive Klänge werden gedehnt.
- Phasiness: Es entsteht eine deutliche Klangfärbung, die als Halligkeit oder Verlust des Raumklangs beschrieben werden kann. (Laroche und Dolson 1997, 1)

In diesem Abschnitt werden nun verschiedene TSM Algorithmen vorgestellt. Die Aufzählung dient als Überblick über die Qualitätsmerkmale und die charakteristischen Artefakte dieser Algorithmen.

²²siehe short-time Fourier transform bzw. TSM basierend auf dem Phase Vocoder (PV-TSM)

2.3.3 TSM basierend auf Overlap-Add (OLA)

Dies ist der einfachste TSM Algorithmus und ist der grundsätzlichen Vorgangsweise von TSM Algorithmen²³ sehr ähnlich. Die Analyse-Frames bleiben im Zeitbereich und die Synthese-Frames werden durch zeitliche Fensterung der Analyse-Frames mit einer stetigen Fensterfunktion (z.B. Hann Window) gebildet. Die fixen Abstände der Synthese-Frames werden überlappend gewählt, sodass in Kombination von Fensterung, Überlappung und anschließender Addition der Synthese-Frames ein Ausgangssignal entsteht, das keine zusätzlichen Unstetigkeiten aufweist. Außerdem entstehen so keine Amplitudenschwankungen, wie es ohne Fensterung der Fall wäre.

Qualitätsmerkmale:

- Sehr recheneffizient
- Gute Erhaltung des Klanges transienter Signalanteile (perkussive Klänge) bei sehr kurzer Frame-Länge.
- Schlecht für die Erhaltung des Klanges harmonischer Klänge wegen phase jumping. Nicht geeignet für die qualitative Modifikation von harmonischen Klängen.

Artefakte:

- Phase jumping
- Transient doubling/skipping bei entsprechenden Frame-Längen

(Driedger und Müller 2016, 3–6)

Dieser Algorithmus verursacht die stärksten phase jumping Artefakte von allen Algorithmen, da die Positionen von Analyse-Frames und Synthese-Frames fix eingestellt werden, ohne auf die Erhaltung der lokalen Periodizität zwischen Frames zu achten.

²³siehe Grundsätzliche Vorgangsweise von TSM Algorithmen

2.3.4 TSM basierend auf Waveform Similarity Overlap-Add (WSOLA)

Dieser Algorithmus arbeitet ebenfalls im Zeitbereich und steht hier zur Erklärung und Einordnung auch stellvertretend für ähnliche Algorithmen wie synchronized OLA (SOLA), time-domain pitch-synchronized OLA (TD-PSOLA), oder autokorrelationsbasierte Ansätze.

WSOLA definiert kleine Toleranzen für die Positionen der Analyse-Frames. Innerhalb dieser Toleranz wird die Position des jeweils nächsten Analyse-Frames so gewählt, dass sich die periodischen Strukturen der zwei Frames möglichst überlappen. Die Signale benachbarter Frames sollen also stetig ineinander übergehen, was zu reduziertem phase jumping und so zu besserer Audioqualität führt. Für diese Anpassung wird als Ähnlichkeitsmaß die Kreuzkorrelation benachbarter Frames verwendet.

Wurde die beste Position des aktuellen Analyse-Frames mittels Kreuzkorrelation mit dem letzten Analyse-Frame innerhalb der zeitlichen Toleranz gefunden, arbeitet dieser Algorithmus sonst gleich wie OLA.

Qualitätsmerkmale:

- Gut für monophone Klänge
- Schlecht für Signale mit mehreren Quellen, da nur die prominenteste Signalquelle die Position des nächsten Frames bestimmt. Andere Quellen können so immer noch phase jumping erzeugen.

Artefakte:

- Reduziertes phase jumping
- Transient doubling/skipping

(Driedger und Müller 2016, 6–10)

2.3.5 TSM basierend auf dem Phase Vocoder (PV-TSM)

Das Ziel von TSM-Algorithmen, die im Frequenzbereich arbeiten, ist unter anderem die Vermeidung von phase jumping für alle Signalkomponenten (Driedger und Müller 2016, 10).

Der Algorithmus für einen Audioeffekt basierend auf Manipulationen im Frequenzbereich kann in diese Schritte unterteilt werden:

1. Analyse des Zeitsignals: STFT
2. Manipulation im Zeit-Frequenzbereich
3. Synthese des modifizierten Zeitsignals: ISTFT

“Der phase vocoder ist ein Vorgang um die grobe Frequenzschätzung der STFT mit der gegebenen Phaseninformation zu verfeinern.” (Driedger und Müller 2016, 11).²⁴

Diese Verfeinerung liefert die Momentanfrequenzen des Signals und der Vorgang von Transformation, Manipulation und inverser Transformation wird phase vocoder genannt. (Zölzer und Amatriain 2002, 237–38)

Der PV-TSM Algorithmus besteht also aus diesen Schritten:

1. Analyse: STFT und phase vocoder Vorgang liefern die Momentanfrequenzen.
2. Manipulation:
 - phase propagation: Anpassung der Phasen aller Spektrallinien des aktuellen Frames an die Phasen der Spektrallinien des vorherigen Frames.
 - Relokation²⁵
3. Synthese des modifizierten Zeitsignals: ISTFT

Mit der phase propagation wird die horizontale Phasenkohärenz auf Kosten der vertikalen Phasenkohärenz erlangt, was zu extrem vermindertem phase jumping für

²⁴The phase vocoder is a technique that refines the STFT’s coarse frequency estimate by exploiting the given phase information.

²⁵siehe Grundsätzliche Vorgangsweise von TSM Algorithmen

alle Quellen im Signal führt. Der Verlust der vertikalen Phasenkohärenz resultiert jedoch in phasiness.

Qualitätsmerkmale:

- sehr gut für monophone, als auch für polyphone harmonische Klänge

Artefakte:

- Transient smearing
- Phasiness

(Driedger und Müller 2016, 10–15)

2.3.6 TSM basierend auf Harmonic-Percussive Separation

In diesem Ansatz werden die Vorteile von zwei verschiedenen Algorithmen genutzt, indem das Eingangssignal mittels harmonic-percussive separation (HPS) in ein Signal mit dem harmonischen Teil und in ein Signal mit dem perkussiven Teil aufgeteilt wird. Das harmonische Teilsignal wird mittels PV-TSM und das perkussive Teilsignal wird mittels OLA mit sehr kurzer Frame-Länge verarbeitet. Somit entsteht kein transient smearing im PV-TSM Zweig und kein phase jumping im OLA Zweig, was zu einer insgesamt besseren Audioqualität führt. Schließlich werden beide Teilsignale nach der jeweiligen Verarbeitung zum Ausgangssignal superponiert. Die erreichbare Audioqualität hängt maßgeblich von der Qualität des verwendeten HPS-Filters ab. Eine einfache und effiziente Implementierung von HPS wurde von Derry Fitzgerald im Paper “Harmonic/Percussive Separation Using Median Filtering”²⁶ vorgestellt.

²⁶siehe Fitzgerald (2010)

Qualitätsmerkmale:

- sehr gut für Mischformen aus harmonischen und perkussiven Klängen

Artefakte:

- Phasiness

(Driedger und Müller 2016, 10–18)

2.3.7 TSM basierend auf dem Sinusoidal plus Residual Model

In “DAFX digital audio effects” (Zölzer und Amatriain 2002, 373–78) wird das sinusoidal plus residual model wie folgt beschrieben.

Das Sinusoidal plus Residual Model:

Den meisten Algorithmen, die im Spektralbereich arbeiten, liegt ein Analyse/Synthese-System zugrunde, welches das analysierte Eingangssignal ohne hörbaren Qualitätsverlust resynthetisieren kann (z.B. Time-Frequency Processing, Source-Filter Processing, Spectral Processing). Die Möglichkeiten und Parameter zur Beeinflussung des Signals werden hauptsächlich durch dessen Repräsentation innerhalb dieses Systems bestimmt. Um musikalische Merkmale des Signals auf einem höheren Level beeinflussen zu können, ist ein weiterer Schritt zur Analyse des Spektrums notwendig, welcher feature extraction genannt wird.

Das sinusoidal plus residual model ist ein Algorithmus, der auf einer höheren Abstraktionsebene Merkmale (features) wie

- Obertöne
- Tonhöhe
- Spektrale Einhüllende
- Vibrato
- Notenlängen

identifizieren kann. Dieser zusätzliche Analyseschritt ermöglicht einen Eingriff in das Signal, der dem menschlichen Hörempfinden näher ist, als bei Algorithmen ohne

feature extraction.

Im sinusoidal plus residual model wird wie beim phase vocoder die STFT zur Frequenzanalyse verwendet. Das Eingangssignal $s(t)$ wird dann mittels

$$s(t) = \sum_{r=1}^R A_r(t) \cos(\theta_r(t)) + e(t)$$

aus dessen Spektrum zum Zeitpunkt t modelliert.

$s(t)$... Eingangssignal

$A_r(t) \cos(\theta_r(t))$... zeitlich veränderlicher Sinus; der r te Teilton

$A_r(t)$... Amplitude des r ten Teiltens

$\theta_r(t)$... Phase des r ten Teiltens

$e(t)$... residual; Rauschkomponente des Eingangssignals

Der tonale Teil des Signals wird mit zeitlich veränderlichen Teiltönen $\sum_{r=1}^R A_r(t) \cos(\theta_r(t))$ modelliert. Von den Teiltönen wird angenommen, dass sie sich nur langsam mit Amplitude und Frequenz ändern, was in der Analyse der Spektren berücksichtigt wird. Der Rauschanteil des Signals wird mit $e(t)$ modelliert, wobei diese Größe entweder die Differenz des Originalsignals und des harmonischen Anteils darstellt oder ein rein stochastisches Signal enthält (siehe sinusoidal plus noise model), welches mit gefiltertem Rauschen beschrieben wird.

Mit diesem Modell können die musikalischen Merkmale des Signals analysiert und beeinflusst werden. Das bearbeitete Signal wird mittels additiver Synthese für den harmonischen Teil und mittels subtraktiver Synthese für die Rauschkomponente in eine Form gebracht, die mittels ISTFT transformiert werden kann und so schließlich als Ausgangssignal im Zeitbereich vorliegt. (Zölzer und Amatriain 2002, 403)

TSM mit dem Sinusoidal plus Residual Model:

Ähnlich wie PV-TSM besteht der TSM-Algorithmus mit dem sinusoidal plus residual model grundsätzlich aus den folgenden Schritten.

1. Analyse des Zeitsignals mit STFT: Bildung des Analyse-Frames im Frequenzbereich
2. Manipulation im Zeit-Frequenzbereich: *time scaling* und Bildung des Synthese-Frames
3. Synthese des modifizierten Zeitsignals mit ISTFT: Transformation des Synthese-Frames vom Frequenzbereich in den Zeitbereich

Anders als bei den meisten anderen TSM-Algorithmen, bleibt die Frame-Rate bei der Analyse und Synthese gleich. Somit werden bei time compression manche Analyse-Frames zweimal benutzt und bei time stretching manche Analyse-Frames ausgelassen, was je nach Time-Stretching-Faktor und Frame-Länge zu transient doubling bzw. skipping führen kann.

Die Manipulation im Zeit-Frequenzbereich geschieht im time scaling Modul, welches von Zölzer und Amatriain (2002) auf S.431 wie folgt beschrieben wird:

“The analysis frames AF_n , containing the spectrum amplitude and phase envelopes, are fed to the time scaling module. This module performs a peak detection²⁷ and a peak continuation²⁸ algorithm ... on the current and previous z^{-1} amplitude envelopes. Then, only the peaks that belong to a sinusoidal track are used as inputs to the spectrum phase generation module. Note that the time scaling module only changes the phase, leaving the spectral amplitude envelope as it is. The phase of each peak is computed supposing that the frequency varies linearly between two consecutive frames and that there is some phase deviation $\Delta\varphi$ The usage of the same frame rate in analysis and synthesis allows us to suppose that the phase variation between two consecutive frames is also the same.”

²⁷Der peak detection Algorithmus sucht nach den Teiltönen im Signal. Um die Teiltöne von anderen Frequenzkomponenten des Signals zu unterscheiden, werden so viele Teiltöne wie möglich gefunden und an den peak continuation Algorithmus weitergegeben. (Zölzer und Amatriain 2002, 383)

²⁸Der peak continuation Algorithmus erhält die geschätzten Amplituden, Frequenzen und Phasen der peaks innerhalb eines Frames. Diese Informationen werden nach dem sinusoidal model geordnet, nach bestimmten Kriterien aussortiert und als Sammlung von Frequenztrajektorien den validen Teiltönen zugeordnet. (Zölzer und Amatriain 2002, 390–95)

Schließlich erfolgt die Bildung und Transformation der Synthese-Frames vom Frequenzbereich in den Zeitbereich. Mit weiteren Maßnahmen können phasiness und transient smearing minimiert werden.

Qualitätsmerkmale:

- sehr gute Qualität für Musikschnale verschiedenster Art
- Mehrkanalanwendung möglich, da Amplituden- und Phasenverhältnisse beibehalten werden

Artefakte:

- transient doubling/skipping möglich

(Zölzer und Amatriain 2002, 429–35)

3 Artistic Time-Stretching

Dieses Kapitel beschreibt das grundlegende Konzept, welches für das Bachelorprojekt entwickelt wurde. Zuerst werden die theoretischen Überlegungen erklärt. Anschließend wird die Verwendung des Konzeptes als AVE²⁹ beschrieben, so wie es auch im Theaterprojekt praktisch umgesetzt wurde. Schließlich werden mögliche Effekte vorgestellt, die mit diesem Konzept erzeugt werden können.

3.1 Das Konzept

Artistic Time-Stretching nutzt bewusst Artefakte von **TSM Algorithmen** als künstlerischen Effekt. TSM wird jedoch auch selbst als Effekt zur Zeitmanipulation verwendet, der in unterschiedlichen Kontexten unterschiedliche Wirkungen hat. Zusätzlich kann auch von Pitch-Shifting Gebrauch gemacht werden, da dies mit der Implementierung von TSM einfach zu realisieren ist und eine weitere, künstlerisch wirksame, Komponente ins Spiel bringt.

3.2 Artistic Time-Stretching als Auditory Virtual Environment

Der Einsatz als AVE bedingt eine Echtzeitimplementierung der Algorithmen. Als Live-Effekt ist Time-Compression nicht möglich, bevor Time-Stretching erfolgte. Erst dann befindet sich der Abspielzeitpunkt des Effektes in der Vergangenheit, was eine beschleunigte Wiedergabe des gespeicherten Signals erst möglich macht.

Dieses Konzept ist vor allem in der Lage, eine creational AVE zu erzeugen, was im Rahmen des Theaterprojektes auch gewünscht wurde. Es kann natürlich ausschließlich als einfaches Delay verwendet werden, um eine andere Art von AVE zu erzeugen. Dies würde jedoch den Entwicklungs- und Rechenaufwand für Time-Stretching nicht rechtfertigen.

²⁹siehe **AVE - Auditory Virtual Environment**

3.3 Mögliche Effekte

TSM als Echtzeiteffekt, der ein Live-Signal verändert wiedergibt, verursacht einen Delay-Charakter, der je nach Delay-Time unterschiedlich wirkt. Wird das Originalsignal mit einem zeitverzögerten Signal überlagert, so ergeben sich geordnet nach steigender Delay-Time diese Effekte:

- Klangfärbung durch Bildung eines FIR-Kammfilters
- Slapback
- Echo (Echoschwelle bei ca. 50ms)

Wird die Delay-Time im Sekundenbereich weiter erhöht, so wird das verzögerte Signal bald nicht mehr als Echo empfunden, sondern als vom Originalsignal getrenntes Schallereignis.

Da es sich hier jedoch um eine veränderte Version des Originalsignals handelt und man sich in der Praxis meistens im Echo-Bereich mit Time-Stretching befindet, kann Artistic Time-Stretching als unnatürliches Echo aufgefasst werden. In der Natur kann eine zeitliche Dehnung eines Signals nicht vorkommen.

Die Artefakte in Kombination mit Time-Stretching können je nach Einsatz folgende ästhetische Effekte bei Anwendung auf Sprachsignale erzeugen³⁰:

- Verfremdung des Gesagten durch tonal verzerrte Wiederholung
- Bei Verwendung von leichtem Time-Stretching mit der gleichen Lautstärke wie das Originalsignal driften Gegenwart und Vergangenheit langsam auseinander.
- Wiederholung einer Aussage als Frage, indem die Tonhöhe des Satzendes erhöht wird
- Durch öfteres Wiederholen von Satzteilen in veränderter Form, kann eine Parodie bzw. eine spöttische Imitation des Gesagten erreicht werden.
- Bei Verwendung von Artefakten mit hoher Rauigkeit kann einer Aussage ein

³⁰Die Beschränkung auf Sprachsignale ergibt sich aus der praktischen Anwendung im Theaterprojekt. Dort wurden fast ausschließlich Sprachsignale von Schauspielern als Eingangssignal verwendet

agressiver Nachdruck verliehen werden.

- Wird Time-Stretching immer wieder leise angewendet, so kann eine schwebende oder verwirrende Atmosphäre (AVE) geschaffen werden.
- Kurze Aktionen wie ein Aufschrei oder Seufzen können über längere Zeit gedehnt und intensiviert werden, was den Charakter der ganzen Szene beeinflusst. Wird ein Schrei mit starken Artefakten angereichert, kann es der Aktion einen viel stärkeren Ausdruck geben.

In Kombination mit Hall und anderen Effekten wurde so eine creational AVE geschaffen, die den absurden Charakter der Theaterproduktion unterstreichen konnte.

4 Softwareentwicklung

In diesem Kapitel wird der Softwareentwicklungsprozess für die Software dargestellt, die im Theaterprojekt als Realtime Audioeffekte Verwendung fand.³¹ [Artistic Time-Stretching](#) lieferte das Konzept und die Theorie, und [Time-Stretching](#) lieferte die Algorithmen zum Softwaredesign.

Es wird beschrieben, welche Anforderungen an die Software gestellt wurden und welche konkrete Auswahl getroffen wurde. Anschließend wird die implementierte Software vorgestellt, wobei hauptsächlich die GUI³² erklärt wird, mit der die entwickelten Audioeffekte bedient wurden.³³

Es wurden zwei unterschiedliche TSM-Effekte im Repository [artistic_time_stretching](#) erstellt, welche schließlich in das Repository [ave-absurdum](#) integriert wurden, um in der Live-Performance im Theaterprojekt Anwendung zu finden. Im Laufe des Theaterprojektes wurden noch weitere Softwarekomponenten entwickelt, um die gewünschte Funktionalität für das Theaterstück zu gewährleisten.

4.1 Anforderungen und Konkretisierung

Für die Realisierung des Konzeptes von Artistic Time-Stretching als AVE wurde festgelegt, dass zwei unterschiedliche TSM-Algorithmen verwendet werden sollen.

1. ein Algorithmus mit möglichst vielen und prägnanten Artefakten, um diese künstlerisch einsetzen zu können
2. ein Algorithmus mit möglichst hoher Qualität, um auch starkes Time-Stretching ohne Artefakte zu ermöglichen

Die Limitierung auf nur zwei verschiedene Algorithmen hatte den Grund, dass die Bedienung im Live-Einsatz von mehr als zwei parallelen Effekten nicht sinnvoll

³¹siehe [Proben und Live-Performance](#)

³²graphical user interface

³³Weiterführende Links zu den Repositories befinden sich am Anfang des Unterkapitels [Implementierung](#).

erschien. Trotzdem haben zwei TSM-Effekte genug Potential, um im gesamten Stück eingesetzt zu werden.

Die gewünschte Software sollte auf einem Debian-System in einem gemeinsamen Pd-Haup-Patch verwendet werden.

Nach einer Softwarerecherche wurde gewählt, dass der erste Algorithmus TSM basierend auf OLA sein soll, weil dieser sehr starke charakteristische phase jumping und transient doubling/skipping Artefakte erzeugen kann. Die Artefakte von Algorithmen, die im Frequenzbereich arbeiten (transient smearing, phasiness) wären auch eine gute Wahl gewesen, wurden jedoch nicht verwendet, da OLA eine sehr starke Rauigkeit und extremes transient doubling erzeugen kann, was als interessanterer Effekt für dieses Projekt erschien. Der Algorithmus sollte direkt in Pd umgesetzt werden.

Der zweite Algorithmus sollte als Pd-External in C mit einer freien TSM/Pitch-Shifting Bibliothek entwickelt werden, da so der Entwicklungsaufwand gering gehalten werden kann und trotzdem flexible und getestete Funktionen zur Verfügung standen. Diese können bei der Verwendung von freier quelloffener Software auf das Stück angepasst werden, ohne selbst eine Bibliothek entwickeln zu müssen.

Bei einer Recherche wurden folgende echtzeitfähige Bibliotheken für Linux Systeme mit GPL³⁴ bzw. LGPL³⁵ Lizenzen ausgewählt, die für dieses Projekt in Frage kamen.

- [libsoundtouch](#)
- [librubberband](#)

4.2 Implementierung

Der Quellcode für Artistic Time-Stretching ist im Repository [artistic_time_stretching](#) zugänglich.

Weitere Software wurde im Repository [ave-absurdum](#) entwickelt, in dem die Effekte von Artistic Time-Stretching verwendet werden.

³⁴GNU General Public License

³⁵GNU Lesser General Public License

In diesem Abschnitt wird die Kenntnis der Programmiersprache [Pure Data](#) kurz Pd vorausgesetzt. Wichtig ist die Unterscheidung zwischen Pd-Subpatch, Pd-Abstraction und Pd-External.

4.2.1 Pd-Abstraction: Realtime TSM basierend auf OLA

Dies ist der artefaktreiche Algorithmus *Realtime Time Scale Modification basierend auf Overlap and Add* als Pd-Abstraction (rt_tsm_OLA~). Die Fensterfunktion wird extern zur Verfügung gestellt, was auch einen Einfluss auf die entstehenden Artefakte hat. Zusätzlich ist Pitch-Shifting und die Einstellung der Frame-Länge möglich. Der Effekt arbeitet mit einer Delay-Line, die als Audiobuffer vorwärts und rückwärts durchlaufen werden kann. Die Länge des Audiobuffers wird dem Objekt als Argument übergeben und ist somit vordefiniert.

Die GUI (siehe Abbildung 1) wurde so gestaltet, dass oft verwendete Referenzwerte für alle Parameter, die auch mit Schieberegler eingestellt werden können, verfügbar sind.

Bedienelemente:

- frame length: Frame-Länge in ms
- speed: Abspielgeschwindigkeit
- transp: Transpositionsfaktor
- reset: Dieser Knopf stellt Abspielgeschwindigkeit und Transposition auf die Ausgangswerte und führt die Abspielposition im Buffer in die Gegenwart.
- normal: stellt Abspielgeschwindigkeit und Transposition auf die Ausgangswerte

Der Audiobuffer wird am unteren Ende der GUI graphisch dargestellt (grauer Querbalken mit der Beschriftung “position”). Man stelle sich vor, dass das Eingangssignal den Buffer von rechts nach links in Originalgeschwindigkeit durchwandere, wie etwa ein Tonband. So zeigt der orange Strich (ab jetzt playhead genannt) die aktuelle Abspielposition des Eingangssignals an, wie etwa ein Wiedergabekopf eines Tonbandgerätes oder die Nadel eines Plattenspielers. Die Differenzgeschwindigkeit des Originalsignals und des playheads ergibt die Abspielgeschwindigkeit und ist somit eine

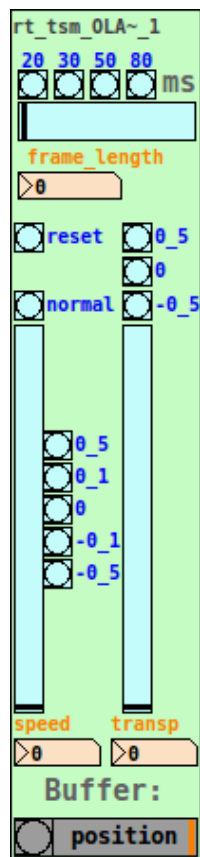


Abbildung 1: GUI für den Effekt RT TSM basierend auf OLA (Manuel Planton 31.08.2017)

graphische Rückmeldung des aktuellen Zustandes des Buffers. Steht der playhead still, wird das Signal, das den Buffer durchläuft in Originalgeschwindigkeit wiedergegeben. Bewegt er sich nach links, handelt es sich um Time-Stretching; bewegt er sich nach rechts, so handelt es sich um Time-Compression. Erreicht der playhead das linke Ende des Buffers, so ist Time-Stretching nicht mehr möglich, weil kein gespeichertes Signal mehr zur Verfügung steht. In diesem Fall wird das um die gesamte Bufferlänge verzögerte Signal zwangsweise in Normalgeschwindigkeit abgespielt. Daher wurde die Warnanzeige links neben der Buffer-Anzeige (position) realisiert. Nähert man sich dem Ende des Buffers, so fängt die Warnanzeige zu blinken an. Somit hat man noch eine Zeitreserve, um auf die Situation zu reagieren. Befindet sich der playhead ganz rechts, so wird das Signal ohne zeitliche Verzögerung wiedergegeben. In diesem Zustand ist Time-Compression nicht möglich, da kein gespeichertes Signal zur Verfügung steht, welches mit einer höheren Geschwindigkeit als Normalgeschwindigkeit durchlaufen werden kann.

4.2.2 Pd-External: Realtime TSM mittels librubberband

Dies ist der qualitativ hochwertige Algorithmus *Realtime Time Scale Modification* basierend auf *librubberband* (`rt_tsm_rubber~`).

Der erste Implementierungsversuch wurde mit `libsoundtouch` realisiert. Da es jedoch verschiedene Probleme bei der Implementierung gab, die Bibliothek lediglich mit einem WSOLA-ähnlichen Algorithmus arbeitet und die Latenz typischerweise 100ms beträgt³⁶, wurde als nächster Ansatz eine Implementierung mit `librubberband` gewählt. `Librubberband` verwendet PV-TSM mit Berücksichtigung von perkussiven Transienten und anderen Verbesserungen³⁷.

Das Pd-External `rubberband~` wurde in C mit `librubberband` entwickelt. Dieses wurde als Pitch-Shifter in der Pd-Abstraction `rt_tsm_rubber~` verwendet um TSM über Pitch-Shifting und Resampling zu realisieren. Diese Wahl wurde getroffen, da lediglich ein Pitch-Shifter als LADSPA³⁸-Plugin in der Dokumentation der Bibliothek

³⁶siehe („SoundTouch library README“ 2017)

³⁷siehe („Rubber Band Audio Time Stretcher Library“ 2017)

³⁸Linux Audio Developer's Simple Plugin API

verfügbar war³⁹ („Breakfastquay/RubberBand/source/ - Bitbucket“ 2017). Nach diesem Beispiel konnte das External entwickelt werden. Die restliche Funktionalität wurde in Pd implementiert.

Die Effektparameter, die Bedienung und auch die GUI (siehe Abbildung 2) sind dem oben beschriebenen artefaktreichen TSM-Effekt sehr ähnlich. Lediglich die Frame-Länge kann nicht bestimmt werden, da diese für optimale Qualität intern festgelegt wird.

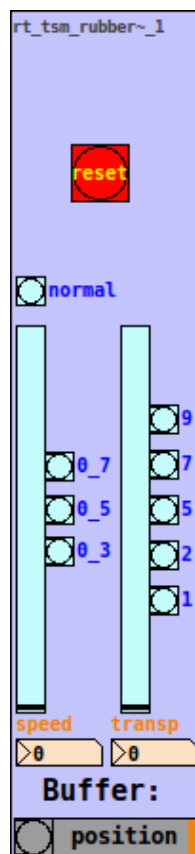


Abbildung 2: GUI für den Effekt RT TSM mittels librubberband (Manuel Planton 31.08.2017)

³⁹siehe RubberBandPitchShifter.cpp

4.2.3 Integration in ave-absurdum

Die zwei TSM-Effekte wurden in das Repository [ave-absurdum](#) integriert, das den Pd Haupt-Patch mit GUI zur Live-Performance enthält.

Folgende Features und Objekte wurden zusätzlich entwickelt:

- Integration in das Speichersystem (data storage - ds) für Presets und Szenen
- Lautstärkeregelung und Metering für Ein- und Ausgangssignale der TSM-Effekte
- Loop-Player für alle Eingangssignale
- eine 8 auf 4 Signalschaltmatrix, um die Mikrofonsignale auf die 4 TSM-Effekte für die jeweilige Szene aufzuteilen
- MIDI-Kontrolle mit zwei Fader-Boards (Behringer BCF2000 & Korg Nano Kontrol)

4.2.4 Haupt-GUI

Die Haupt-GUI (siehe Abbildung 3) wurde zur Live-Performance im Stück genutzt. Sie enthält alle Bedienelemente, die für die Erstellung der AVE verwendet wurden.

Strukturell ist die Haupt-GUI in drei Teile gegliedert:

links: (Aufzählung von oben nach unten)

- Monitoring, Master-Fader, Set und Szenen des Speichersystems für *links* und *mitte*
- Soundfile-Player, Raum in Raum Ambisonic Hall
- Lautstärke für Soundfile-Playback im Foyer, Loop-Generator für das Finale

mitte: Mixer für Live-Verstärkung

rechts: Artistic Time-Stretching Bereich (Aufzählung von links nach rechts)

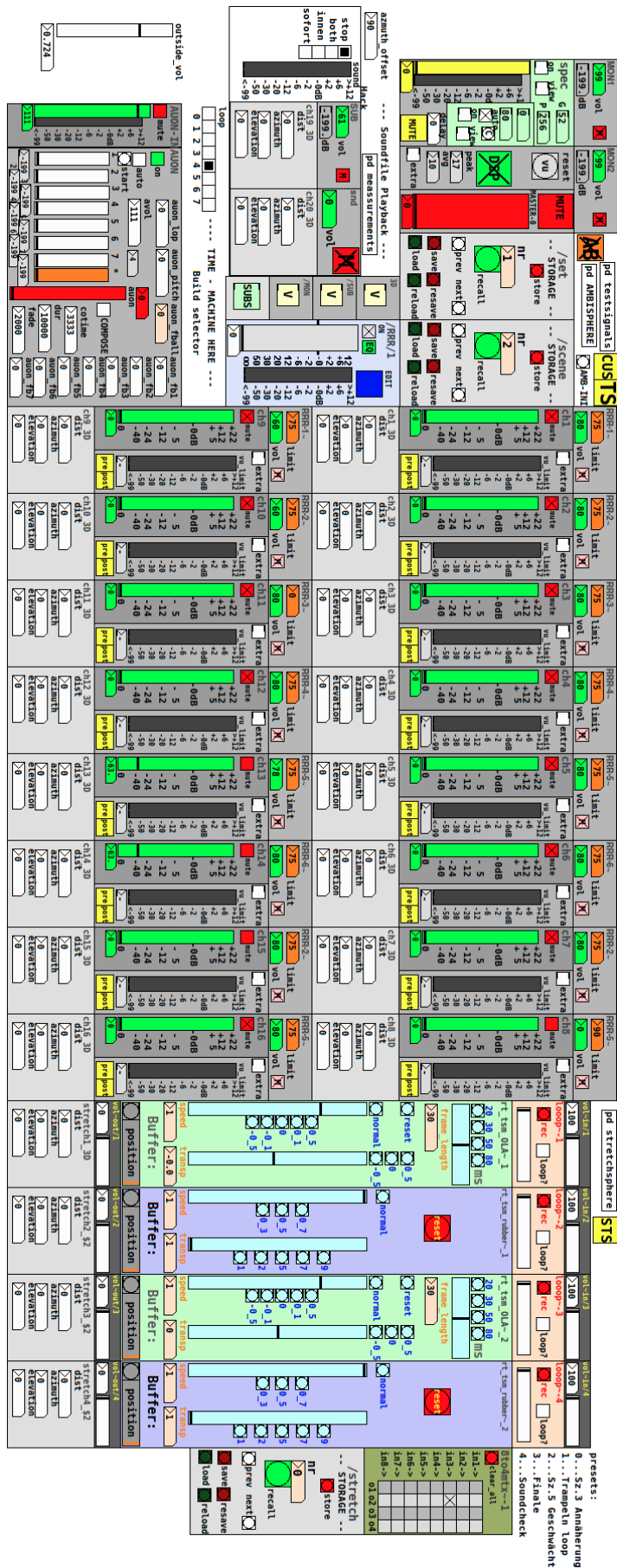


Abbildung 3: Haupt-GUI für ave-absurdum (Manuel Planton 30.10.2017)

- zwei Kanäle mit TSM basierend auf OLA (grün)
- zwei Kanäle mit TSM basierend auf librubberband (violett)
- 8 auf 4 Signalschaltmatrix
- Szenen des Speichersystems für den Artistic Time-Stretching Bereich

Die Kanäle im Artistic Time-Stretching Bereich enthalten die erwähnten Zusätze wie die Lautstärkeregelung und Metering für Ein- und Ausgänge und einen Loop-Player. Die Spatialisation der Signale erfolgt über die grauen Boxen am unteren Ende der Kanäle.

5 ave-absurdum

Ziel des Projekts ave-absurdum war es, eine AVE⁴⁰ mit der Ambisonics-Anlage im Györgyi-Ligeti-Saal als Teil des Theaterstücks⁴¹ in Echtzeit zu erzeugen. Das gleichnamige Repository [ave-absurdum](#) diene als zentraler Projektordner und enthält die verwendete Software.

5.1 Das Audiosystem

Die Beschallungsanlage im Györgyi-Ligeti-Saal ist als Ambisonic-Anlage ausgeführt und ermöglicht so die dreidimensionale Beschallung mit einer Lautsprecherhemisphäre (siehe Abbildung 4).

Die Verteilung der Lautsprecher ist in Abbildung 5 zu sehen.⁴²

Die Mikrofonierung im Györgyi-Ligeti-Saal gestaltete sich folgendermaßen:

- 6 gehängte Mikrofone (Superniere, Hypernieren) als Hotspots im Raum (siehe Abbildung 4)
- Headsets für Schauspieler
- ein Richtmikrofon (Keule) in Bodennähe für trampelnden Chor

Die gehängten Mikrofone wurden hauptsächlich als elektroakustische Hotspots im Raum genutzt, unter denen sich der Chor bewegte. Da der Chor oft in Bewegung war, wurden einzelne Sängerinnen und Sänger unterschiedlich nahe mikrofoniert. Dies resultierte in einem Effekt, der diese Stimmen abwechselnd aus dem Gesamtklang des Chores hervor hob. Stand der Chor still, so fungierte diese Anordnung als eine Art traditionelle Chormikrofonierung.

Alle Mikrofonensignale waren an beiden Computern zur Signalverarbeitung verfügbar. Zusätzlich wurden Audio-Streams von den anderen Aufführungsorten bereitgestellt.

⁴⁰siehe [AVE - Auditory Virtual Environment](#)

⁴¹siehe [Das Projekt](#)

⁴²Die Abbildung soll lediglich einen Überblick der Audioanlage verschaffen. Nicht lesbare Texte in der Abbildung sind also zu vernachlässigen.



Abbildung 4: Lautsprecherhemisphäre im Györgyi-Ligeti-Saal (Manuel Planton 15.11.2016)

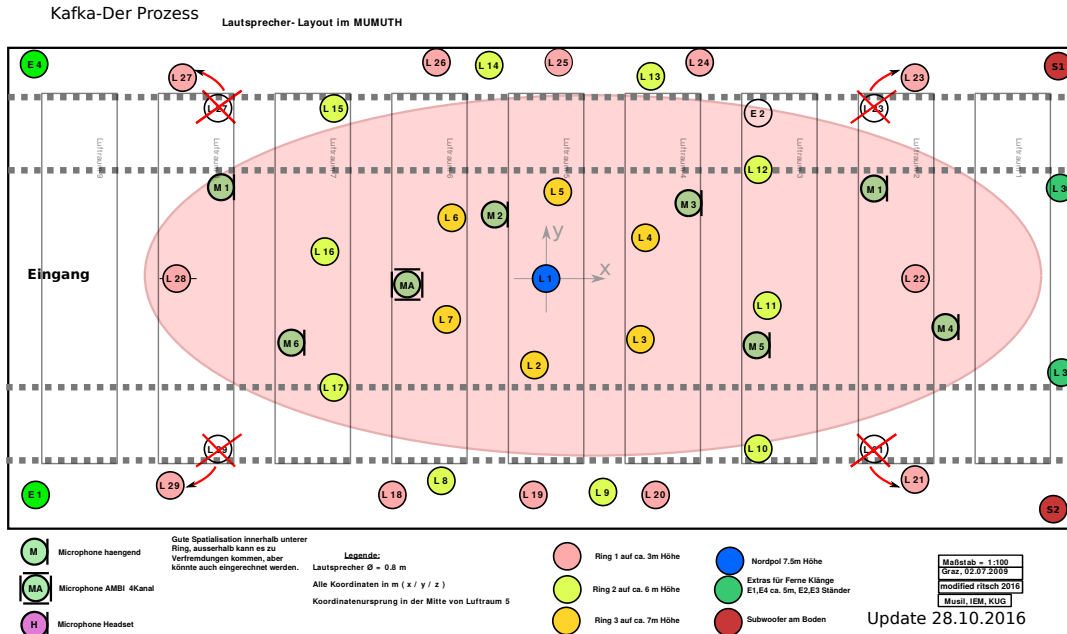


Abbildung 5: Lautsprecherpositionen im Györgyi-Ligeti-Saal (Thomas Musil bearb. Winfried Ritsch 28.10.2016)

5.2 Aufgabenbereiche

Die AVE im Györgyi-Ligeti-Saal für das Theaterprojekt wurde von einem Team aus drei Personen erstellt:

Paul Tirk war für die Live-Mischung und für den Ambisonics-Hall verantwortlich und arbeitete gleichzeitig mit mir auf einem Linux-PC. In der Live-Performance-Situation war dies mit getrennten Hardware-Controllern möglich. Christof Ressi bediente an einem Mac ein räumliches Spektraldelay und setzte auch andere Effekte ein. Meine Aufgabe war das Artistic Time-Stretching und ich war auch für einzelne Effekte zuständig und bediente den Loop am Ende des Finales.

In Abb. 6 ist der Arbeitsplatz des gesamten Teams zu sehen.

Mit diesen Effekten konnte eine AVE erzeugt werden, deren Charakteristik je nach Wunsch der Regisseure geändert wurde. Bis auf ein Sample, das in der Pause vor dem Finale abgespielt wurde, wurden ausschließlich Live-Signale zur Erzeugung der AVE verwendet.

Meine Hauptaufgabe war jedoch das Artistic Time-Stretching, welches hauptsächlich



Abbildung 6: Arbeitsplatz (Manuel Planton 09.11.2016)

mit den Sprachsignalen der Schauspieler arbeitete.

5.3 Proben und Live-Performance

Nachdem die technische Funktion der gesamten Anlage inklusive Software bereitgestellt war, konnten die Proben beginnen. In Einzelproben mit den jeweiligen Subregisseuren wurde die AVE entwickelt, die schlussendlich bei den Aufführungen erzeugt werden sollte. Dies gestaltete sich zuerst über freie Improvisation, bis das Team und die Regisseure ein Bild davon hatten, welche klanglichen Möglichkeiten in diesem Kontext erreicht werden konnten. Nach dieser experimentellen Phase wurden die Vorstellungen der Regisseure als Komposition an den Text geknüpft. Ein Auszug der Notation im Text für Artistic Time-Stretching in der Advokat Szene ist in Abbildung 7 und 8 zu sehen.

In der Advokat Szene wurde Artistic Time-Stretching auf mehrere Textteile (in den Abbildungen eingekreist) angewendet⁴³. Beispielsweise wurde mit dem Geschnaufe des Advokaten⁴⁴ am Anfang der Szene eine bedrückende, verwirrende Atmosphäre mit starken Artefakten mit hoher Rauigkeit erzeugt. Dies unterstrich seine Krankheit. In der Textstelle am Anfang von Abbildung 8 geschieht ein Aufschrei, der ebenfalls mit einer hohen Rauigkeit und starken Artefakten zeitlich extrem gedehnt wurde, was diese Aktion intensivierte. Im Text des Mädchens am Ende der Szene wurden einzelne Satzteile für Time-Stretching mit hoher Qualität und geringer Lautstärke ausgewählt. Damit wurde das Empfinden von K. in dieser Situation als AVE hörbar, in der er verwirrt und geschwächt ist und das Mädchen auf ihn einredet.

In den Gesamtproben wurde die Erzeugung der AVE weiter eingeübt und hin und wieder musste auch die Software an die Vorstellungen der Regisseure angepasst werden.

Die Steuerung von Artistic Time-Stretching und weiteren Effekten wurde hauptsächlich mit der Bedienung von zwei MIDI-Controllern (Behringer BCF2000 & Korg Nano Kontrol) realisiert (siehe Abb. 9).

⁴³siehe Mögliche Effekte

⁴⁴siehe Kafkas 'Der Prozess' 4. Absatz

Trampeln

PRESET 2

5 SZENE ADVOKAT MUMUTH

Josef K. | Advokat | Leni

Berenice Advokat: Geschwänze

Advokat: Ich warte schon sehr lange auf Sie,

K.: Ich gehe bald wieder weg.

Advokat: Ich werde Sie nächstens zu dieser späten Stunde nicht mehr verlassen.

K.: Das kommt meinem Anliegen entgegen

Advokat: Setzen Sie sich. War sie wieder zudringlich?

K.: Zudringlich?

Advokat: Ja, Leni...

Advokat: Sie haben doch wohl ihre Zudringlichkeit schon bemerkt?

Es ist eine Sonderbarkeit Lenis, die ich ihr übrigens längst verzeihen habe. Diese Sonderbarkeit besteht darin, dass Leni (die meisten Angeklagten schön findet). Sie hängt sich an alle, liebt alle. doch nicht alle bestraft.

Advokat: Sind Sie heute mit einer bestimmten Absicht zu mir gekommen?

K.: Ja, ich wollte Ihnen sagen, dass ich Ihnen mit dem heutigen Tage meine Vertretung entziehe.

Advokat: Was? Verstehe ich Sie recht?

K.: Ich nehme es an,

Advokat: Nun, wir können ja auch diesen Plan besprechen,

K.: Es ist kein Plan mehr,

Advokat: Mag sein,

Advokat: wir wollen aber trotzdem nichts übereilen.

K.: Es ist nicht übereilt. Der Entschluss ist endgültig.

Ich danke Ihnen für Ihre freundliche Gesinnung. Ich erkenne auch an, dass Sie sich meiner Sache so sehr angenommen haben, wie es Ihnen möglich ist und wie es Ihnen für mich vorteilhaft scheint. Ich habe die Überzeugung gewonnen, dass das nicht genügend ist. Verzeihen Sie mir, die Sache aber ist wichtig genug.

Advokat: Sie sind mir im Laufe der Zeit lieb geworden. Ich gestehe das offen ein. Ich brauche mich dessen nicht zu schämen.

+ Hall

K.: Ich habe kaum mehr die Wahl, den Prozess anzunehmen oder abzulehnen. Ich stehe mitten darin und muss mich wehren. Das Gericht ist eine große Organisation und der Sinn dieser großen Organisation besteht darin, dass unschuldige Personen verhaftet werden und gegen sie ein sinnloses und meistens ergebnisloses Verfahren eingeleitet wird.

Advokat: Von einem gewissen Zeitpunkt der Praxis an, ereignet sich nichts wesentlich Neues mehr. Wie viele Parteien sind in ähnlichen Stadien der Prozesse ähnlich wie Sie vor mir gestanden und haben ähnlich gesprochen!

Text geändert

13

Abbildung 7: Text mit Notation für Effekte 1 (Kafka und Studierende der Universität für Musik und darstellende Kunst Graz, Institut 11 - Bühnengestaltung (2016) bearb. Manuel Planton 31.10.2017)

Tisch

K: Niemals früher hatte ich so große Sorgen wegen des Prozesses wie seit der Zeit, seitdem Sie mich vertreten. Unaufhörlich und immer gespannter erwartete ich Ihr Eingreifen, aber es blieb aus.

Advokat: Sie werden bemerkt haben, dass ich zwar eine große Kanzlei habe, aber keine Hilfskräfte beschäftige, heute arbeite ich allein. Und außerdem wurde ich vor Überanstrengung krank.

Es ist oft besser im Keller

K.: Was wollen Sie in meiner Sache unternehmen, wenn Sie die Vertretung behalten?

Advokat: In dem, was ich für Sie bereits unternommen habe, weiter fortfahren. *NICHT* in Ketten, als frei zu sein.

+ Hall

WECHSEL

↳ rubber!

Auftritt Erzählerin+Mädchenchor

Mädchen: Wollen Sie sich nicht setzen? Sie haben ein wenig Schwindel, nicht? Machen Sie sich darüber keine Gedanken, das ist hier nichts Außergewöhnliches, fast jeder bekommt einen solchen Anfall, wenn er zum ersten Mal herkommt. Sie sind zum ersten Mal hier? Nun ja, das ist also nichts Außergewöhnliches. Die Sonne brennt hier auf das Dachgerüst, und das heiße Holz macht die Luft so dumpf und schwer. Der Ort ist deshalb für Büroräumlichkeiten nicht sehr geeignet, so große Vorteile er allerdings sonst bietet. Aber was die Luft betrifft, so ist sie an Tagen großen Parteienverkehrs, und das ist fast jeder Tag, kaum mehr atembar. Wenn Sie dann noch bedenken, dass hier auch vielfach Wäsche zum Trocknen ausgehängt wird - man kann es den Mietern nicht gänzlich untersagen -, so werden Sie sich nicht mehr wundern, dass Ihnen ein wenig übel wurde. Aber man gewöhnt sich schließlich an die Luft sehr gut. Wenn Sie zum zweiten- oder dritten Mal herkommen, werden Sie das Drückende hier kaum mehr spüren. Fühlen Sie sich schon besser?

Siehe verlaucht!

Hier können Sie nicht bleiben, hier stören wir den Verkehr -

Auftritt Richter

Richter: Ruhe, ich lasse den Saal räumen.

bes. Türhüter sind

Zum Finale: AVON fertig machen

- LOP: auf 10064

- Puteh: 1

→ AVON-roster aufpassen

Abbildung 8: Text mit Notation für Effekte 2 (Kafka und Studierende der Universität für Musik und darstellende Kunst Graz, Institut 11 - Bühnengestaltung (2016) bearb. Manuel Planton 31.10.2017)



Abbildung 9: Live-Setup Artistic Time-Stretching (Manuel Planton 16.11.2016)

Die Live-Performance zur Erzeugung der AVE bestand aus einer Mischung der Ausführung notierter Komposition und von Improvisation, die auf die gegenwärtige Interpretation des Schauspielers einging und dadurch eine AVE entstehen ließ, die der Schauspieler verstand und mit der er auch umgehen konnte. Es war wichtig, die Reproduzierbarkeit von einzelnen Effekten (insbesondere stark hörbare) zu gewährleisten, damit sich Schauspieler an die AVE der jeweiligen Szene gewöhnen und in ihr ohne Probleme und ohne Ablenkungen agieren konnten.

6 Fazit

Im Vorfeld wurde das Konzept Artistic Time-Stretching entwickelt und TSM-Algorithmen nach Qualitätsmerkmalen und erzeugten Artefakten eingeteilt. Diese Einteilung ermöglichte die Auswahl von Algorithmen für die Verwendung als Echtzeiteffekte zur Erstellung einer Auditory Virtual Environment in einer Theaterproduktion von Kafkas ‘Der Prozess’. Diese Algorithmen wurden als Echtzeitprogramme in Pd und C realisiert, was die Verwendung von flexiblen und freien Pd-Objekten in einem Pd-Hauptpatch ermöglichte. Im Projekt ave-absurdum als Teil des Theaterstückes, welches im Györgyi-Ligeti-Saal stattfand, wurde die entwickelte TSM-Software und weitere Effekte zur Erzeugung der AVE verwendet. In einer experimentellen Phase wurden die möglichen TSM-Effekte erforscht und den Regisseuren der jeweiligen Szenen dargeboten, um schließlich zusammen eine an den Text der Schauspielerinnen und Schauspieler geknüpfte Komposition zu erarbeiten. Die Live-Performance im Stück bestand aus einer Mischung dieser Komposition und einer situationsbezogenen Improvisation, welche auf die aktuelle Interpretation der Schauspielerinnen und Schauspieler einging. In den Vorstellungen konnte eine AVE geschaffen werden, die als Teil des Stückes selbst, dessen kafkaesken Charakter unterstrich.

7 Links

Diese Links waren im Februar 2018 zugänglich. Das Originaldokument ist unter [artistic_time_stretching](#) “https://git.iem.at/mi/artistic_time_stretching” zugänglich.

[MUMUTH](#)⁴⁵ <https://www.kug.ac.at/studium-weiterbildung/studium/infrastruktur/das-mumuth.html>

[TiP](#)⁴⁶ https://www.kug.ac.at/ueber-die-universitaet/ueber-die-universitaet/gebaeude-in-graz-oberschuetzen.html?tx_kuglocations_pi1%5Bgeb_nr%5D=1438&cHash=430ff072b0917313efff013997c4069e

[ave-absurdum](#) <https://git.iem.at/cm/ave-absurdum>

[ave-streamer](#) <https://git.iem.at/thomasdeppisch/ave-streamer>

[ave-tetrahedron](#) <https://git.iem.at/cm/ave-tetrahedron>

[ave-kafka](#) <https://git.iem.at/cm/ave-kafka>

[Projektdokumentation](#) <https://iaem.at/kurse/projekte/ave-kafka>

[artistic_time_stretching](#) https://git.iem.at/mi/artistic_time_stretching

[libsoundtouch](#) <https://www.surina.net/soundtouch/>

[librubberband](#) <http://breakfastquay.com/rubberband/>

[Pure Data](#) <https://puredata.info/>

⁴⁵Haus für Musik und Musiktheater

⁴⁶Theater im Palais

Literaturverzeichnis

„Antares Auto-Tune“. 2017. *Wikipedia*. https://de.wikipedia.org/w/index.php?title=Antares_Auto-Tune&oldid=161363136.

Baran, Tom. 2017. „Tom Baran - Autotalent“. Zugegriffen März 8. <http://tombaran.info/autotalent.html>.

Blauert, Jens, Hrsg. 2005. *Communication acoustics*. Berlin: Springer-Verlag.

„Breakfastquay/RubberBand/source/ - Bitbucket“. 2017. Repository. Zugegriffen August 31. <https://bitbucket.org/breakfastquay/rubberband/src>.

Cliff, Dave. 2000. „Hang the DJ: Automatic Sequencing and Seamless Mixing of Dance-Music Tracks“. *HP LABORATORIES TECHNICAL REPORT HPL 104*.

Cockos Inc. 2017. „Up and Running: A REAPER User Guide - ReaperUserGuide533e.pdf“. Februar. <http://dlz.reaper.fm/userguide/ReaperUserGuide533e.pdf>.

Driedger, Jonathan, und Meinard Müller. 2016. „A Review of Time-Scale Modification of Music Signals“. *Applied Sciences* 6 (2): 57. doi:[10.3390/app6020057](https://doi.org/10.3390/app6020057).

Fitzgerald, Derry. 2010. „Harmonic/Percussive Separation Using Median Filtering“. *Conference Papers* 13th International Conference on Digital Audio Effects (DAFX10), Graz, Austria, 2010. (Jänner): 4. <http://arrow.dit.ie/argcon/67>.

„Franz Kafka: Der Prozess“. 2016. Oktober 31. <https://www.kug.ac.at/news-veranstaltungen/news/kug-aktuell/details/article/franz-kafka-der-prozess.html>.

Kafka, Franz. 2006. *Der Prozess: Roman*. Köln: Anaconda-Verl.

Kafka, Franz, und Studierende der Universität für Musik und darstellende Kunst Graz, Institut 11 - Bühnengestaltung. 2016. *Der Prozess: Bühnenfassung*.

Laroche, J., und M. Dolson. 1997. „Phase-vocoder: about this phasiness business“.

In, 4. IEEE. doi:[10.1109/ASPAA.1997.625603](https://doi.org/10.1109/ASPAA.1997.625603).

„LV2“. 2017. Zugriffen März 8. <http://lv2plug.in/>.

Malham, David G., und Anthony Myatt. 1995. „3-D Sound Spatialization using Ambisonic Techniques“. *Computer Music Journal* 19 (4): 58–70. doi:[10.2307/3680991](https://doi.org/10.2307/3680991).

Moinet, Alexis, Thierry Dutoit, und Philippe Latour. 2013. „Audio Time-Scaling for Slow Motion Sports Videos“. In *Proceedings of the International Conference on Digital Audio Effects (DAFx), Maynooth, Ireland*, 7. http://dafx13.nuim.ie/papers/51.dafx2013_submission_18.pdf.

„Rubber Band Audio Time Stretcher Library“. 2017. Zugriffen August 31. <http://breakfastquay.com/rubberband/technical.html>.

„Sliding Time Scale / Pitch Shift - Audacity Manual“. 2017. Zugriffen März 8. http://manual.audacityteam.org/man/sliding_time_scale_pitch_shift.html.

„SoundTouch library README“. 2017. Zugriffen August 31. <https://www.surina.net/soundtouch/README.html>.

„The Ardour Manual“. 2017. Zugriffen März 7. <http://manual.ardour.org/editing/editing-regions-and-selections/stretching/>.

„Virtual Studio Technology“. 2016. *Wikipedia*. https://de.wikipedia.org/w/index.php?title=Virtual_Studio_Technology&oldid=156123370.

„Zeta-AT1 - Quick Guide“. 2017. Zugriffen März 8. <http://kokkinizita.linuxaudio.org/linuxaudio/zita-at1-doc/quickguide.html>.

Zölzer, Udo, und Xavier Amatriain. 2002. *DAFX Digital Audio Effects*. Chichester: Wiley.

Abbildungsverzeichnis

1	GUI für den Effekt RT TSM basierend auf OLA (Manuel Planton 31.08.2017)	30
---	---	----

2	GUI für den Effekt RT TSM mittels librubberband (Manuel Planton 31.08.2017)	32
3	Haupt-GUI für ave-absurdum (Manuel Planton 30.10.2017)	34
4	Lautsprecherhemisphäre im Györgyi-Ligeti-Saal (Manuel Planton 15.11.2016)	37
5	Lautsprecherpositionen im Györgyi-Ligeti-Saal (Thomas Musil bearb. Winfried Ritsch 28.10.2016)	38
6	Arbeitsplatz (Manuel Planton 09.11.2016)	39
7	Text mit Notation für Effekte 1 (Kafka und Studierende der Universität für Musik und darstellende Kunst Graz, Institut 11 - Bühnengestaltung (2016) bearb. Manuel Planton 31.10.2017)	41
8	Text mit Notation für Effekte 2 (Kafka und Studierende der Universität für Musik und darstellende Kunst Graz, Institut 11 - Bühnengestaltung (2016) bearb. Manuel Planton 31.10.2017)	42
9	Live-Setup Artistic Time-Stretching (Manuel Planton 16.11.2016) . .	43