Hauptseminar Kommunikationssysteme
Compressed Compute-and-Forward mit korrelierten Audiosignalen

Lucas Weber, Raphael Hildebrand, Florian Roth, Orell Garten

13.07.2016

Inhaltsverzeichnis

1 Aufgabenstellung

Es ist ein Programm zu entwerfen, welches Audiodateien einliest und dateiweise die beiden Stereo-Kanäle miteinander korreliert. Es sollen Maßzahlen entworfen und berechnet werden, die wesentliche charakteristische Eigenschaften der Korrelationsfunktion, insbesondere den Anteil dominanter Komponenten und deren Abklingverhalten, widerspiegeln. Dafür sind Audiosignale aufzunehmen, bezüglich der verwendeten Maße zu klassifizieren und entsprechend ihrer Klassifzierung systematisch abzuspeichern.

2 Motivation

Im Jahr 2022 werden 500 Milliarden internetfähige Geräte erwartet, die mit einander kommunizieren sollen. Das führt zu extrem hohen Datenmengen, die in kürzestmöglicher Zeit von A nach B transportiert werden müssen. Große Herausforderungen bestehen darin, dass man sehr kurze Verzögerungszeiten und eine hohe Widerstandsfähigkeit garantieren muss. Idealerweise benötigen die Geräte wenig Energie. Ein Ansatz zur Lösung dieses Problems ist die Netzwerkcodierung.
In bestimmten Szenarien ist eine große Anzahl an Geräten mit Sensorik zu Erfassung der Umgebung mit hohen Anforderungen an die Netzwerkkapazität zur Übertragung der erfassten Daten verbunden. Es stellt sich die Frage, wie viele Sensoren für eine ausreichend genaue Abbildung benötigt werden. Da die Quellen teilweise korrelierte Datenströme erzeugen, lässt sich die zu übertragende Gesamtdatenmenge reduzieren, was durch eine geeignete Kombination von Netzwerkcodierung mit Methoden des Compressed Sensing erreicht werden soll.

3 Theoretische Vorbetrachtung

3.1 Kreuzkorrelationsfunktion

Die Basis für die Bemessung der aufgenommenen Audiosignale bildet die sogenannte Kreuzkorrelationsfunktion (KKF). Wie in der Aufgabenstellung schon beschrieben werden an ihr die Bemessungsparameter festgelegt. Aufgrund der verschiedenen Blocklängen und der Masse an Daten, die korreliert werden sollen muss die Berechnung der KKF effizient und zeitsparend implementiert werden. Im Folgenden Abschnitt wird die KKF kurz theoretisch eingeführt und das das mathematische Konzept erklärt, auf dem die effiziente Berechnung der KKF beruht.

Zuerst haben wir für die KKF eine Funktion genutzt, die zur Berechnung Summen verwendete. Dabei ergab sich, dass die Berechnung zu langsam war. In der nun vorliegenden Octave-Version wird die KKF im Frequenzbereich berechnet.

3.1.1 Berechnungsvorschrift

Die KKF ist als aus zwei verschiedenen Funktionen gebildeter Erwartungswert definiert. Hier werden die Formeln allgemein für die Korrelation der Prozesse **X** und **Y** angegeben.

$$\psi_{\mathbf{X}Y}(t_1, t_2) = E\{\mathbf{X}(t_1) \cdot \mathbf{Y}(t_2)\} \tag{1}$$

Reale aufgenommene Audiosignale s(t), die hier mit durch die KKF verrechnet werden, sind in jedem Fall Energiesignale, da sie rein reell sind, einen begrenzten Wertebereich haben und nach einer bestimmten Zeit enden.

$$Signalenergie := \int_{-\infty}^{\infty} s^2(t)\, dt < \infty \tag{2}$$

Da die aufgenommenen Audiosignale zeit-diskrete Energiesignale sind wird hier auch nur die zeit-diskrete Kreuzkorrelation beschrieben. Für zeit-diskrete Energiesignale ergibt sich die folgende Berechnungsvorschrift, wobei x(n) und y(n) Realisierungen der Prozesse **X** und **Y** sind.

$$\boxed{\psi^E_{\mathbf{XY}}(k) = \sum_{n=-\infty}^{\infty} x(n) \cdot y(n+k)} \tag{3}$$

vgl. [ISV] S. 84 folgende

3.1.2 Kreuzkorrelation im Frequenzbereich und Faltungssatz

Am Anfang unserer Arbeit haben wir uns mit der Berechnung der Kreuzkorrelation beschäftigt. Da die Bildung der Summe der Multiplikation von x(n) und y(n+k) sehr rechenaufwändig ist, haben wir nach schnelleren Möglichkeiten gesucht die KKF der beiden Signale zu berechnen. Eine geeignet Möglichkeit ist die Berechnung der KKF im Frequenzbereich. Für die Berechnung der KKF im Frequenzbereich macht man sich die Ähnlichkeit der KKF zur Faltung und den Faltungssatz zunutze.

KKF als Faltung

Zeit-kontinuierlicher Fall:

$$\psi^E_{\mathbf{XY}}(\tau) = x(-\tau) * y(\tau) \tag{4}$$

Zeit-diskreter Fall:

$$\psi^E_{\mathbf{XY}}(k) = x(-k) * y(k) \tag{5}$$

[ISV] Formel (2.217) S.89

Faltungssatz in Verbindung mit KKF

Wir schreiben zuerst die die KKF als Faltung. Danach transformieren wir die Faltung in den Frequenzbereich. Durch geschicktes Erweitern und Substitution findet man einen Ausdruck, um die KKF im Frequenzbereich zu berechnen.

$$\psi(t) = x(-t) * y(t) = \int_{-\infty}^{\infty} x(-\tau) \cdot y(t-\tau)\, d\tau \tag{6}$$

$$\tag{7}$$

$$\underline{\Psi}(\omega) = \int_{-\infty}^{\infty} x(-t) * y(t) \cdot e^{-j\omega t}\, dt \tag{8}$$

$$= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(-\tau) \cdot y(t-\tau)\, d\tau \right] \cdot e^{-j\omega(t-\tau+\tau)}\, dt \tag{9}$$

$$= \int_{-\infty}^{\infty} x(-\tau) \cdot e^{j\omega(-\tau)} \underbrace{\left[\int_{-\infty}^{\infty} y(t-\tau) e^{-j\omega(t-\tau)}\, dt \right]}_{*}\, d\tau \tag{10}$$

Durch Substitution von $(t-\tau)$ durch t' ergibt sich * zur Fourier-transformierten $\underline{Y}(\omega)$ von y(t). Der restliche Ausdruck wird durch das positive Vorzeichen in der e-Funktion zur komplex konjugierten Transformierten $\underline{X}^*(\omega)$ der Funktion x(t).

Die KKF lässt sich im Frequenzbereich also als

$$\boxed{\underline{\Psi}(\omega) = \underline{X}^*(\omega) \cdot \underline{Y}(\omega)} \tag{11}$$

schreiben.
vgl. [ISV] S.180

Wenn man nun die KKF im Frequenzbereich zeitsparend durchführen will, muss man die FFT für x(k) und y(k) (dabei $k \in \mathbb{N}_0^+$) der Länge N durchführen. Dabei muss man beachten, dass bei der FFT ein Linienspektrum ergibt. Die FFT beruht vor allem auch auf der Annahme, dass sich die N diskreten Werte periodisch wiederholen. Durch die IFFT von $\underline{\Psi}(\omega)$ ergibt sich also die periodische KKF $\tilde{\psi}(t)$.
vgl. [ISV] S.135

3.1.3 Zero Padding

Die periodische KKF $\tilde{\psi}(t)$ ist in unserem Projekt zudem die bessere Wahl. Berechnet man die KKF im Frequenzbereich wird durch die implementierten Funktionen von Octave, so wie von Python, sogenanntes Zero-Padding durchgeführt. Vorstellen kann man sich das als auffüllen der Daten mit N Werten = 0 an jeweils den Rändern einer der beiden Funktionen, da durch die Verschiebung über den Rand der anderen hinausragt.

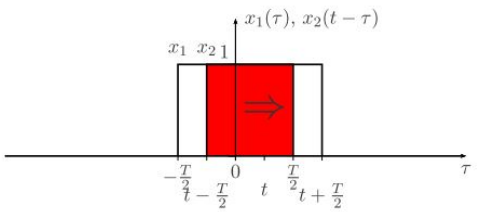

Abbildung 1: Anschauliches Beispiel zur Faltung

vgl. [NT] Seite 28
Wie man in Abbildung 1 erkennen kann ist die Funktion x_2 soweit verschoben, dass sie über den

rechten Rand der Funktion x_1 hinausragt. Wenn man die beiden Rechtecksignale als Bereiche sieht in denen echte Funktionswerte liegen, würde nun auf der positiven x-Achse ab $T/2$ mit Nullen aufgefüllt werden. Dadurch bekommt die KKF automatisch ein Abklingverhalten an ihren Rändern, das nur durch die begrenzte Anzahl an Werten verursacht wird. Die Korrelation nimmt an den Rändern nicht zwingend ab. Durch dieses Abklingverhalten würden unsere Ergebnisse also abgefälscht werden. Die periodische KKF ist also aussagekräftiger, da unser reales Signal in jedem Fall an den Rändern der korrelierten Blöcke nicht auf Null abklingt.Wenn man die periodische KKF im Zeitbereich berechnen möchte, müsste man an den Rändern der Funktionen nochmal die Funktion x_1 anhängen.

3.2 Maßzahlen

3.3 Gauß-Regression

4 Verfügbare Technik

4.1 Software

Softwareseitig haben wir Octave benutzt. Als freie Alternative zu Matlab vereint Octave gute Performance, syntaktische Gleichheit zu Matlab, sowie kostenfreie Benutzung unter einem Dach. Im Vergleich mit Python haben wir festgestellt, dass die Geschwindigkeit aufwendiger Rechnungen, wie der Korrelation, bei Python schlechter ist. Somit haben wir uns für Ocatve entschieden. Um unseres selbstgeschriebenes Programm zu verifizieren haben wir eine Autokorrelation durchgeführt und diese mit der von Audactity berechneten AKF verglichen. Wir sind dabei zu dem Ergebnis gekommen, dass unser Programm funktioniert.

4.2 Hardware

Uns standen zwei hochwertige Kondensatormikrofone (M5 Matched Pair Compac 1/2"Cardioid Condenser Microphones von Rode zur Verfügung. Diese Mikrofone sind für dieses Projekt besonders geeignet, da durch die Abstimmung (matched pair) nur die Unterschiede im Signal vor der Aufnahme Einfluss auf die Korrelation haben. Für die Digitalisierung der Signale stand uns ein hochwertiges Audio-USB-Interface (Scarlett 2i2 von Focusrite) zur Verfügung. Die Aufnahmen wurden in .wav gespeichert und sind somit verlustfrei. Außerdem konnten wir ein Stativ mit einer Mikrofonschiene verwenden, wodurch die Mikrofone konstanten Abstand hatten. Zur Erzeugung eines reproduzierbaren Klangsignales wurde eine portable Bluetooth-Anlage (Soundlink III von Bose) verwendet.

5 Programm

5.1 Aufbau des Programms

Das Programm ist in 3 große Teile gegliedert. Dazu zählt das Einlesen der Audiodateien, die benötigte Signalverarbeitung inklusiver Berechnung der gewünschten Parameter und das Speichern der gewonnenen Werte in Form einer Excel-Datei.

Einlesen der Audiodaten Die Audiodateien liegen im WAV-Format als Stereoaufnahme vor. Zunächst wird eine Liste mit allen Dateien in einem bestimmten Ordner erstellt, damit die Dateien nacheinander eingelesen werden können. Im nächsten Schritt werden die beiden Kanäle voneinander getrennt, um diese dann in die Signalverarbeitung zu übergeben.

Signalverarbeitung Das Kernstück der Signalverarbeitung ist eine periodische Korrelationsfunktion, die den linken und rechten Kanal miteinander korreliert. Die dabei entstandene Korrelationsfunktion wird dann weiter untersucht. Als nächstes wird eine Art Einhüllende berechnet, die ein Maß für die Steilheit der Kurve ist. Wie bereits im Abschnitt 3.3 "Gauß-Regression"beschrieben, wird dann mit Hilfe der Methode der kleinsten Quadrate eine Gauß-Glocke so angepasst, dass sie den Verlauf der Hüllkurve der KKF möglichst gut abbildet. Die Parameter ripple, σ, Gleichanteil und Zeitverschiebung des Maximums aus dem Ursprung werden danach an eine Funktion übergeben, die diese Daten in einer Excel-Tabelle speichert.

Speicherung Die Speicherung der Daten erfolgt in einer Excel-Datei. Dabei wird zu erst der Dateiname des Samples und alles dazugehörigen Werte gespeichert. Außerdem wird noch ein Link zum Graphen der Korrelationsfunktion angegeben, damit man sich diese bei der Auswahl der Test-Signale anschauen kann.

5.1.1 Programmablaufplan

5.2 Mögliche Einstellungen

Im Code gibt es diverse Eintellungen, die das Verhalten des Programms je nach Wunsch des Anwenders verändern. Diese sind am Beginn der main.m-Datei festgelegt und werden im folgenden beschrieben:

- *path* - Pfad zur Sammlung der WAV-Dateien
- *excel_path* - Pfad unter dem Excel-Datei mit Lösungen gespeichert wird
- *output* - Unterscheidung, ob Ergebnisse gespeichert oder angezeigt werden
- *calc* - Wechsel zwischen Berechnung im Zeit- und Frequenzbereich möglich.
- *x_axes* - Unterscheidung, ob Korrelation gegen Samples oder Zeit aufgetragen wird
- *priority* - Unterscheidung, ob angegebene Blocklänge oder Zeitdauer priorisiert wird
- *t_start* - Startzeitpunkt der Korrelation
- *t_dur* - Array $A1$ mit Menge an Zeitdauern die korrelierten werden sollen, Korrelation beginnt immer bei *t_start*
- *Ncor_init* - Array, mit identischer Länge zu $A1$. Gibt an wie oft korrespondierender Eintrag in $A1$ hintereinander korreliert wird.
- *Lcor* - Blocklänge der Korrelation

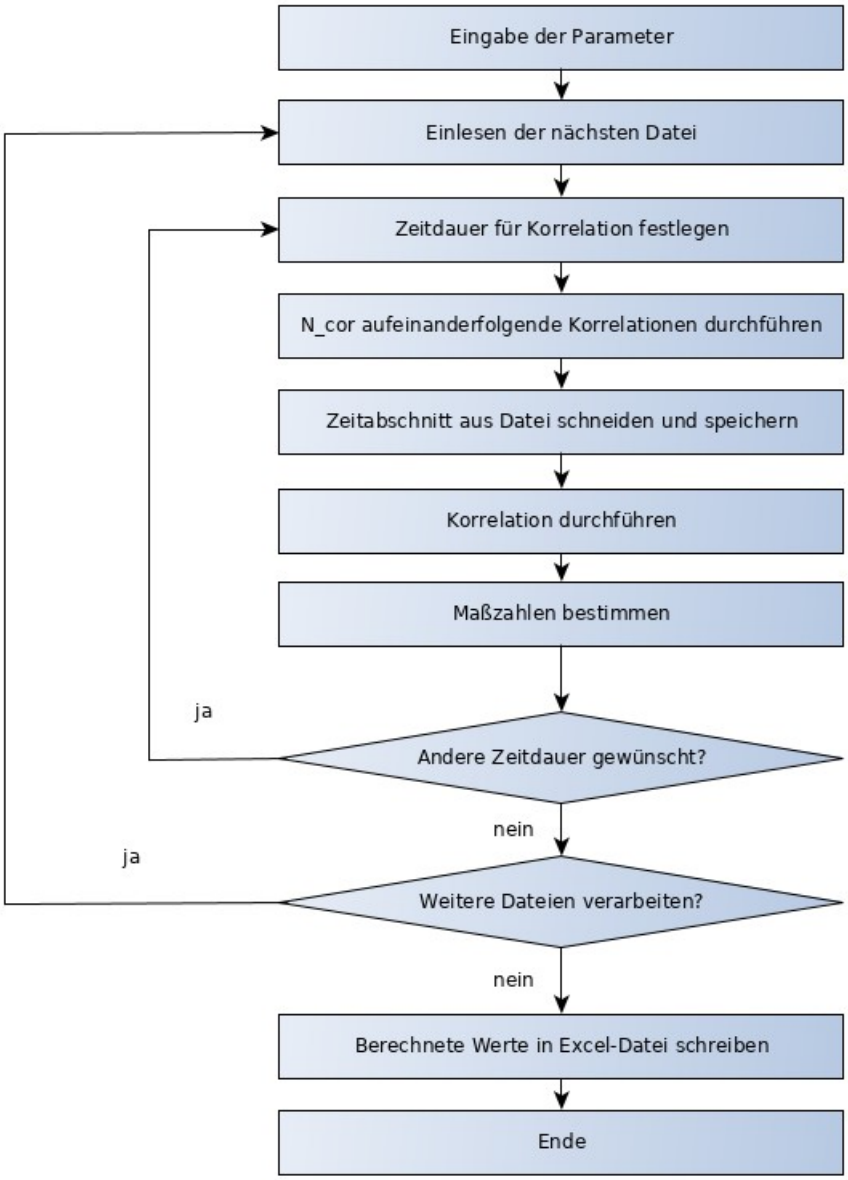

Abbildung 2: Programmablaufplan

5.3 Probleme

Zu Beginn unserer Arbeit wurde die Korrelation zuerst in Python implementiert. Allerdings nahm die Berechnungsdauer mit zunehmende Blocklänge stark zu. Da die .wav Dateien mit 44100Hz abgetastet werden werden die zu berechnenden Summen schnell sehr groß. Wie in der Theorie und im Abschnitt zur Software schon erklärt, wird nun Octave und der Fouriertransformation genutzt.

Darauf aufbauen war noch die Problemstellung der Bemessung der korrelierten Signale zu lösen. Dabei mussten Maßzahlen entstehen, die viele verschiedene Faktoren beachteten und möglichst gute Aussagen über „Peakyness" und Streuung der KKF trafen. Die Idee war dann ein Fit mit einer Funktion wie der Gauß-Kurve, die als Parameter schon eine Verteilung beinhaltet. Allerdings ist die KKF nach der Berechnung sehr ëckigünd weißt viele Nullstellen auf. Um den Fit nicht durch diese Eigenschaften verzerren zu lassen, wird die KKF durch einen Tiefpass gefiltert. Auf die geglättete Kurve angewendet, ist die angewendete Regression aussagekräftiger.

6 Signalauswahl

Wir haben für die Aufnahme der Signale viel verschiedene Situationen ausgesucht, um ein möglichst breites Spektrum an Raum-Effekten zu erhalten. Aufnahmeorte waren beispielsweise der Platz vor dem HSZ, die Wiese zwischen Physik- und Mathematikgebäude, sowie der Trefftzbau. Außerdem wurde in einer Wohnung gemessen, um Effekte von schallabsorbierenden Stoffen wie Teppich oder Bett zu erhalten. Soweit mgölich, haben wir die natürliche Geräuschkulisse am jeweiligen Ort eingefangen. Zusätzlich dazu wurde ein definiertes Signal mittels eines Lautsprechers erzeugt, um Direktschall zu nutzen. Bei diesen Aufnahmen sollten die Effekte des Raumes am deutlichsten hervortreten.

6.1 Beispielsignale

6.1.1 Signal 1 - trefftz_wiese_m

Aufnahmesituation Dieses Signal wurde auf der Wiese zwischen dem Gebäude der Mathematik- und Physikfakultät aufgenommen. Das heißt, es ist eine relativ große Freifläche mit wenigen Hindernissen mit ungehinderter Schallausbreitung. Es wurde keine zusätzliche Primärschallquelle genutzt.

Signalbeschreibung Wie in Abbildung 3 zu sehen ist, ändern sich beide Kanäle relativ langsam. Sowohl Kanal A als auch Kanal B sind klar definiert und im Vergleich zum Rauschen relativ groß. Man erkennt jedoch bereits beim einfachen Betrachten, dass sich beide Seiten nur sehr geringfügig ähnlich sehen.

Beschreibung der KKF Die Kreuzkorrelationsfunktion schwankt sehr stark über den gesamtem Zeitbereich. Deswegen ist auch die Hüllkurve stark schwankend. Da die Regression über die Hüllkurve berechnet wird, wird diese dem Signal auch nicht gerecht.

Auswertung der Maßzahlen

6.1.2 Signal 2 – trefftz_fahrstuhl_m

Aufnahemsituation Diese Aufnahme fand im Fahrstuhl des Trefftzbaus statt. Das heißt, der Raum war relativ klein und ist mit schallharten Begrenzungen versehen. Um ein Signal zu erhalten wurde eine Primärschallquelle in Form eines hochwertigen Lautsprechers genutzt.

Signalbeschreibung In Abbildung 4 erkennt man sehr gut, dass sich beide Kanäle sehr schnell ändern und einen ähnlichen Verlauf haben. Lediglich die Stärke des Signals ist unterschiedliche. Das stört jedoch nicht für die Berechnung der Maßzahlen.

Beschreibung der KKF Für dieses Signal ist auch an der Kreuzkorrelationsfunktion klar zu sehen, dass es ein Maximum in der Mitte gibt. Das heißt, die Signale sind sich sehr ähnlich. Zu den Seiten nimmt die KKF langsam ab. Für solch einen Verlauf ist die Aussage der Hüllkurve und der Regression sehr gut, da dieses Kreuzkorrelation gut mit einer Gauß-Kurve approxmiert werden kann.

Auswertung der Maßzahlen

6.2 Probleme bei der Signalauswahl

Bei der Signalauswahl ergab sich das Problem, dass man möglichst viele verschiedene Raumsituationen erfassen musste, um möglichst viele verschiedene Daten zu bekommen. Dabei war es jedoch nur schwer möglich vor Ort zu entscheiden, ob die entsprechende Aufnahme sinnvolle Ergebnisse liefert. In den meisten Situationen war der Lautstärkepegel im Raum zu gering um 20s aufzunehmen, ohne das der Großteil der Aufnahme einfach Rauschen war. Aus diesem Grund haben wir ein zusätzliches Signal erzeugt. Dadurch gibt es jedoch in den meisten Aufnahmen eine Primärquelle, die das Spektrum

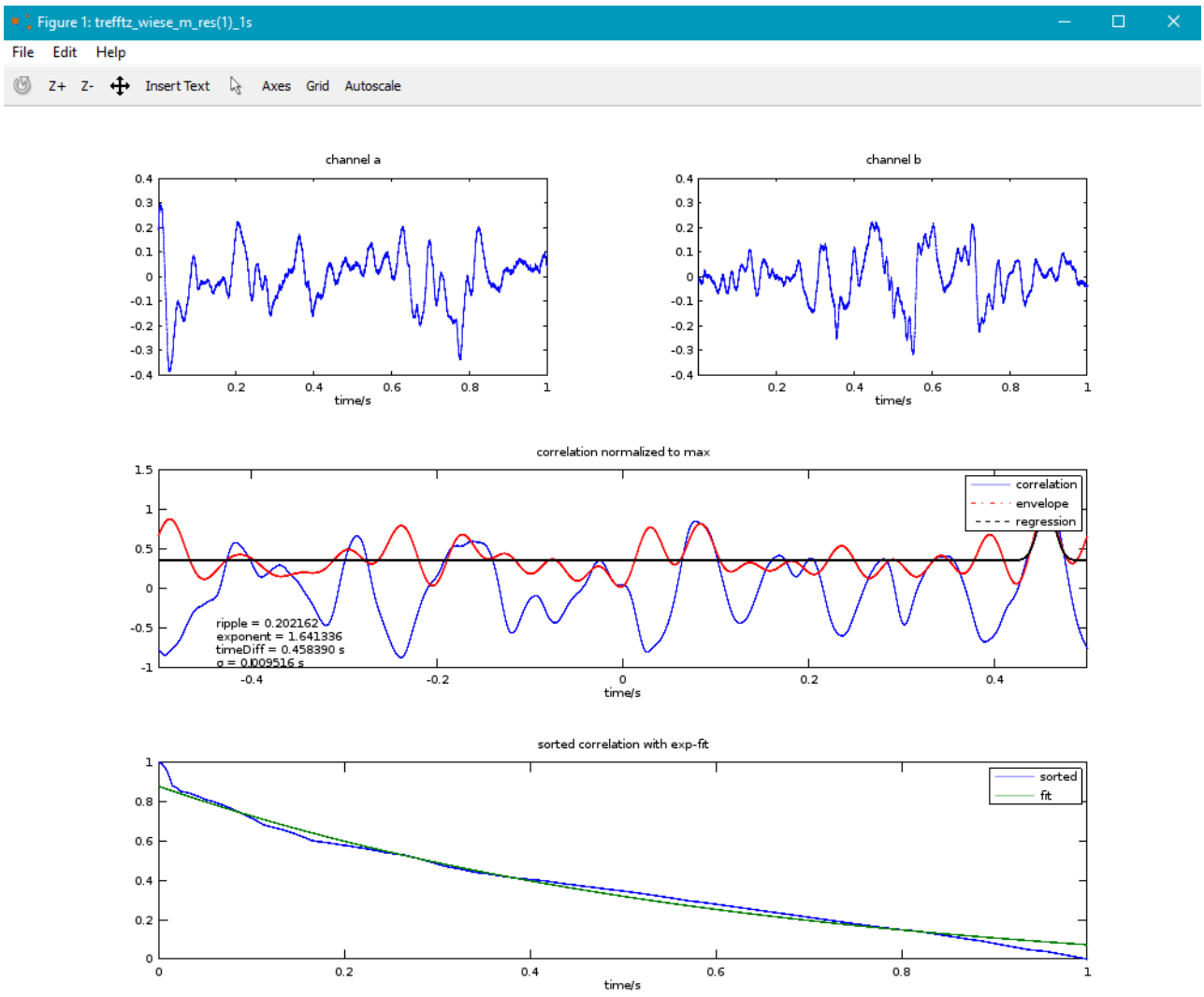

Abbildung 3: Signal 1

6.3 Fazit

Abschließend lässt sich feststellen, dass der Gauß-Fit erst aber einem ripple-Faktor von 0.3 sinnvolle Ergebnisse liefert. Bei Werten kleiner 0.3 ist die Hüllkurve einer Gauß-Kurve zu unähnlich. Es ist jedoch festzustellen, dass die Regression der Exponentialfunktion mit kleiner werdendem ripple besser über der nach Größe sortierten Amplituden liegt. Es ist empfehlenswert, Signale in Räumen aufzunehmen, die ausreichend klein sind, damit die Raumeffekte Auswirkungen auf das Signal haben. Sonst nimmt man größtenteils rauschen auf, welches sehr geringe Aussagen zulässt.

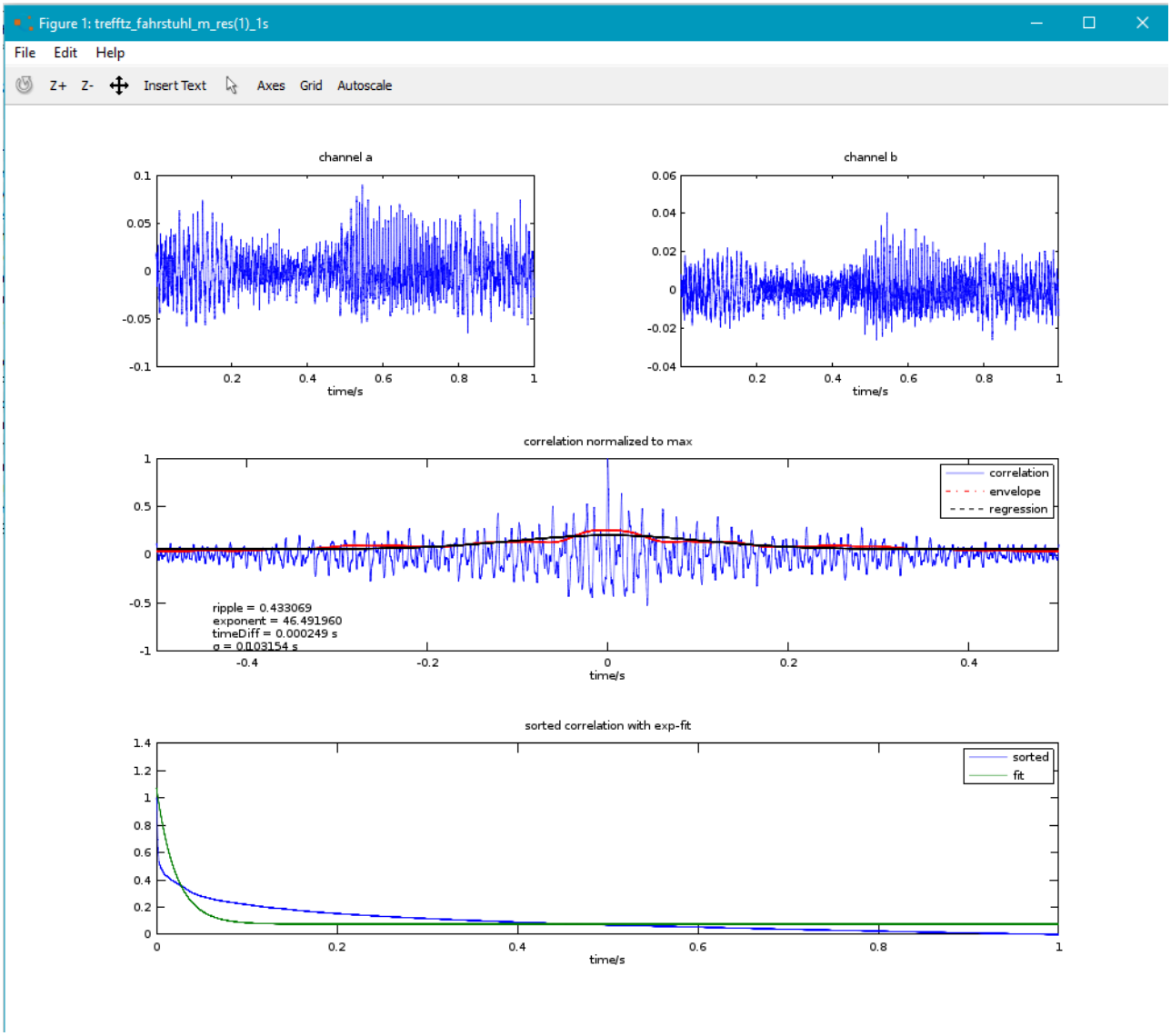

Abbildung 4: Signal 2

7 Zusammenfassung

Es wurde ein Ocatve-Skript entwickelt mit welchem sich die Kreuzkorrelationsfunktion der beiden Stereo-Kanäle einer Audioaufnahme berechnen lässt. Auf Basis der KKF wurden einige einfache Maßzahlen zur Charakterisierung der Aufnahmen entwickelt. Damit lassen sich für bestimmte Anwendungen Signale zu Testzwecken auswählen. In Zukunft kann die Software auf bestimmte Anwendungsfälle angepasst werden, in dem neue Regressionsmodelle implementiert werden, die der gewünschen Nutzung der Signale besser gerecht werden.

Literatur

[ISV] Rüdiger Hoffmann, Matthias Wolff: Intelligente Signalverarbeitung 1. Springer Verlag Berlin Heidelberg 2014

[NT] Prof. Dr.-Ing. Dr. h.c. Gerhard P. Fettweis: Einführung in die Nachrichtentechnik. Technische Universität Dresden, Fakultät Elektrotechnik, Vodafone Stiftungslehrstuhl Mobile Nachrichtensysteme, D-01062 Dresden Sommersemester 2015