

# Heterogeneous Computing

## Portfolio

Aufgabensammlung

Universität Trier  
FB IV - Informatikwissenschaften  
Professur für Systemsoftware und Verteilte Systeme

Semester: SoSe 2024

Name und Matrikelnummer:

Simon Szulik, 1474315

# Inhaltsverzeichnis

<b>1</b>	<b>Fourieranalyse der geheimnisvollen Wellenlängen</b>	<b>1</b>
1.1	Implementierung . . . . .	1
1.1.1	Ergebnisse . . . . .	1
1.2	Speicherplatzbedarf . . . . .	3
1.3	Variationen . . . . .	3
<b>2</b>	<b>Nächste Übung</b>	<b>4</b>
2.1	Aufgabe x . . . . .	4
<b>3</b>	<b>Nächste Übung</b>	<b>5</b>
3.1	Aufgabe x . . . . .	5
	<b>Literaturverzeichnis</b>	<b>6</b>

# 1. Fourieranalyse der geheimnisvollen Wellenlängen

Die Fourieranalyse ist eine mathematische Methode, die es ermöglicht, komplexe periodische Funktionen in eine Reihe von einfachen sinusförmigen Wellen zu zerlegen. Sie wird häufig in der Signalverarbeitung und der Analyse periodischer Phänomene wie Schwingungen und Wellen angewendet, um deren Bestandteile und Frequenzanteile zu untersuchen. Durch die Anwendung der Fourieranalyse kann man komplexe Signale in ihre Grundbestandteile zerlegen und dadurch besser verstehen.

## 1.1 Implementierung

Für die Implementierung wurde Python verwendet, sowie die folgenden Bibliotheken:

- **scipy**: Zum Einlesen der Audio-Datei (.wav) & dem Anwendungen einer schnellen Fourier-Analyse.
- **numpy**: Für numerische Operationen und die Berechnung der einzelnen Frequenzen.
- **matplotlib**: Als visuelle Darstellungsmöglichkeit der Daten und zum Erstellen von Plots.
- **argparse**: Zum Parsen der Kommandozeilenargumente, um die Eingabeparameter wie Dateiname, Blockgröße und Ausgabetyt zu bestimmen.

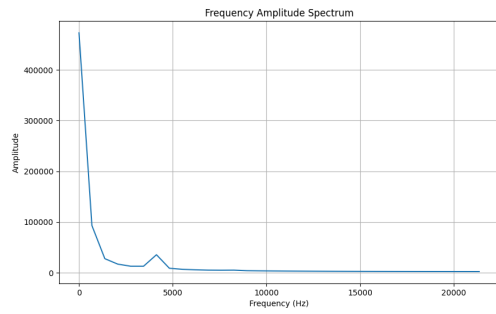
Im Beispiel 1.1 ist eine typische Ausführung des Skriptes zu sehen. Mit Hilfe der Argumente lässt sich sowohl die Blockgröße, als auch die Art des Diagramms einfach bestimmen.

```
1 py.exe Fourier.py audio.wav -b 512 -o f -p 1
```

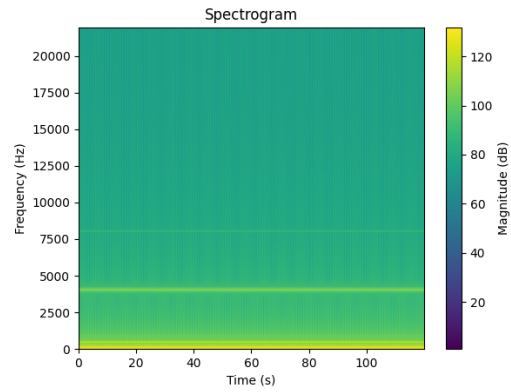
Beispiel 1.1: Skriptausführung in der Knsole

### 1.1.1 Ergebnisse

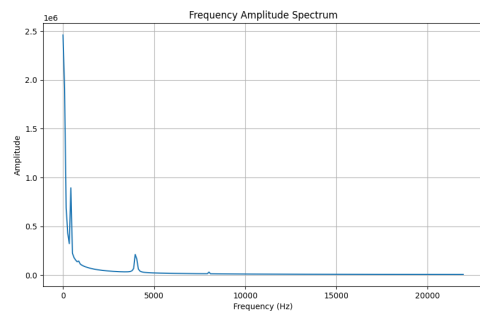
In diesem Abschnitt betrachten wir ein paar Beispiel Plots der Analyse. Bezüglich der Blockgröße gibt es einige Charakteristika die zu beachten sind. Behandelt man Signale mit schnellen und häufigen Änderungen wie Sprachaufnahmen, so können kleinere Blöcke (128-512) besser sein. Signale mit stabileren Frequenzen können problemlos mit größeren Blöcken analysiert werden. Um eine möglichst breite spanne von Fällen abzudecken, befindet sich in Abbildung 1.1 eine ansammlung von verschiedenen Plots der zu analysierenden Datei.



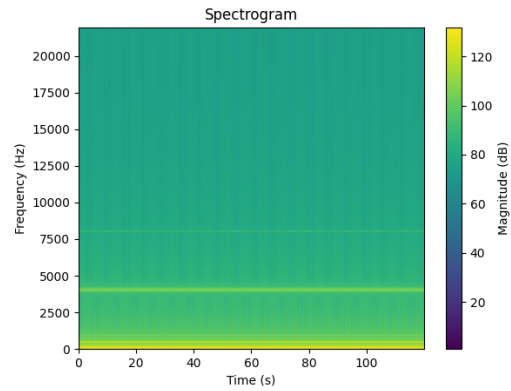
(a) Frequenz-Plot mit Blockgröße 64



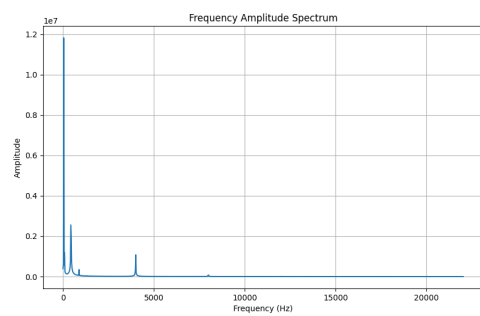
(b) Spektrogramm mit Blockgröße 64



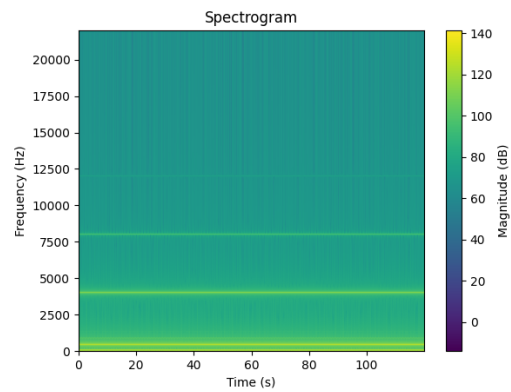
(c) Frequenz-Plot mit Blockgröße 512



(d) Spektrogramm mit Blockgröße 512



(e) Frequenz-Plot mit Blockgröße 2048



(f) Spektrogramm mit Blockgröße 2048

Abbildung 1.1: Beispielergebnisse

## **1.2 Speicherplatzbedarf**

hola

## **1.3 Variationen**

hola

## **2. Nächste Übung**

### **2.1 Aufgabe x**

## **3. Nächste Übung**

### **3.1 Aufgabe x**

# Literaturverzeichnis