

Aufgabe 4: MFCCs

In dieser Übung soll die Merkmalsextraktion um die Berechnung der Mel-Frequenz Cepstral-koeffizienten (MFCCs) und der ersten und zweiten Zeitableitung erweitert werden.

Allgemeiner Hinweis: Bitte beachten Sie bei allen Berechnungen innerhalb der Merkmalsextraktion, dass die erste Dimension der Matrix, welche die extrahierten Merkmale enthält, immer der Anzahl der Rahmen entspricht. Die zweite Dimension soll dementsprechend die Dimension der jeweiligen Merkmale beschreiben.

Erweiterung der Merkmalsextraktion

4.1 Implementieren Sie eine Funktion

```
def compute_cepstrum(mel_spectrum, num_ceps):  
    ...
```

in der Datei `feature_extraction.py`, die das reelle Cepstrum aus einem gegebenen Mel-Spektrum `mel_spectrum` bzw. \mathbf{x}_{MEL} (aus Aufgabe 3.4) nach der Vorschrift

$$\mathbf{x}_{\text{CEP}}[\tau] = \text{DCT}\left(\log \left\{ \left| \mathbf{x}_{\text{MEL}}(\tau) \right| \right\}\right)$$

berechnet. Hierbei bezeichnen τ den Rahmenindex. Um numerische Probleme zu vermeiden, sollten Nullen im Mel-Spektrum durch den kleinsten darstellbaren Wert (Maschinenepsilon-Wert von NumPy) ersetzt werden. Beachten Sie, dass die Funktion lediglich die ersten `num_ceps` Koeffizienten zurückgeben soll.

Hilfreiche Funktionen: `scipy.fftpack.dct()` (bitte setzen Sie beim Aufruf dieser Funktion den optionalen Parameter `norm` auf `'ortho'`)

4.2 Implementieren Sie eine Funktion

```
def get_delta(x):  
    ...
```

in der Datei `feature_extraction.py`, die von Cepstrum \mathbf{x}_{CEP} (wie in Aufgabe 4.1 berechnet) die erste zeitliche Ableitung gemäß

$$\Delta \mathbf{x}_{\text{CEP}}[\tau] = \frac{1}{2} \left(\mathbf{x}_{\text{CEP}}[\tau + 1] - \mathbf{x}_{\text{CEP}}[\tau - 1] \right)$$

berechnet und zurückgibt. An den Rändern des Spektrums für $\tau = 0$ und $\tau = T - 1$ (wobei T die Anzahl der Rahmen bezeichnet), können hierbei die Beziehungen

$$\Delta \mathbf{x}_{\text{CEP}}[0] = \mathbf{x}_{\text{CEP}}[1] - \mathbf{x}_{\text{CEP}}[0]$$

und

$$\Delta \mathbf{x}_{\text{CEP}}[T - 1] = \mathbf{x}_{\text{CEP}}[T - 1] - \mathbf{x}_{\text{CEP}}[T - 2]$$

verwendet werden.

4.3 Implementieren Sie eine Funktion

```
def append_delta(x, delta):  
    ...
```

in der Datei `feature_extraction.py`, die einen Merkmalsvektor `x` mit dessen erste Ableitung `delta` konkateniert, so dass in jedem zeitlichen Rahmen ein gemeinsamer Merkmalsvektor mit der Dimension $2N_{\text{ceps}}$ entsteht.

4.4 Erweitern Sie die Funktion zur Merkmalsextraktion `compute_features()` in der Datei `feature_extraction.py`, so dass diese für ein Audiosignal in Abhängigkeit vom Argument `feature_type` folgende Rückgabewerte liefert:

- Die MFCCs, wenn für `feature_type` der Wert `MFCC` angegeben wird,
- die MFCCs und dessen erste zeitliche Ableitung, wenn für das Argument `feature_type` der Wert `MFCC_D` angegeben wird, sowie
- die MFCCs und dessen erste und zweite zeitliche Ableitung (konkateniert als $3N_{\text{ceps}}$ -dimensionale Merkmalsvektoren), wenn für `feature_type` der Wert `MFCC_DD` angegeben wird.

Darstellung der extrahierten Merkmale

4.5 Schreiben Sie ein Skript in der Datei `uebung4.py`, welches mit Hilfe der zuvor implementierten Funktionen die MFCCs sowie deren erste und zweite zeitliche Ableitung für die Audiodatei `TEST-MAN-AH-3033951A.wav` grafisch darstellt. Verwenden Sie hierbei die Standardeinstellungen der Funktion `compute_features()`. Abb. 1 zeigt eine Darstellung des gewünschten Ergebnisses.

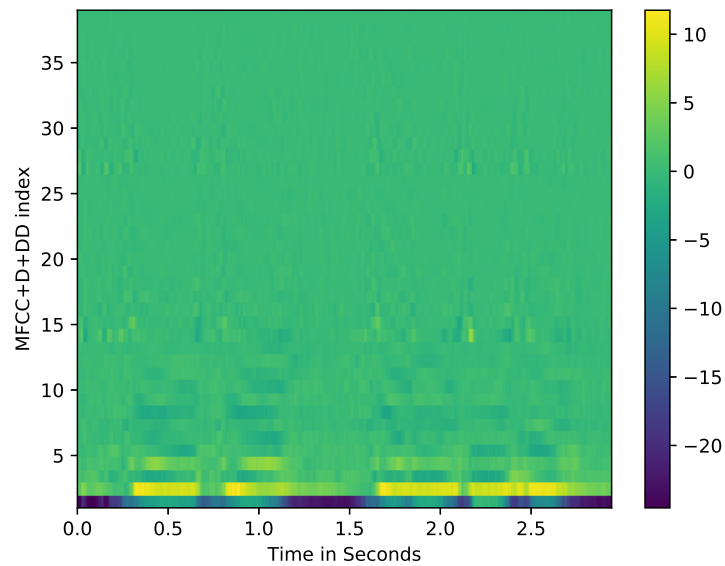


Figure 1: Darstellung der aus der Audiodatei TEST-MAN-AH-3033951A.wav extrahierten Merkmale für die Standardeinstellungen der Funktion `compute_features()`, wenn für `feature_type` der Wert `MFCC_D_DD` angegeben wird.