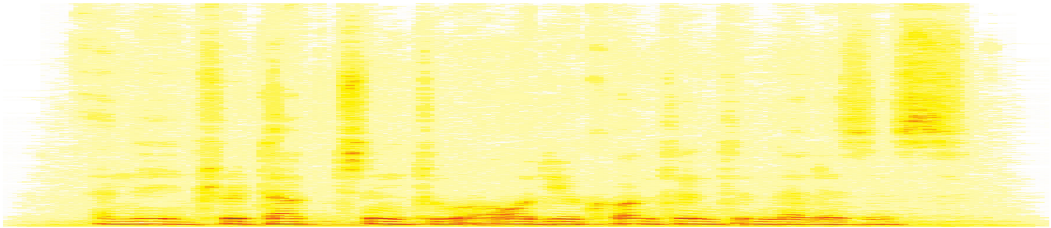
Introduction to Audio Content Analysis

Module 3.0: Feature Extraction — Introduction and Pre-processing

alexander lerch



Georgia Tech Center for Music Technology
College of Design

introduction

overview

Georgia Tech Center for Music Technology
College of Design

corresponding textbook section

Chapter 3 — Instantaneous Features: pp. 31–35

- **lecture content**
  - introduction to the concept of features
  - audio pre-processing for feature extraction
- **learning objectives**
  - describe the process of feature extraction
  - list possible pre-processing option and explain potential use cases

introduction

overview

Georgia Tech Center for Music Technology
College of Design

corresponding textbook section

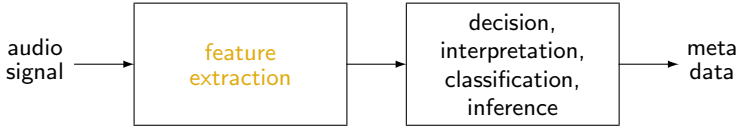
Chapter 3 — Instantaneous Features: pp. 31–35

- **lecture content**
  - introduction to the concept of features
  - audio pre-processing for feature extraction
- **learning objectives**
  - describe the process of feature extraction
  - list possible pre-processing option and explain potential use cases

instantaneous features

introduction

remember the flow chart of a general ACA system:



feature:

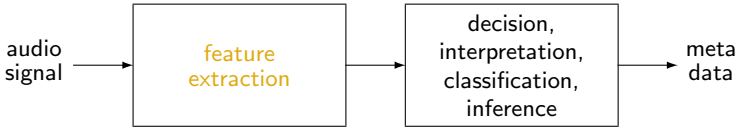
- *terminology*:
  - audio descriptor
  - instantaneous/short-term/low-level feature
- *characteristics*:
  - not necessarily musically, perceptually, or semantically meaningful
  - low-level: usually one value per block

instantaneous features

introduction

Georgia Tech Center for Music Technology College of Design

remember the flow chart of a general ACA system:



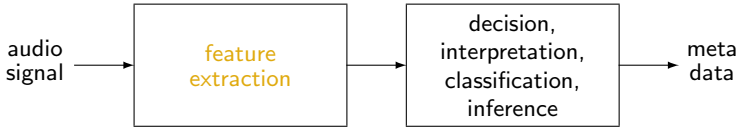
feature:

- *terminology*:
  - audio descriptor
  - instantaneous/short-term/low-level feature
- *characteristics*:
  - not necessarily musically, perceptually, or semantically meaningful
  - low-level: usually one value per block

instantaneous features

introduction

remember the flow chart of a general ACA system:



feature:

- *terminology*:
  - audio descriptor
  - instantaneous/short-term/low-level feature
- *characteristics*:
  - not necessarily musically, perceptually, or semantically meaningful
  - low-level: usually one value per block

instantaneous features

feature

Georgia Tech Center for Music Technology
College of Design

a feature . . .

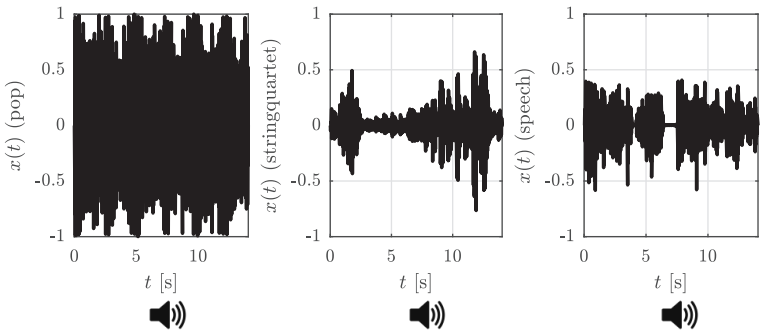
- is task-specific, i.e. contains information relevant to the task,
- may be custom-designed, chosen from a set of established features, or learned from data,
- can be a representation of any data (audio, meta data, other features, ...),
- is not necessarily musically, perceptually, or semantically meaningful or interpretable

instantaneous features

feature example

Georgia Tech Center for Music Technology College of Design

waveform envelope of three different signals



matlab source: matlab/displayWaveforms.m

- envelopes of waveforms can have distinct shape

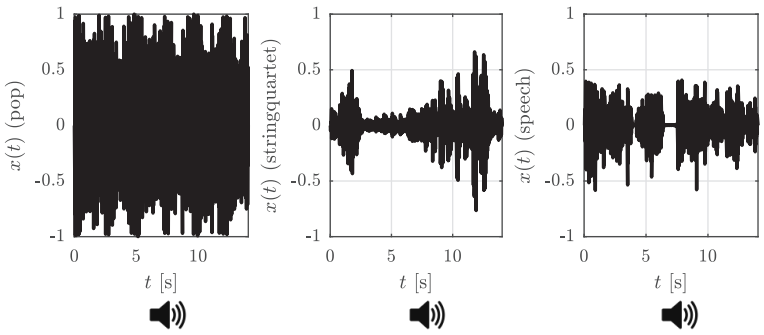
$\Rightarrow$ a feature describing envelope shape could help to distinguish these signal types

instantaneous features

feature example

Georgia Tech Center for Music Technology College of Design

waveform envelope of three different signals



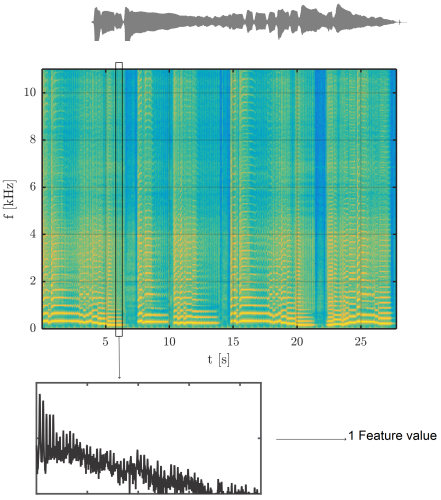
- envelopes of waveforms can have distinct shape

⇒ a feature describing envelope shape could help to distinguish these signal types

matlab source: matlab/displayWaveforms.m

instantaneous features

feature extraction



- repeat for every block
- repeat for every feature: *Spectral Centroid*, *RMS*, *MFCCs*, ...

⇒ feature matrix per audio input

instantaneous features

audio pre-processing

- **pre-processing**: audio is treated before feature extraction (task dependent)
- **possible goals**
  - *reduce amount of data* (e.g., down-sampling)
  - *remove irrelevant information* (e.g., surround channels of multi-channel signal)
  - *remove information that might impact analysis* (e.g., DC offset)
  - *increase robustness* (e.g., normalization)

instantaneous features

audio pre-processing examples 1/2

- **down-mixing**

$$x(i) = \frac{1}{C} \sum_{c=0}^{C-1} x_c(i)$$

  - *variants*: different channel weights, $\pi/2$ phase shift in one channel, ...

- **normalization**

$$x(i) = \frac{x_s(i)}{\max_{\forall i} \left(|x_s(i)| \right)}$$

  - *variants*: RMS, LUFS normalization
  - real-time?

instantaneous features

audio pre-processing examples 1/2

- **down-mixing**

$$x(i) = \frac{1}{C} \sum_{c=0}^{C-1} x_c(i)$$

  - *variants*: different channel weights, $\pi/2$ phase shift in one channel, ...

- **normalization**

$$x(i) = \frac{x_s(i)}{\max_{\forall i} \left(|x_s(i)| \right)}$$

  - *variants*: RMS, LUFS normalization
  - real-time?

instantaneous features

audio pre-processing examples 2/2

Georgia Tech Center for Music Technology College of Design

- **filtering**
  - *DC removal*

$$x(i) = x_{\mathrm{DC}}(i) - \frac{1}{\mathcal{I}} \sum_{i=0}^{\mathcal{I}-1} x_{\mathrm{DC}}(i)$$

  - other filters: high/band pass, smoothing, ...
- **sample rate reduction**
- **quality enhancement** (denoising, etc.)
- ...

instantaneous features

audio pre-processing examples 2/2

- **filtering**
  - *DC removal*

$$x(i) = x_{\mathrm{DC}}(i) - \frac{1}{\mathcal{I}} \sum_{i=0}^{\mathcal{I}-1} x_{\mathrm{DC}}(i)$$

  - other filters: high/band pass, smoothing, ...
- **sample rate reduction**
- **quality enhancement** (denoising, etc.)
- ...

instantaneous features

audio pre-processing examples 2/2

- **filtering**
  - *DC removal*

$$x(i) = x_{\mathrm{DC}}(i) - \frac{1}{\mathcal{I}} \sum_{i=0}^{\mathcal{I}-1} x_{\mathrm{DC}}(i)$$

  - other filters: high/band pass, smoothing, ...
- **sample rate reduction**
- **quality enhancement** (denoising, etc.)
- ...

summary

lecture content

Georgia Tech Center for Music Technology
College of Design

- **feature**
  - descriptor with condensed relevant information
  - not necessarily interpretable by humans
- **low-level feature extraction**
  - usually extracted per short block of samples
  - many features can be extracted from audio data, resulting in feature matrix
- **pre-processing**
  - remove irrelevant data,
  - clean relevant data