



# Introduction to **Audio Content Analysis**

module 7.3.4: fundamental frequency detection in polyphonic signals

alexander lerch

# introduction

## overview

### corresponding textbook section

#### section 7.3.4

#### ■ lecture content

- overview of “historic” methods for polyphonic pitch detection
- introduction to Non-negative Matrix Factorization (NMF)

#### ■ learning objectives

- describe the task and challenges of polyphonic pitch detection
- list the processing steps of iterative subtraction and relate them to the introduced approaches
- describe the process of NMF and discuss advantages and disadvantages of using NMF for pitch detection



# introduction

## overview

### corresponding textbook section

#### section 7.3.4

#### ■ lecture content

- overview of “historic” methods for polyphonic pitch detection
- introduction to Non-negative Matrix Factorization (NMF)

#### ■ learning objectives

- describe the task and challenges of polyphonic pitch detection
- list the processing steps of iterative subtraction and relate them to the introduced approaches
- describe the process of NMF and discuss advantages and disadvantages of using NMF for pitch detection



# polyphonic pitch tracking

## problem statement

- **monophonic** fundamental frequency detection:
  - exactly one fundamental frequency with sinusoidals at multiples of  $f_0$  (harmonics)
  
- **polyphonic** fundamental frequency detection:
  - multiple/unknown number of fundamental frequencies with harmonics
  - number of voices might change over time
  - complex mixture with overlapping frequency content

# polyphonic pitch tracking

## iterative subtraction: introduction

### ■ principle

- 1 find most salient fundamental frequency
  - ▶ e.g., with monophonic pitch tracking
- 2 remove this frequency and related frequency components
  - ▶ e.g., mask or subtraction
- 3 repeat until termination criterion
  - ▶ e.g., number of voices

### ■ challenges

- reliably *identify fundamental frequency* in a mixture
- *identify/group components* and amount to subtract
  - ▶ overlapping components
  - ▶ spectral leakage
- define *termination criterion*
  - ▶ e.g., unknown number of voices or overall energy

# polyphonic pitch tracking

## iterative subtraction: introduction

### ■ principle

- 1 find most salient fundamental frequency
  - ▶ e.g., with monophonic pitch tracking
- 2 remove this frequency and related frequency components
  - ▶ e.g., mask or subtraction
- 3 repeat until termination criterion
  - ▶ e.g., number of voices

### ■ challenges

- reliably *identify fundamental frequency* in a mixture
- *identify/group components* and amount to subtract
  - ▶ overlapping components
  - ▶ spectral leakage
- define *termination criterion*
  - ▶ e.g., unknown number of voices or overall energy

# polyphonic pitch tracking

## iterative subtraction: Cheveigné

### 1 compute squared AMDF

$$\text{ASMDF}_{xx}(\eta, n) = \frac{1}{i_e(n) - i_s(n) + 1} \sum_{i=i_s(n)}^{i_e(n)} (x(i) - x(i + \eta))^2$$

### 2 find fundamental frequency

$$\eta_{\min} = \operatorname{argmin} (\text{ASMDF}_{xx}(\eta, n))$$

### 3 apply comb cancellation filter, IR:

$$h(i) = \delta(i) - \delta(i - \eta_{\min})$$

### 4 repeat process

# polyphonic pitch tracking

## iterative subtraction: Cheveigné

### 1 compute squared AMDF

$$\text{ASMDF}_{xx}(\eta, n) = \frac{1}{i_e(n) - i_s(n) + 1} \sum_{i=i_s(n)}^{i_e(n)} (x(i) - x(i + \eta))^2$$

### 2 find fundamental frequency

$$\eta_{\min} = \operatorname{argmin} (\text{ASMDF}_{xx}(\eta, n))$$

### 3 apply comb cancellation filter, IR:

$$h(i) = \delta(i) - \delta(i - \eta_{\min})$$

### 4 repeat process



# polyphonic pitch tracking

## iterative subtraction: Cheveigné

### 1 compute squared AMDF

$$\text{ASMDF}_{xx}(\eta, n) = \frac{1}{i_e(n) - i_s(n) + 1} \sum_{i=i_s(n)}^{i_e(n)} (x(i) - x(i + \eta))^2$$

### 2 find fundamental frequency

$$\eta_{\min} = \operatorname{argmin} (\text{ASMDF}_{xx}(\eta, n))$$

### 3 apply comb cancellation filter, IR:

$$h(i) = \delta(i) - \delta(i - \eta_{\min})$$

### 4 repeat process

# polyphonic pitch tracking

## iterative subtraction: Cheveigné

### 1 compute squared AMDF

$$\text{ASMDF}_{xx}(\eta, n) = \frac{1}{i_e(n) - i_s(n) + 1} \sum_{i=i_s(n)}^{i_e(n)} (x(i) - x(i + \eta))^2$$

### 2 find fundamental frequency

$$\eta_{\min} = \operatorname{argmin} (\text{ASMDF}_{xx}(\eta, n))$$

### 3 apply comb cancellation filter, IR:

$$h(i) = \delta(i) - \delta(i - \eta_{\min})$$

### 4 repeat process

# polyphonic pitch tracking

## iterative subtraction: Meddis

### 1 auditory pitch tracking:

$$r_{zz}(c, n, \eta) = \sum_{i=0}^{\mathcal{K}-1} z_c(i) \cdot z_c(i + \eta)$$

- 2 detect most likely frequency for all bands
- 3 remove all bands with a max at detected frequency
- 4 reiterate until most bands have eliminated

# polyphonic pitch tracking

## iterative subtraction: Meddis

### 1 auditory pitch tracking:

$$r_{zz}(c, n, \eta) = \sum_{i=0}^{\mathcal{K}-1} z_c(i) \cdot z_c(i + \eta)$$

### 2 detect most likely frequency for all bands

### 3 remove all bands with a max at detected frequency

### 4 reiterate until most bands have eliminated

# polyphonic pitch tracking

## iterative subtraction: Meddis

### 1 auditory pitch tracking:

$$r_{zz}(c, n, \eta) = \sum_{i=0}^{\mathcal{K}-1} z_c(i) \cdot z_c(i + \eta)$$

### 2 detect most likely frequency for all bands

### 3 remove all bands with a max at detected frequency

### 4 reiterate until most bands have eliminated

# polyphonic pitch tracking

## iterative subtraction: Meddis

### 1 auditory pitch tracking:

$$r_{zz}(c, n, \eta) = \sum_{i=0}^{\mathcal{K}-1} z_c(i) \cdot z_c(i + \eta)$$

### 2 detect most likely frequency for all bands

### 3 remove all bands with a max at detected frequency

### 4 reiterate until most bands have eliminated

# polyphonic pitch tracking

## iterative subtraction: spectral

- 1 find salient fundamental frequency (e.g. auditory approach, HPS)
- 2 estimate current model for harmonic magnitudes
- 3 subtract the model spectrum
- 4 repeat process

# polyphonic pitch tracking

## iterative subtraction: spectral

- 1 find salient fundamental frequency (e.g. auditory approach, HPS)
- 2 estimate current model for harmonic magnitudes
- 3 subtract the model spectrum
- 4 repeat process



# polyphonic pitch tracking

## iterative subtraction: spectral

- 1 find salient fundamental frequency (e.g. auditory approach, HPS)
- 2 estimate current model for harmonic magnitudes
- 3 subtract the model spectrum
- 4 repeat process

# polyphonic pitch tracking

## iterative subtraction: spectral

- 1 find salient fundamental frequency (e.g. auditory approach, HPS)
- 2 estimate current model for harmonic magnitudes
- 3 subtract the model spectrum
- 4 repeat process

# polyphonic pitch tracking

## exhaustive search

- 1 define set of all possible fundamental frequencies
- 2 compute all possible pairs of fundamental frequency
- 3 repeatedly filter the signal with two comb cancellation filters (all combinations)
- 4 find combination with minimal output energy

# polyphonic pitch tracking

## exhaustive search

- 1 define set of all possible fundamental frequencies
- 2 compute all possible pairs of fundamental frequency
- 3 repeatedly filter the signal with two comb cancellation filters (all combinations)
- 4 find combination with minimal output energy

# polyphonic pitch tracking

## exhaustive search

- 1 define set of all possible fundamental frequencies
- 2 compute all possible pairs of fundamental frequency
- 3 repeatedly filter the signal with two comb cancellation filters (all combinations)
- 4 find combination with minimal output energy

# polyphonic pitch tracking

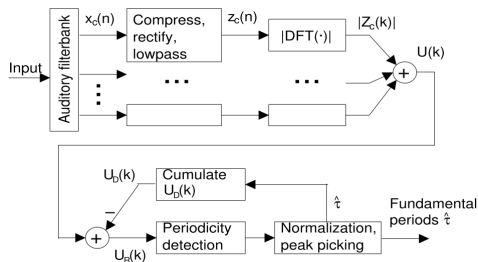
## exhaustive search

- 1 define set of all possible fundamental frequencies
- 2 compute all possible pairs of fundamental frequency
- 3 repeatedly filter the signal with two comb cancellation filters (all combinations)
- 4 find combination with minimal output energy

# polyphonic pitch tracking

klapuri

- 1 gammatone **filterbank** (100 bands)
- 2 **normalization**, HWR, smoothing, ...
- 3 **STFT** per filter channel (magnitude)
- 4 use **delta pulse templates** to detect frequency patterns
- 5 **pick most salient frequencies**, remove them



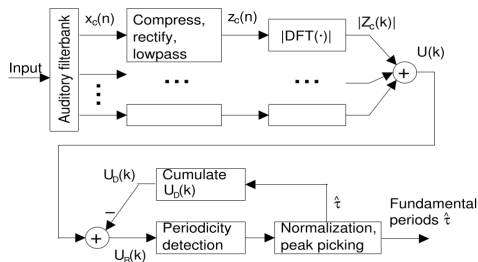
graph from<sup>1</sup>

<sup>1</sup>A. P. Klapuri, "A Perceptually Motivated Multiple-F0 Estimation Method," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, 2005.

# polyphonic pitch tracking

klapuri

- 1 gammatone **filterbank** (100 bands)
- 2 **normalization**, HWR, smoothing, ...
- 3 STFT per filter channel (magnitude)
- 4 use **delta pulse templates** to detect frequency patterns
- 5 **pick most salient frequencies**, remove them



graph from<sup>1</sup>

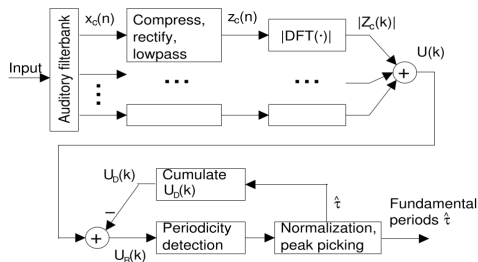
<sup>1</sup>A. P. Klapuri, "A Perceptually Motivated Multiple-F0 Estimation Method," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, 2005.



# polyphonic pitch tracking

klapuri

- 1 gammatone **filterbank** (100 bands)
- 2 **normalization**, HWR, smoothing, ...
- 3 **STFT** per filter channel (magnitude)
- 4 use **delta pulse templates** to detect frequency patterns
- 5 **pick most salient frequencies**, remove them



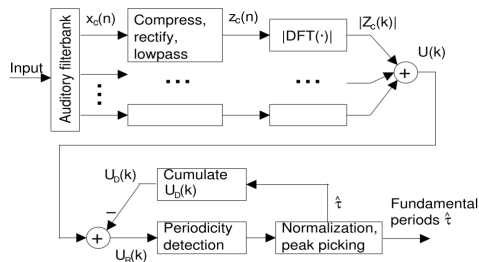
graph from<sup>1</sup>

<sup>1</sup>A. P. Klapuri, "A Perceptually Motivated Multiple-F0 Estimation Method," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, 2005.

# polyphonic pitch tracking

## klapuri

- 1 gammatone **filterbank** (100 bands)
- 2 **normalization**, HWR, smoothing, ...
- 3 **STFT** per filter channel (magnitude)
- 4 use **delta pulse templates** to detect frequency patterns
- 5 **pick most salient frequencies**, remove them



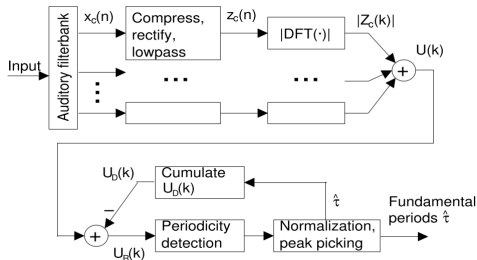
graph from<sup>1</sup>

<sup>1</sup>A. P. Klapuri, "A Perceptually Motivated Multiple-F0 Estimation Method," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, 2005.

# polyphonic pitch tracking

klapuri

- 1 gammatone **filterbank** (100 bands)
- 2 **normalization**, HWR, smoothing, ...
- 3 **STFT** per filter channel (magnitude)
- 4 use **delta pulse templates** to detect frequency patterns
- 5 **pick most salient frequencies**, remove them



graph from<sup>1</sup>

<sup>1</sup>A. P. Klapuri, "A Perceptually Motivated Multiple-F0 Estimation Method," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, 2005.

# non-negative matrix factorization

## introduction

### ■ Non-negative Matrix Factorization (NMF)

Given a  $m \times n$  matrix  $V$ , find a  $m \times r$  matrix  $W$  and a  $r \times n$  matrix  $H$  such that

$$V \approx WH$$

- all matrices must be non-negative
- rank  $r$  is usually smaller than  $m$  and  $n$

### ■ advantage of non-negativity?

- additive model
- relates to probability distributions
- efficiency?

# non-negative matrix factorization

## introduction

### ■ Non-negative Matrix Factorization (NMF)

Given a  $m \times n$  matrix  $V$ , find a  $m \times r$  matrix  $W$  and a  $r \times n$  matrix  $H$  such that

$$V \approx WH$$

- all matrices must be non-negative
- rank  $r$  is usually smaller than  $m$  and  $n$

### ■ advantage of non-negativity?

- additive model
- relates to probability distributions
- efficiency?

# non-negative matrix factorization

## introduction

### ■ Non-negative Matrix Factorization (NMF)

Given a  $m \times n$  matrix  $V$ , find a  $m \times r$  matrix  $W$  and a  $r \times n$  matrix  $H$  such that

$$V \approx WH$$

- all matrices must be non-negative
- rank  $r$  is usually smaller than  $m$  and  $n$

### ■ advantage of non-negativity?

- additive model
- relates to probability distributions
- efficiency?

# non-negative matrix factorization

## introduction

### ■ Non-negative Matrix Factorization (NMF)

Given a  $m \times n$  matrix  $V$ , find a  $m \times r$  matrix  $W$  and a  $r \times n$  matrix  $H$  such that

$$V \approx WH$$

- all matrices must be non-negative
- rank  $r$  is usually smaller than  $m$  and  $n$

### ■ advantage of non-negativity?

- additive model
- relates to probability distributions
- efficiency?

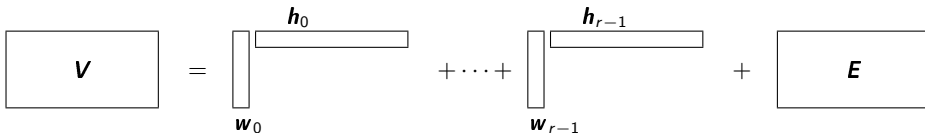
# non-negative matrix factorization

## overview

alternative formulation<sup>2</sup> to  $V \approx WH$

$$V = \sum_{i=1}^r w_i \cdot h_i + E$$

- $V \in \mathbb{R}^{m \times n}$
- $W = [w_1, w_2, \dots, w_r] \in \mathbb{R}^{m \times r}$
- $H = [h_1, h_2, \dots, h_r]^T \in \mathbb{R}^{r \times n}$
- $E$  is the error matrix



<sup>2</sup>A Cichocki, R Zdunek, A. Phan, et al., *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.



# objective function

## distance and divergence

- task: **iteratively minimize objective function**  $D(V || WH)$
- typical distance measures ( $B = WH$ ):
  - squared Euclidean distance:

$$D_{\text{EU}}(V || B) = \| V - B \|^2 = \sum_{ij} (V_{ij} - B_{ij})^2$$

- generalized K-L divergence:

$$D_{\text{KL}}(V || B) = \sum_{ij} \left( V_{ij} \log \left( \frac{V_{ij}}{B_{ij}} \right) - V_{ij} + B_{ij} \right)$$

- others<sup>3</sup>: Bregman Divergence, Alpha-Divergence, Beta-Divergence, ...

---

<sup>3</sup> A Cichocki, R Zdunek, A. Phan, *et al.*, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

# objective function

## distance and divergence

- task: **iteratively minimize objective function**  $D(V || WH)$
- typical distance measures ( $B = WH$ ):
  - squared Euclidean distance:

$$D_{\text{EU}}(V || B) = \| V - B \|^2 = \sum_{ij} (V_{ij} - B_{ij})^2$$

- generalized K-L divergence:

$$D_{\text{KL}}(V || B) = \sum_{ij} \left( V_{ij} \log \left( \frac{V_{ij}}{B_{ij}} \right) - V_{ij} + B_{ij} \right)$$

- others<sup>3</sup>: Bregman Divergence, Alpha-Divergence, Beta-Divergence, ...

---

<sup>3</sup>A Cichocki, R Zdunek, A. Phan, *et al.*, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

# objective function

## gradient descent

### ■ minimization of objective function

### ■ **gradient descent**: minimum can be found as zero of derivative

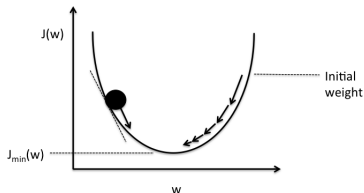
- 2D example: given a function  $f(x_1, x_2)$ , find the minimum  $x_1 = a$  and  $x_2 = b$

1 initialize  $x_i(0)$  with random numbers

2 update points iteratively:

$$x_i(n+1) = x_i(n) - \alpha \cdot \frac{\partial f}{\partial x_i}, \quad i = [1, 2]$$

⇒ as iteration number  $n$  increases,  $x_1, x_2$  will be closer to  $a, b$ .



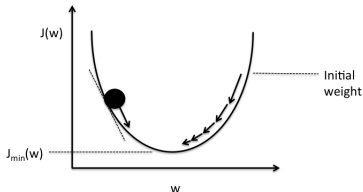
# objective function

## gradient descent

- minimization of objective function
- **gradient descent**: minimum can be found as zero of derivative
  - 2D example: given a function  $f(x_1, x_2)$ , find the minimum  $x_1 = a$  and  $x_2 = b$ 
    - 1 initialize  $x_i(0)$  with random numbers
    - 2 update points iteratively:

$$x_i(n+1) = x_i(n) - \alpha \cdot \frac{\partial f}{\partial x_i}, \quad i = [1, 2]$$

⇒ as iteration number  $n$  increases,  $x_1, x_2$  will be closer to  $a, b$ .



# objective function

## additive vs. multiplicative update rules

optimization of objective function<sup>4</sup>  $D_{\text{EU}}(V \parallel WH) = \|V - WH\|^2$

■ **additive** update rules:

$$H \leftarrow H + \alpha \frac{\partial J}{\partial H} = H + \alpha[(W^T V) - (W^T WH)]$$

$$W \leftarrow W + \alpha \frac{\partial J}{\partial W} = W + \alpha[(VH^T) - (WHH^T)]$$

■ **multiplicative** update rules:

$$H \leftarrow H \frac{(W^T V)}{(W^T WH)}$$

$$W \leftarrow W \frac{(VH^T)}{(WHH^T)}$$

<sup>4</sup>D Seung and L Lee, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001, pp. 556–562. [Online]. Available: <http://www.public.asu.edu/~jye02/CLASSES/Fall-2007/NOTES/lee01algorithms.pdf>.

# objective function

## additive vs. multiplicative update rules

optimization of objective function<sup>4</sup>  $D_{\text{EU}}(V \parallel WH) = \|V - WH\|^2$

■ **additive** update rules:

$$H \leftarrow H + \alpha \frac{\partial J}{\partial H} = H + \alpha[(W^T V) - (W^T WH)]$$

$$W \leftarrow W + \alpha \frac{\partial J}{\partial W} = W + \alpha[(VH^T) - (WHH^T)]$$

■ **multiplicative** update rules:

$$H \leftarrow H \frac{(W^T V)}{(W^T WH)}$$

$$W \leftarrow W \frac{(VH^T)}{(WHH^T)}$$

<sup>4</sup>D Seung and L Lee, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001,

pp. 556–562. [Online]. Available: <http://www.public.asu.edu/~jye02/CLASSES/Fall-2007/NOTES/lee01algorithms.pdf>.

# objective function

## additional cost function constraints

- additional penalty terms (regularization terms) may be added to objective function
- example: sparsity in  $W$  or  $H$

$$D = \| V - WH \|^2 + \alpha J_W(W) + \beta J_H(H)$$





- $\alpha, \beta$ : coefficients for controlling degree of sparsity
- $J_W$  and  $J_H$ : typically  $L_1, L_2$  norm

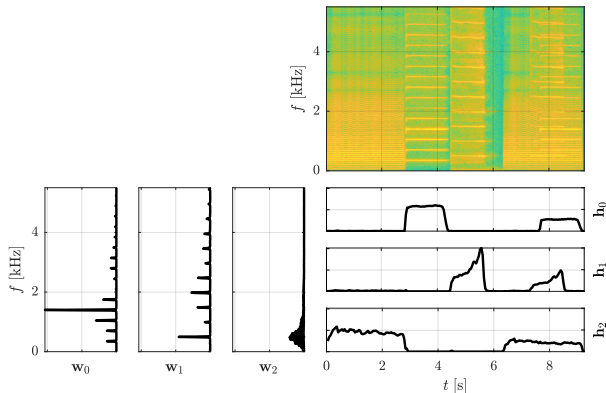
# nmf example

## template extraction

- unsupervised extraction of templates and activations

- input audio:

-  horn
-  oboe
-  violin
-  mix





# nmf use cases

## piano transcription

### ■ **separate template adaptation** from activation matrix adaptation:

- 1 train/set template matrix
- 2 order template matrix to have fixed pitch mapping
- 3 keep template matrix fixed and only update activation matrix
- 4 pick activation magnitude to determine active pitches

### ■ **potential problems**

- detuned piano/tuning frequency deviation
- template differs significantly from sound analyzed
- spectral changes dependent on time and/or loudness

# summary

## lecture content

### ■ polyphonic pitch detection

- highly challenging task with
  - ▶ unknown number of sources
  - ▶ unknown harmonic structure
  - ▶ spectral overlap of sources
  - ▶ time-varying mixture

### ■ traditional approaches

- iterative subtraction (detect one pitch, remove it, repeat analysis)
- multi-band processing

### ■ non-negative matrix factorization

- iterative process minimizing an objective function
- split a matrix into a template matrix and an activation matrix

