Synchronizing Audio Signals
-
via Landmark Cross Correlation

by Lea N. Possberg

Seminar: Speech Processing

November 30, 2017

Motivation

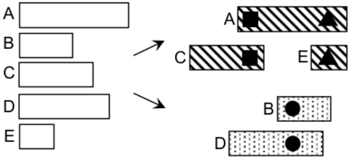


Figure: Clustering and Synchronization (Bryan, Smaragdis, Mysore, 2012)

Cross Correlation Function R

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

What does R do?

Video

(https://www.youtube.com/watch?v=L6YJqhbsuFY)

Cross Correlation Function R - Landmark Signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

Cross Correlation Function R - Landmark Signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

Cross Correlation Function R - Landmark Signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t) g(\tau + t)$$

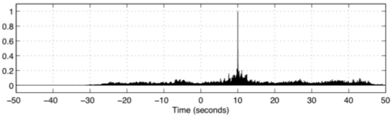
$$R_{L_i,L_j}(\tau) = \sum_{t=-\infty}^{\infty} L_i(\tau)^T L_j(\tau + t)$$

Cross Correlation Function R - Landmark Signals

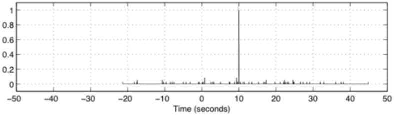
$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t) g(\tau + t)$$

$$R_{L_i,L_j}(\tau) = \sum_{t=-\infty}^{\infty} L_i(t)^T L_j(\tau + t)$$

Landmark Signals - Why?



(a) Normalized absolute time-domain cross-correlation.



(b) Normalized landmark cross-correlation.

Figure: Absolute vs. Landmark Cross Correlation (Bryan, Smaragdis, Mysore, 2012)

Generating Landmark Signals - How?

Follow Steps 1-4

1. Take the Short Time FT of a given audio signal

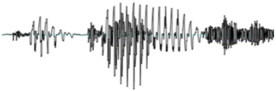


Figure: Given audio signal

$\xrightarrow{\text{STFT}}$

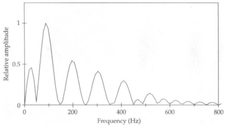


Figure: Spectrum

(Images: Wagner, 2017)

2. Identify frequency peaks and create landmark tuple

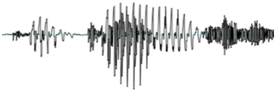


Figure: Given audio signal

STFT
⟶

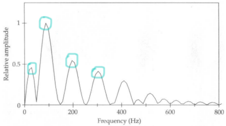


Figure: Spectrum

(Images: Wagner, 2017)

2. Identify frequency peaks and create landmark tuple

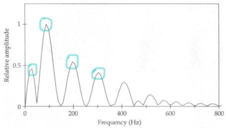


Figure: Spectrum

$$\text{landmark} = (\text{frequency}, \text{timevalue}) = (f_1, t_1)$$

"time idexed frequency value"

2. Identify frequency peaks and create landmark tuple

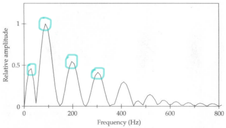


Figure: Spectrum

$$\text{landmark} = (f_1, t_1)$$

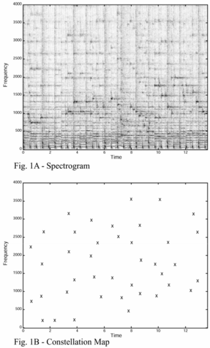


Figure: Spectrogram (Wang, 2003)

3. Pair landmarks with nearest other landmarks

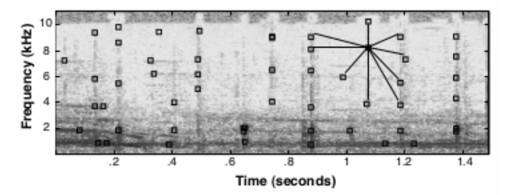


Figure: Spectrogram (Kennedy, Naaman, 2009)

3. Pair landmarks with nearest other landmarks

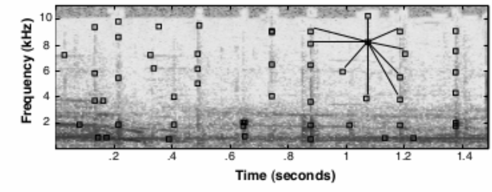


Figure: Spectrogram (Kennedy, Naaman, 2009)

$$\text{landmark-pair} = \text{landmark} = (\text{frequency}, \text{frequency}, \text{timevalue})$$
$$= (f_1, f_2, t_2 - t_1)$$

"time indexed landmark"

4. Hash each landmark and create feature vector

$$hash : landmark \rightarrow integer$$

$$(f_1, f_2, t_2 - t_1) \mapsto 2$$

4. Hash each landmark and create feature vector

$$hash : landmark \to integer$$

$$(f_1, f_2, t_2 - t_1) \mapsto 2$$

create feature vector: $\begin{pmatrix} 0 \\ 0 \\ \dots \\ 0 \end{pmatrix}$

4. Hash each landmark and create feature vector

$$hash : landmark \to integer$$

$$(f_1, f_2, t_2 - t_1) \mapsto 2$$

create feature vector: $\begin{pmatrix} 0 \\ 0 \\ \dots \\ 0 \end{pmatrix} \to \begin{pmatrix} 0 \\ 1 \\ \dots \\ 0 \end{pmatrix} = L(t = 6)$

4. Hash each landmark and create feature vector

$$hash : landmark \rightarrow integer$$

$$(f_1, f_2, t_2 - t_1) \mapsto 2$$

$$\text{create feature vector: } \begin{pmatrix} 0 \\ 0 \\ \dots \\ 0 \end{pmatrix} \rightarrow \boxed{\begin{pmatrix} 0 \\ 1 \\ \dots \\ 0 \end{pmatrix} = L(t = 6)}$$

Understanding R

$$\boxed{\begin{pmatrix} 0 \\ 1 \\ \dots \\ 0 \end{pmatrix} = L(t=6)}$$

$$R_{L_i,L_j}(\tau) = \sum_{t=-\infty}^{\infty} L_i(t)^T L_j(\tau + t)$$

Understanding R

$$R_{L_i,L_j}(\tau) = \sum_{t=-\infty}^{\infty} \underbrace{L_i(t)^T L_j(\tau + t)}$$

$$\begin{pmatrix} 0 & 1 & \dots & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \dots \\ 1 \end{pmatrix} = constant$$

Understanding R

$$R_{L_i,L_j}(\tau) = \sum_{t=-\infty}^{\infty} \underbrace{L_i(t)^T L_j(\tau + t)}$$

$$\begin{pmatrix} 0 & 1 & \dots & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \dots \\ 1 \end{pmatrix} = constant$$

"Amout of matching landmarks in both signals at one point in time"

Understanding R

$$R_{L_i,L_j}(\tau) = \sum_{t=-\infty}^{\infty} \underbrace{L_i(t)^T L_j(\tau + t)}$$

$$\begin{pmatrix} 0 & 1 & \dots & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \dots \\ 1 \end{pmatrix} = \mathit{constant}$$

"Amout of matching landmarks in both signals at one point in time"

"Total amount of matching landmarks for one time shift τ"

Understanding R

Video

(https://www.youtube.com/watch?v=L6YJqhbsuFY)

Synchronizing both signals

$$\hat{\tau}_{ij} = arg \max_{\tau} R_{L_i, L_j}(\tau)$$

Synchronizing both signals

$$\hat{\tau}_{ij} = arg \max_{\tau} R_{L_i,L_j}(\tau)$$

"time offset to align both signals L_i and L_j"

Complexity ($n \hat{=}$ *file length*)

- 2 loops over t, τ $\longrightarrow$ $\mathbf{O(n^2)}$

Complexity ($n \hat{=}$ *file length*)

- 2 loops over t, τ $\longrightarrow$ $\mathbf{O(n^2)}$
- use FFT: no need to loop over t anymore, only 1 loop over τ $\longrightarrow$ $\mathbf{O(n \log n)}$

FFT of the two signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

FFT of the two signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t) g(\tau + t)$$

Frequency Domain: $FFT(f^*)$

FFT of the two signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

Frequency Domain: $FFT(f^*) \quad FFT(g)$

FFT of the two signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

Frequency Domain: $FFT(f^*) * FFT(g)$

FFT of the two signals

$$R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$$

Frequency Domain: $FFT(f^*) * FFT(g)$

Time Domain: IFFT ($FFT(f^*) * FFT(g)$)

FFT of the two signals

$$R_{f,g}(\tau) = \xcancel{\sum_{t=-\infty}^{\infty}} f^*(t)g(\tau + t)$$

Frequency Domain: $FFT(f^*) * FFT(g)$

Time Domain: IFFT ($FFT(f^*) * FFT(g)$)

Take home

1. $R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$

Take home

1. $R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$
2. Reduce costs by transforming absolute signals f, g to Landmark signals L_i, L_j

Take home

1. $R_{f,g}(\tau) = \sum_{t=-\infty}^{\infty} f^*(t)g(\tau + t)$
2. Reduce costs by transforming absolute signals f, g to Landmark signals L_i, L_j
3. Reduce complexity by applying the FFT to both signals before multiplying them, then apply IFFT to the result

Sources

"Clustering and Synchronizing Multi-Camera Video via Landmark Cross-Correlation", Bryan, Smaragdis, Mysore, 2012

https://www.youtube.com/watch?v=L6YJqhbsuFY

Wagner, 2017 (Lecture "Phonetik und Phonologie" Slides "Akustische Phonetik")

"An Industrial-Strength Audio Search Algorithm", Wang, 2003

"Less Talk, More Rock: Automated Organization of Community-Contributed Collections of Concert Videos", Kennedy and Naaman, 2009