$\boxed{\text{Lab 06}}$ filterbank approaches for noise suppression and feature extraction

개요
① uniform filterbank energies for noise suppression
② mel-scale 〃 〃 〃 〃
③ log filterbank energies for speech recognition
　mel-frequency filterbank energies for speech recognition
④ MFCC : mel-frequency cepstral coefficients
⑤ dynamic time warping (DTW) for isolated word recognition

I. uniform filterbank energies for noise suppression

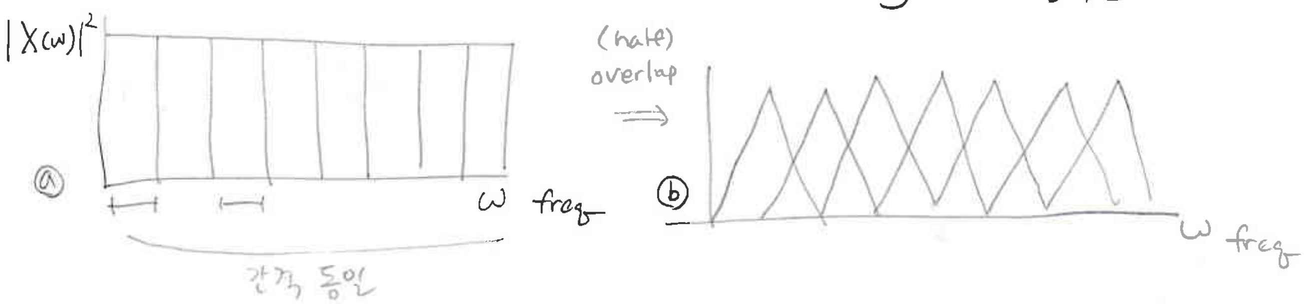
noise suppression을 위한 multi-variate probabilistic model을 만들 때 입력 dimension이 매우 크다

(20ms @ 16kHz → 320 samples → 512 FFT → 257 0~π freq components)

dimension 크기가 큰 것 자체는 문제가 되지 않지만

① energy가 거의 0인 frequency componenents : 주로 high
② 인접한 성분들이 유사함 : 연산 낭비
③ 어차피 FIR filter는 31 tab (order 30)을 쓰는데
　257 >> 31

따라서, 전처리 257 차원을 filterbank energies로 줄여보자.



동일한 간격의 bandpass filter들을 N_{fb}개 만들고 평균 energy를 계산한다.

ⓐ와 같이 크기가 $\frac{F_{Nyquist}}{N_{fb}}$로 고정된 사각 bandpass filter들로 만들 수도 있지만 경계 (boundaries)에서 불연속이 생기기 때문에 ⓑ와 같이 삼각 필터들로 구현하는 것이 더 일반적

$$E_i = \sum_{k=1}^{K_f} W_{fb}(i,k) \cdot |X(\omega_k)|^2$$

← sum을 할 때는 $|\cdot|^2$으로
∵ $|X(\omega_1)| + |X(\omega_2)|$ 와 같이 절대값의 합은 말이 안 됨

$W_{fb}(1,:)$
$W_{fb}(2,:)$
$W_{fb}(3,:)$

Gaussian, Hanning 등으로 Smoothed sum을 할 수도 있지만 굳이 그럴 필요 없음

* 참고: 용어 정리

1) 문서들에서 filter banks, filterbanks 라고 많이 적히는데
엄밀히 말하면 틀린 표현임
filter "bank" 자체가 "set of" filters 라는 의미이기 때문
따라서 filterbank (filter bank) energies 또는 bank of filters
가 맞는 표현

2) $|X(\omega)| + |Y(\omega)|$ vs $|X(\omega)|^2 + |Y(\omega)|^2$?
우리가 원하는 것은 $|X(\omega) + Y(\omega)|$ 을 근사하는 것
$$|X(\omega) + Y(\omega)|^2 = X^2 + \underbrace{2XY}_{=0 \text{ iff } X \text{ and } Y \text{ are uncorrelated}} + Y^2 \approx X^2 + Y^2$$
$|X(\omega) + Y(\omega)| \fallingdotseq |X(\omega)| + |Y(\omega)|$
근사할만한 backgroud가 없다

3) Energy vs Power
$$E_f = \int_{-\infty}^{\infty} |f(t)|^2 dt$$ Energy는 무한대 함
$$P_f = \lim_{T\to\infty} \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} |f(t)|^2 dt$$ 평균 energy로 볼 수 있음
practically, finite range에서는 상수차이
(구간길이)
Short-time Fourier transform에서는 큰 차이가 없으며
혼용되고 있음 (정의상 구간길이로 나누는 것이 없기 때문에
Fourier transform은 energy가 맞긴 함)

4) Spectrum, Why?
빛의 분광기 (spectrum analyser)에서 유래함.
빛은 파장에 따라 프리즘을 통과하면 여러가지 색으로 나타나고
주파수 분석 (Fourier analysis) 역시 같은 의미를 가짐.

$|X(\omega)|^2$ 은 spectral energy
$|X(\omega)|$ 는 spectral magnitude

Noise Suppression 구현

$$\{Y(\omega)\}_{K} \xrightarrow[\text{filter bank}]{} \{E_Y(i)\}_{N_{fb}} \rightarrow \{H_E(i)\} \xrightarrow[\text{inversion}]{} \{H(\omega)\}$$

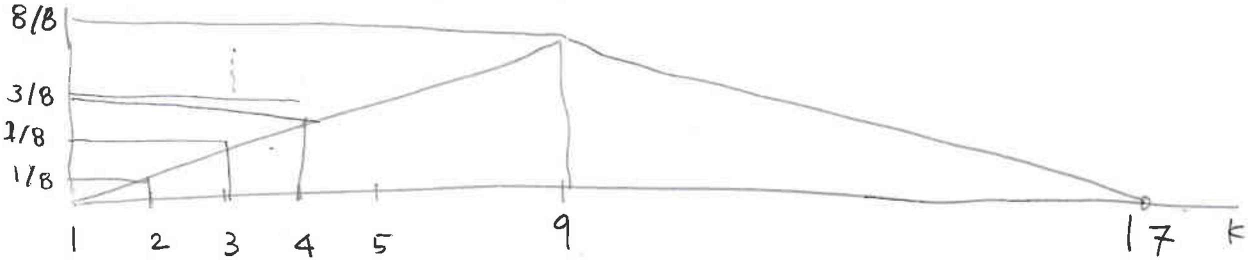
filter의 개수는 30~40개가 일반적이고 order/2 ≤ ≤ $N_{fft}/2+1$ 이어야 함.

여기 예제는 나누어 떨어지도록 ~~3~~2개로 해 본다 (31개)

$N_{fft} = 512 \rightarrow N_{0\sim\pi} = 257$ boundaries

구간의 폭은 256/32 = 8 따라서 삼각 필터의 크기는

0과 1을 포함해서 8+1+8 = 17



$W_{fb}(1, 1{:}17) = [\overset{①}{0},\ 1/8,\ 2/8, \ldots\ldots, 7/8,\ 1,\ 7/8, \ldots\ldots, 1/8,\ \overset{⑰}{0}]$

나머지는 0

$W_{fb}(2, 9{:}25) = [0 \ldots\ldots 1 \ldots\ldots 0]$

$W_{fb}(3, 17{:}33) =$ //

⋮

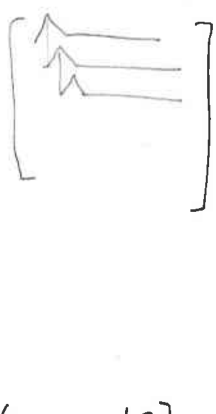
$W_{fb}(31, 241{:}257) =$ //

* $\omega = 0$, $\omega = \pi$ 는 W_{fb}가 0이다. 고려하면 dc 성분이라 필요없다.

* matlab indexing, 시작과 끝 포함, 1-based indexing

numpy로 작성하면 $W_{fb}(1, 1{:}17) \rightarrow W_f[0, 0{:}17]$

$$W_{fb} = \frac{1}{8}\begin{bmatrix} 0 & 1 & \cdots & 8 & \cdots & 1 & 0 & 1 & \cdots & & & \\ & & & & & 0 & 1 & \cdots & 7 & 8 & 7 & \cdots & 1 & 0 \\ & & & & & & & & 0 & 1 & \cdots & 7 & 8 & 7 & \cdots \\ & & & & & & & & & & & & & \end{bmatrix} =$$



처음 7개와 마지막 7개 제외하면 모두 1

$\mathrm{Sum}(W_{fb}, \mathrm{axis}=0) = \frac{1}{8}[0, 1, 2 \ldots\ 8\ 8\ 8 \_\_\_\_\_\_\_\_ 8\ 7\ 6 \ldots 1\ 0]$

$$\underset{31\times1}{E_{fb}} = \underset{31\times257}{W_{fb}} \, \underset{257\times1}{|Y(\omega)|^2}$$

31개의 filterbank energies를 이용, RMM, GMM 등으로 Noise estimate의 FB energy를 구함 ($E_{fb}^{(\tilde{N})}$)

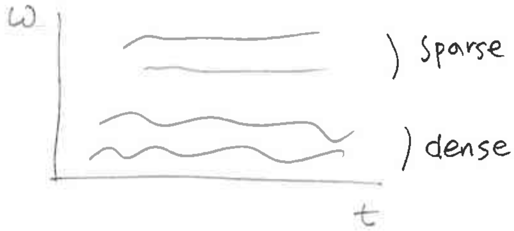
$$\underset{257\times1}{|\hat{N}(\omega)|^2} = \underset{257\times31}{W_{fb}^T} \, \underset{31\times1}{E_{fb}^{(\tilde{N})}} \Rightarrow H(\omega) = \max\left\{\varepsilon, \frac{|Y(\omega)|^2 - |\hat{N}(\omega)|^2}{|Y(\omega)|^2}\right\}$$

* 이 구현 예제에서는 W_{fb}의 column 합이 1이라서 문제가 없지만 (lowest highest 제외) 겹침이 많도록 filterbank를 설계하면 W_{fb}^T를 적절히 normalize해 주어야 한다

즉, $$|\hat{N}(\omega_k)|^2 = \frac{W_{fb}(k,:) \, E_{fb}^{(\tilde{N})}}{\sum W_{fb}(k,:)}$$

* Nfft가 Nfb로 나누어 떨어지지 않으면 center frequency index가 정수가 아니게 되고 filterbank coefficients를 공유할 수 없다. 실용적으로는 가장 가까운 integer index로 치환하여 사용

II. mel-scal filterbank



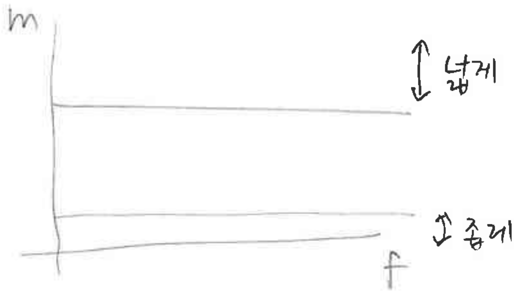
spectrum은 그림과 같이 low frequency 대역은 dense하게 분포하지만 high frequency 대역은 매우 sparse하고 분해능(resolution)이 떨어진다. 따라서, 모든 주파수에 uniform bandwidth를 줄 필요는 없다
인간의 청각특성을 반영한 mel-scale로

*mel-scale (logarithmic), wikipedia

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right)$$ 상용로그

$$m = 1127 \ln\left(1 + \frac{f}{700}\right)$$ 자연로그

$$m = \frac{1000}{\log_b 2} \log_b\left(1 + \frac{f}{700}\right)$$ any base $b > 0$



Lab 06 page 5/6

mel filter의 수는 8kHz sampling rate에서 24개 ~ 29개
16 kHz에서 30 ~ 40 개 정도를 쓴다.

1) 최대 주파수에서 mel 값을 구한다.

$$m(F_s/2) = 1127 \ln\left(1 + \frac{F_s/2}{700}\right)$$

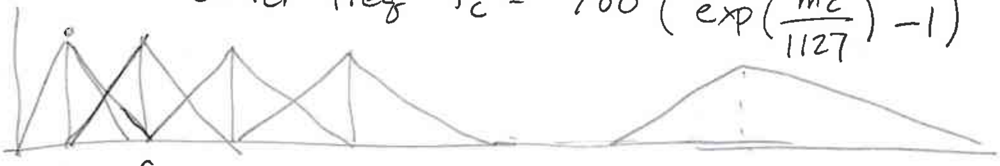
2) $0 \sim m(F_s/2)$를 center mel m_c N_{fb}개를 구한다

⇒ $(N_{fb}+2)$개의 구간들로 나누고 처음과 마지막 제외한다

$$\text{Center mel } m_c = \text{linspace}\left(0,\ m\left(\tfrac{F_s}{2}\right),\ N_{fb}+2\right) \underbrace{[1:-1]}_{\text{처음과 끝 제외}}$$

3) center freq들을 중심으로 삼각 필터를 만든다

$$\text{Center freq } f_c = 700\left(\exp\left(\frac{m_c}{1127}\right) - 1\right)$$



* non-integer center

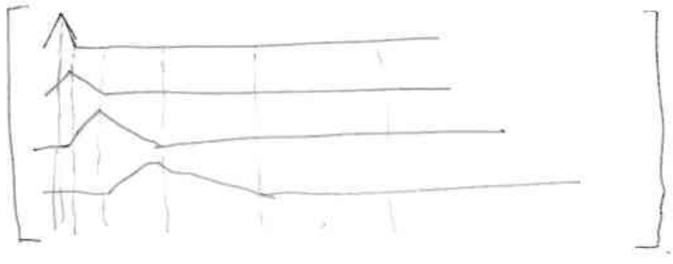
f_c가 FFT bin들의 center와 일치하지 않으면 W_{fb}는 center에서 1이 아니다.

즉,

ⓐ 가장 가까운 bin으로 옮긴다.

ⓑ center가 꼭 bin에 맞지 않아도 되니 그냥 W_{fb}를 쓴다. 단, 구현에 주의

W_{fb} =



더하면 1 (처음과 마지막 구간 제외)

더하면 1

$$\underset{N_{fb}\times 1}{E^{(n)}} = \underset{N_{fb}\times 257}{W_{fb}}\ \underset{257\times 1}{Y} \xrightarrow[\text{RMM / GMM}]{} E^{(\tilde N)} \rightarrow \hat N = W_{fb}^T E^{(\tilde N)} \xrightarrow{\text{Wiener}} H(\omega) \xrightarrow{\text{ifft}+\sim} h[n]$$

↓ FIR filtering

$\hat{x}[n]$

III. uniform & mel-scale log filterbank energies for speech recognition

사람은 음성의 크기의 비로 차이를 인식함.

즉, $\frac{|X_1(\omega)|}{|Y_1(\omega)|} = \frac{|X_2(\omega)|}{|Y_2(\omega)|}$ 이면 같은 perceptual distance라고 생각한다.

따라서 log filterbank energy를 음성인식을 위한 특징으로 사용가능

use $\log E[i]$, $\log E'_{mel}[i]$ for speech recognition

IV. Mel-frequency cepstral coefficients

* 의미 및 이론 → 강의 자료

* uniform filterbank or low spectrum으로부터 cepstrum을 얻을 수도 있지만 거의 쓰지 않는다.

MFCC 구현

$$Y(\omega) \longrightarrow E_{mel}[i] \rightarrow \log E_m[i] \xrightarrow{DCT} C[i]$$

$$Y(k,\omega) = STFT[y[n]]$$

$$\underset{1\times 40}{E_{mel}[k,:]} = \underset{1\times 257}{|Y(k,:)|^2} \cdot \underset{257\times 40}{W_{mfb}^T} \quad \text{for } N_{fb}=40$$

$$\underset{1\times 12}{C[k,1{:}13]} = \underset{1\times 40}{E_{mel}[k,:]} \cdot \underset{40\times 12}{W_{DCT}^T}$$

cepstral coefficients의 수는 12개 (no question)

$$C[k,0] = \log \sum_{i \in \text{frame } k} y^2[i] \approx \log \sum_{\omega} |Y(k,\omega)|^2$$

첫번째 dimension (구현에 따라서 마지막 dim)은 log frame energy사용.

* C[0] energy or 0-quefrency component

DCT의 정의에 따라 (또는 IFFT) C[0]는 zero quefrency 성분. 따라서 $\cos(2\pi \cdot 0 \cdot i) = \cos 0 = 1$ 이 모든 $\log E$에 곱해져서 더한 값을 써야 하지만 그러면 $\sum \log E = \log \prod E$ 즉, 모든 성분의 값을 곱한 것이 되어 의미가 없다. 따라서 C[0]는 안쓰거나 log Energy는 어쨌든 사용.

* Why real only

$\cos\left(\frac{2\pi}{N_{fb}}(n+\frac{1}{2}) \cdot i\right)$ 로 계산한다.

- cepstrum의 정의에 따라 IFFT를 써야 하지만 어차피 $0 \sim \pi$만 있으면 되므로 $0 \sim \pi$ 성분만 곱하는 DCT로 충분
- phase 고려 필요없으므로 real 성분만 사용