Traditional DSP for Audio & Speech

Tal Rosenwein

Goal

Image source

Overview

- Motivation
- Communication
- Anatomy & Speech production system
- Phonetics
- Acoustic/Speech Features
- Traditional Speech Signals Analysis
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression

Motivation

- Huge Market
- Make Our Lives Easier
- Save Human Labour
- Remove Bureaucracy

Source 1, 2

Motivation

- Huge Market
- Make Our Lives Easier
- Save Human Labour
- Remove Bureaucracy

VNI Complete Forecast Highlights

- Globally, consumer Internet video traffic was 56.4 EB per month in 2017, the equivalent of 14 billion DVDs per month, or 19 million DVDs per hour.
- Global consumer Internet video traffic grew 40% in 2017.
- Globally, Internet video traffic will be 82% of all consumer Internet traffic by 2022, up from 73% in 2017.

Source 1

Overview

- Motivation
- **Communication**
- Anatomy & Speech production system
- Phonetics
- Acoustic/Speech Features
- Traditional Speech Signals Analysis
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression

Communication

- Speaker produces acoustic wave that emitted from the mouth/nostrils (nose) and propagates to the hears of the listener
- Multiple organs (nostrils, mouth, throat, teeth, tongue, etc.) are included in speech generation.

Communication

Conversation flow:

- Creating the notion we want to transmit.
- Mapping / transducing this notion into a linguistic domain
- Choosing the words that best represent this notion
- Arranging the words in order according to linguistic rules of a given language
- Adding prosodic features for emphasize different aspects of the notion we want to convey
- The brain sends a series of motoric commands
- We then move the relevant mussels
- Produce the sequence of sounds
- Transmitting an acoustic wave to the ears of the partner through the medium.

Overview

- Motivation
- Communication
- **Anatomy & Speech production system**
- Phonetics
- Acoustic/Speech Features
- Traditional Speech Signals Analysis
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression

Anatomy & Modelling of Speech Production

Audio- Pressure Wave

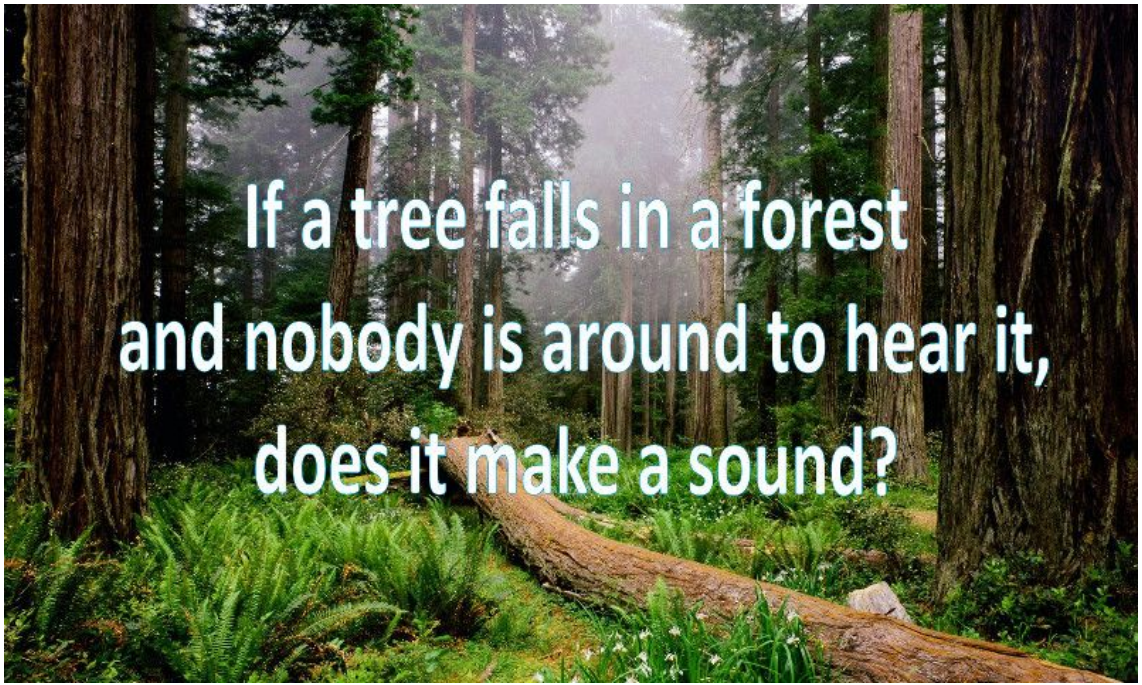


Image source

Anatomy & Modelling of Speech Production

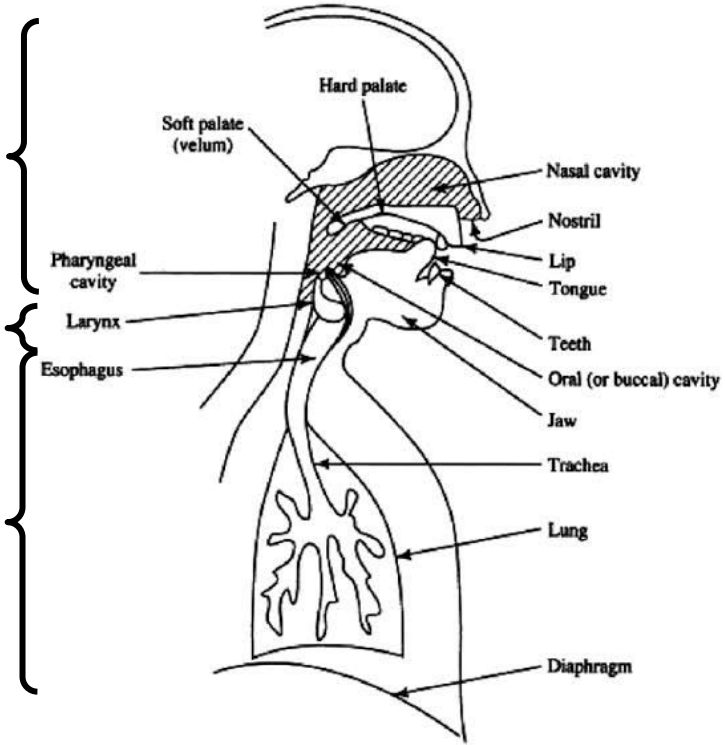
Speech Production Organs

- video

Vocal tract

Larynx

Subglottal Respiratory System

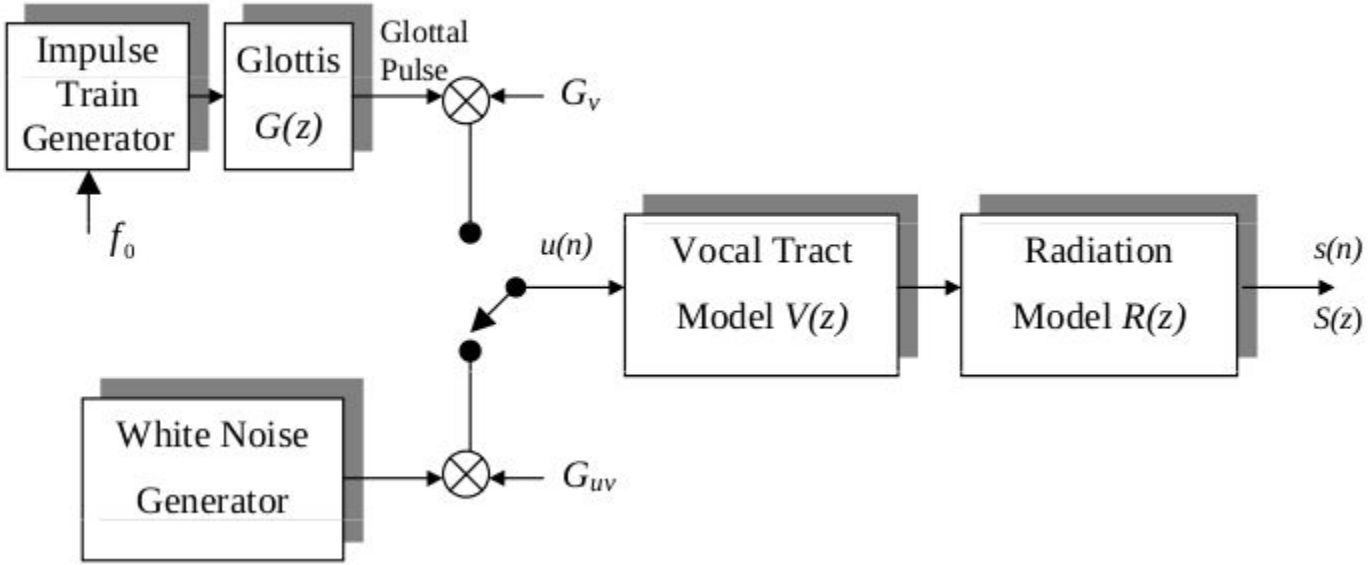


Source 1

Speech Production- Source-Filter Model

Voiced

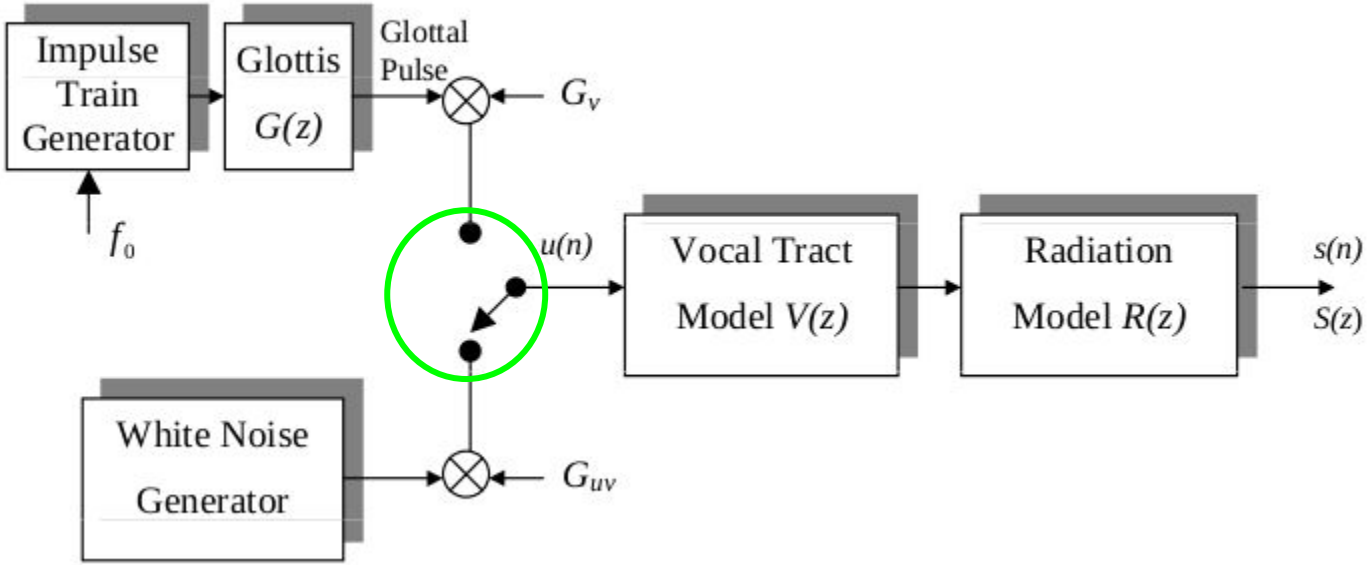
Un-Voiced



Speech Production- Source-Filter Model

Voiced

Un-Voiced



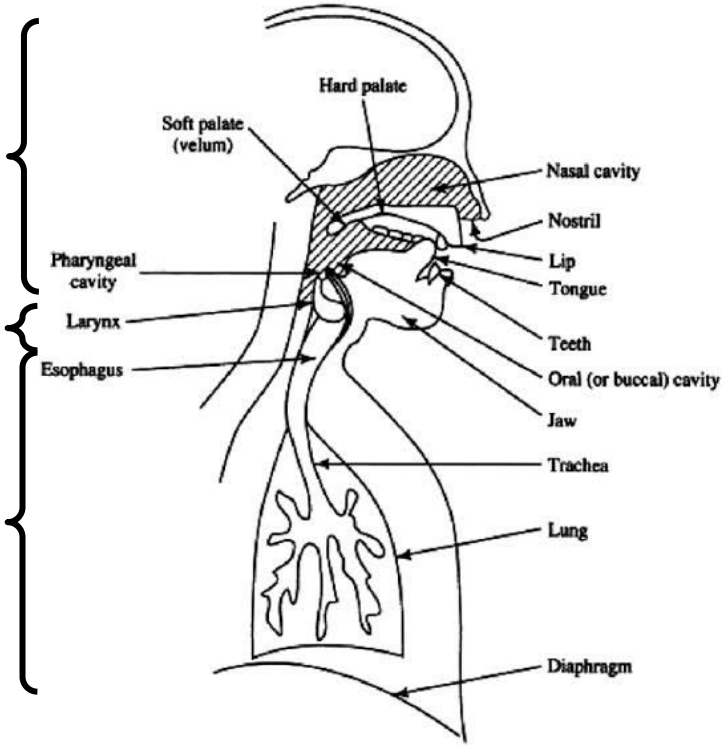
Speech Production Organs

- video

Vocal tract

Larynx

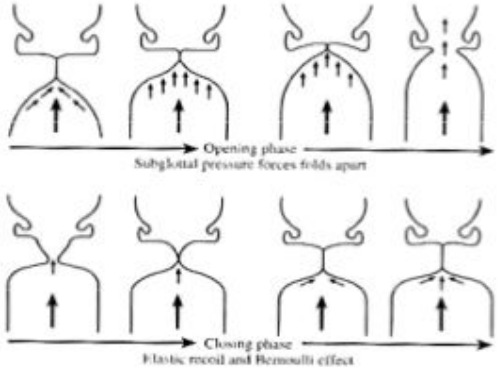
Subglottal Respiratory System

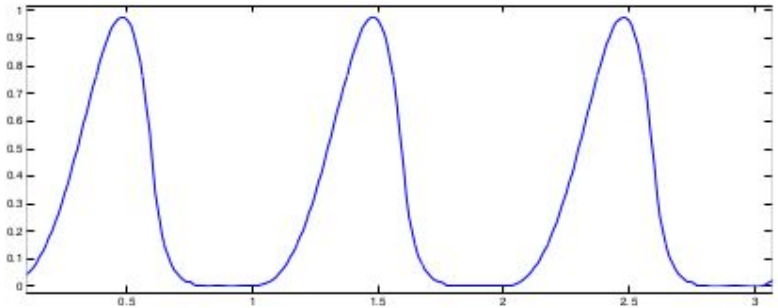


Source 1

Speech Production Organs - Larynx

- Larynx is responsible for different phonation modes (voiceless, voiced, whisper)
- During voiceless phonation (e.g. [s] in “star”) and whisper, vocal cords are apart from each other, and the airstream passes through the open glottis (the opening between the vocal cords)
- In voiced phonation (e.g. [a] in “star”), vocal folds are successively opening and closing, creating a periodic waveform. The frequency of vibration is called fundamental frequency (or pitch) and abbreviated F0.
- “vocal cords in action”- video. This video is less disgusting.

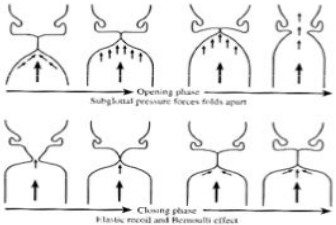


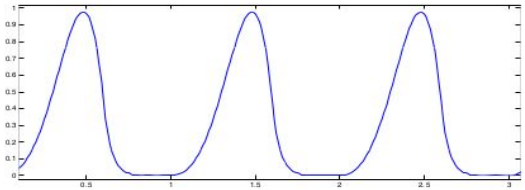


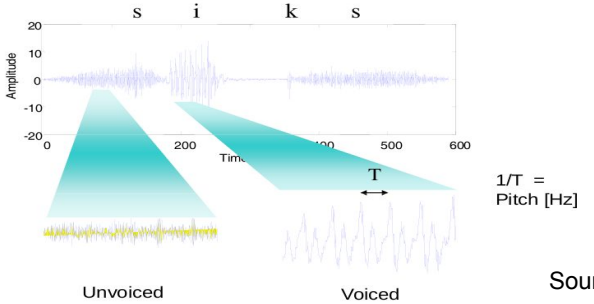
Source 1

Speech Production Organs - Larynx

- Larynx is responsible for different phonation modes (voiceless, voiced, whisper)
- During voiceless phonation (e.g. [s] in “star”) and whisper, vocal cords are apart from each other, and the airstream passes through the open glottis (the opening between the vocal cords)
- In voiced phonation (e.g. [a] in “star”), vocal folds are successively opening and closing, creating a periodic waveform. The frequency of vibration is called fundamental frequency (or pitch) and abbreviated F0.
  - Average F0 values for male, female, and children are about 120, 220 and 330 Hz.
  - The shape of the glottal pulse is individual, and it is an important determinant of the perceived voice quality (“She has a harsh / rough / breathy / clear /.. voice”)

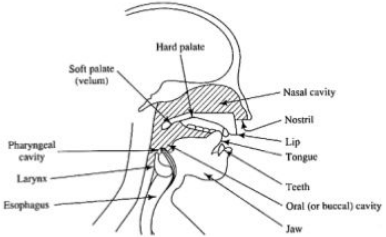


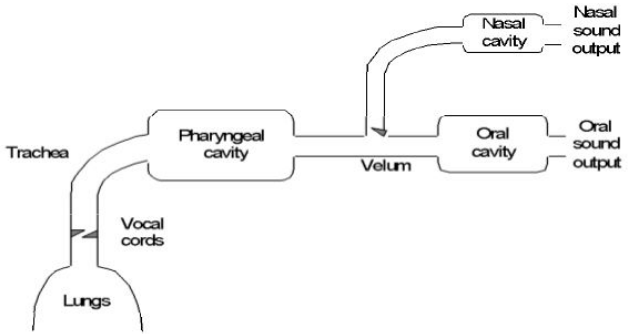




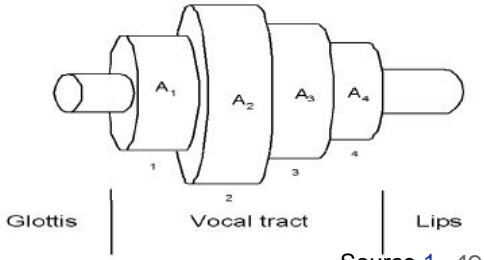
Speech Production Organs - Vocal Tract

- Includes pharyngeal, oral and nasal cavities. The volumes and shapes of these are individual.
- Can be roughly characterized by its (time-varying) resonance frequencies called formants





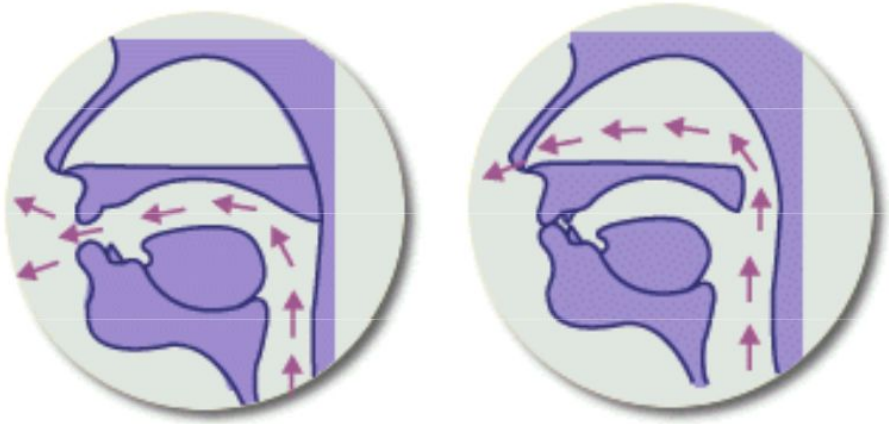
- Can be modeled as a hard-walled lossless tube resonator consisting of *N* tubes with different cross-sectional areas
- Can be modeled by a (time-varying) filter.
- Example: Car Exhaust



Source 1

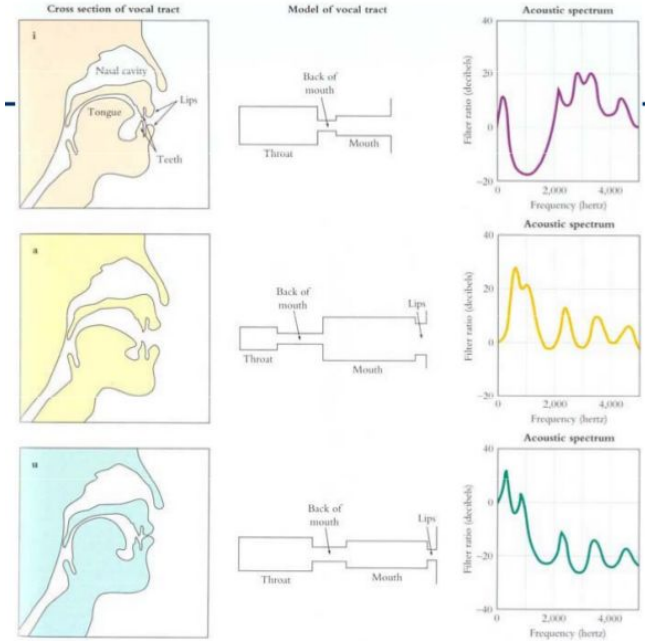
Speech Production Organs - Vocal Tract

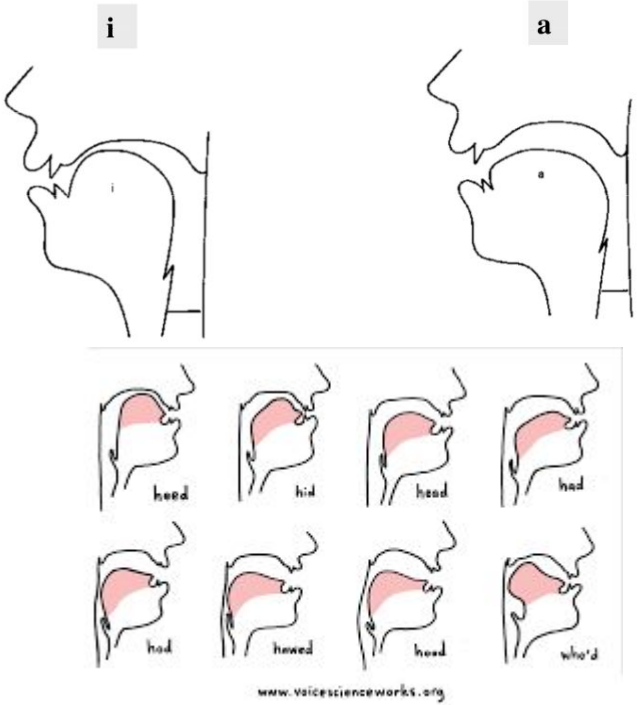
- video
- Oral Vs Nasal sounds:
  - Close your nose and say ‘mmmm’ for 5 sec
  - Close your nose and say ‘ Laaaaaa’ for 5 sec



Speech Production Organs - Vocal Tract

- Vocal Tract (also lips, tongue...) changes when different vowels are pronounced

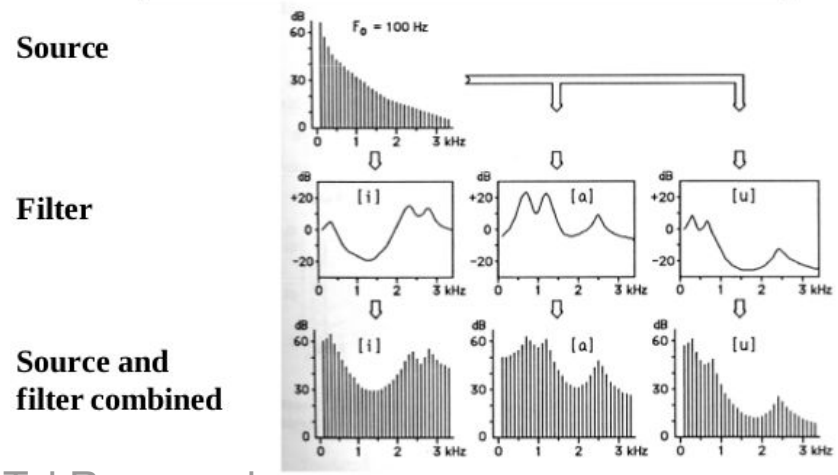


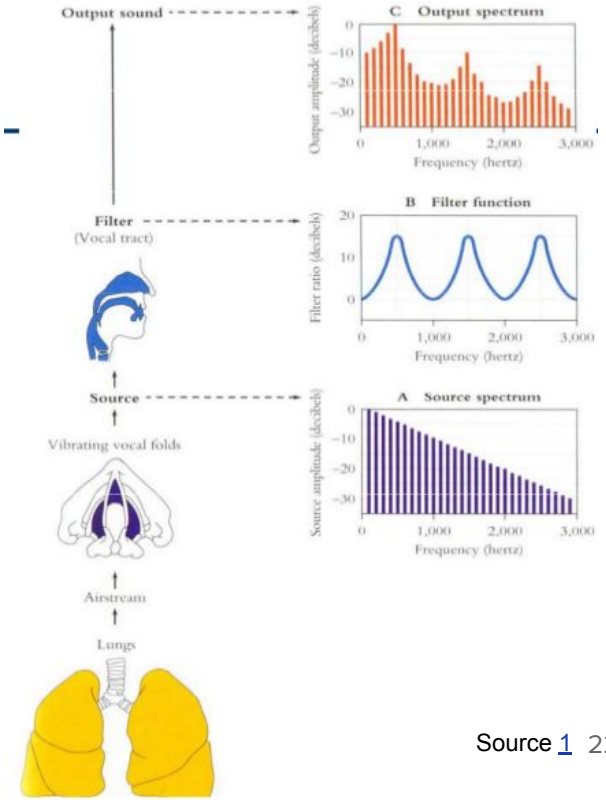


Speech Production- Source-Filter Model

- The resulting speech spectrum S(z) is a combination of the source spectrum U(z) and the vocal tract transfer function H(z):

$S(z) = U(z)H(z)$	$\leftrightarrow$	$s[n] = u[n] * h[n]$
Frequency domain		Time domain

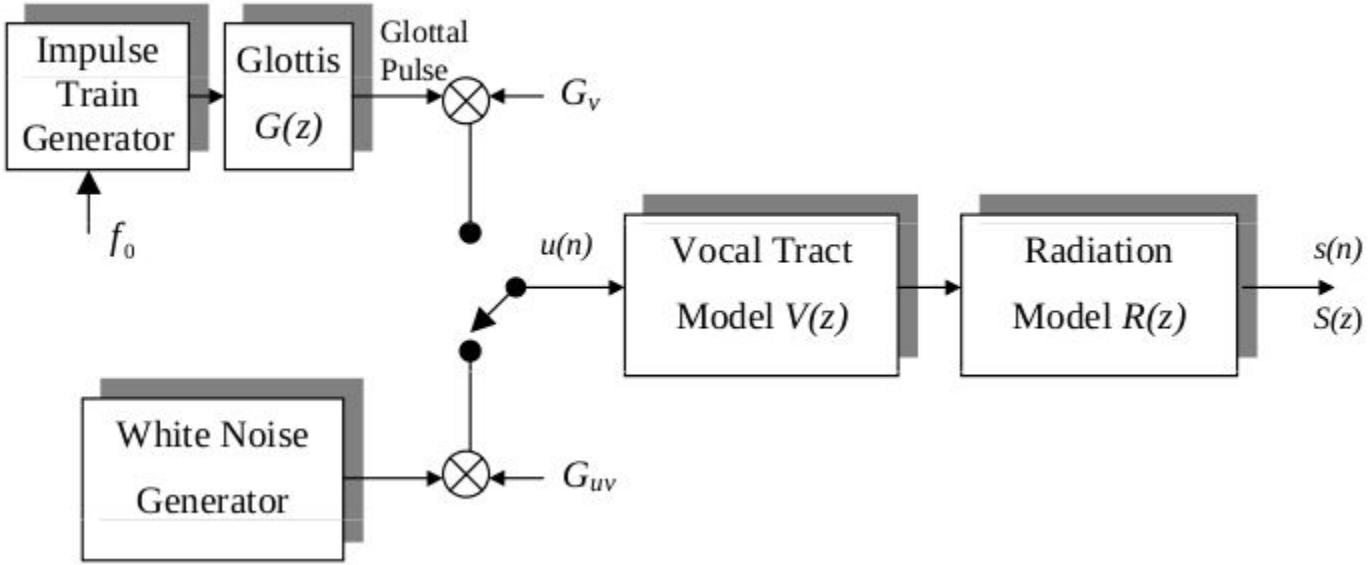




Speech Production- Source-Filter Model

Voiced

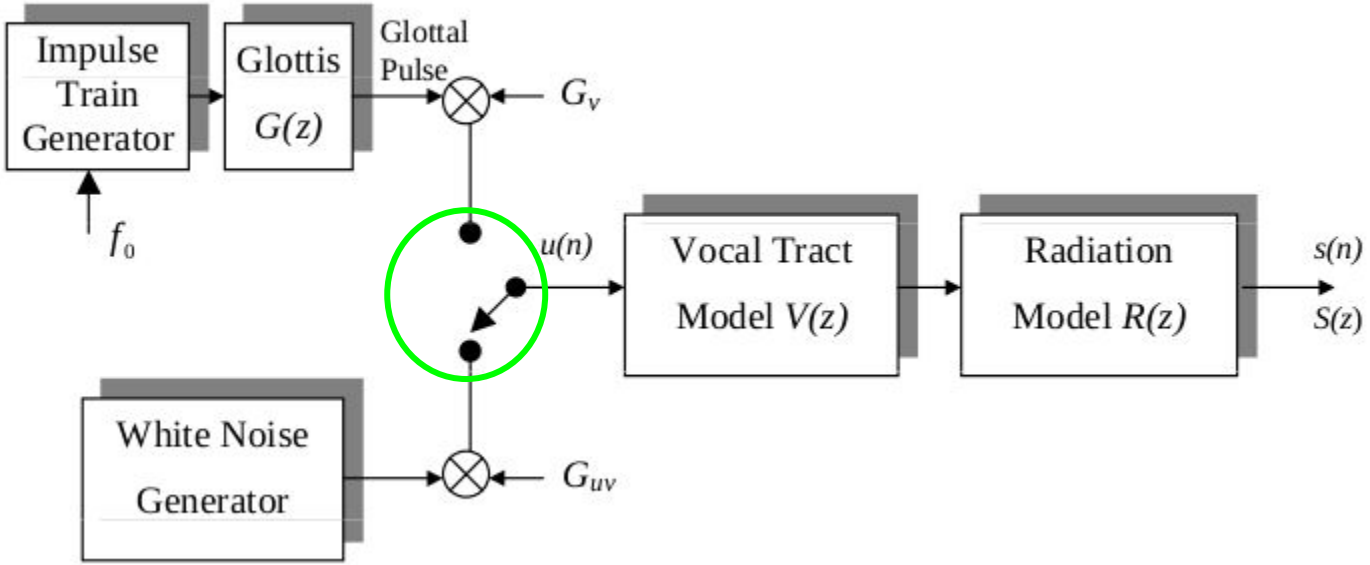
Un-Voiced



Speech Production- Source-Filter Model

Voiced

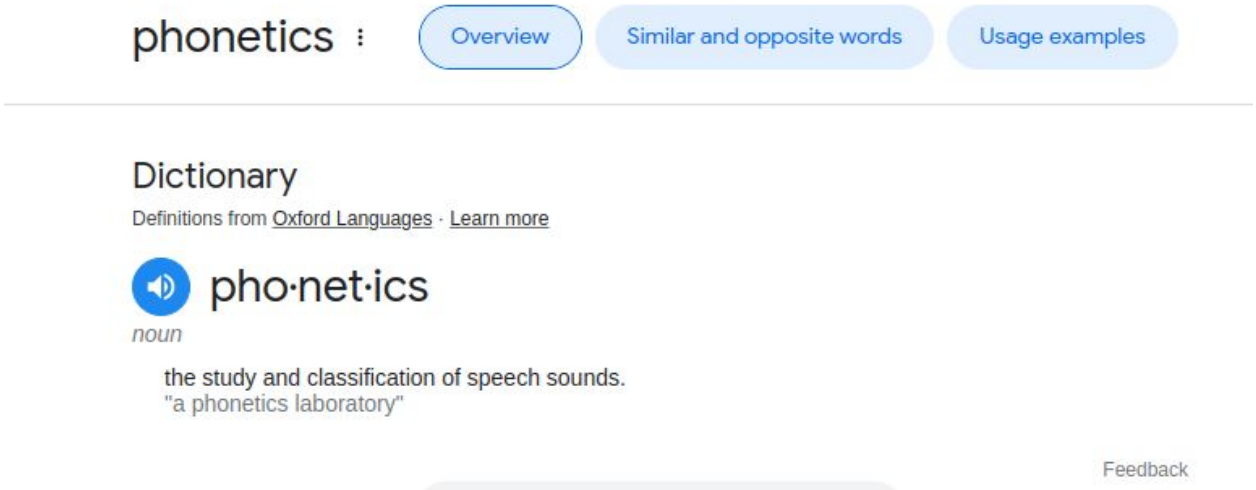
Un-Voiced



Overview

- Motivation
- Communication
- Anatomy & Speech production system
- **Phonetics**
- Acoustic/Speech Features
- Traditional Speech Signals Analysis
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression

Phonetics



Source Google

Phonetics

Phonetic has 3 branches:

- **Articulatory phonetics:** how speech-sound is generated, in terms of mechanics
- **Acoustic phonetics:** how speech-sounds are transmitted, in terms of acoustics.
- **Auditory/perceptual phonetics:** how sound is perceived by our brain

Source Google

Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- **Phone:** Distinct speech sound or gesture, regardless of whether the exact sound is critical to the meaning of words. Phoneme is the basic / atomic piece to convey a lingual message.
  - “Guimar**ã**es”- we can produce **ã** but it has no meaning in English
- **Allophone:** group of phones represented by the same phoneme (variations of phonemes). For example different pronunciation of the letter t in top, stop, butter, are allophones of the phoneme /t/.

Source Google

Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/

Phoneme	Example	Translation
AA	odd	AA D
AE	at	AE T
AH	hut	HH AH T
AO	ought	AO T
AW	cow	K AW
AY	hide	HH AY D
B	be	B IY
CH	cheese	CH IY Z
D	dee	D IY
DH	thee	DH IY
EH	Ed	EH D
ER	hurt	HH ER T
EY	ate	EY T
F	fee	F IY
G	green	G R IY N
HH	he	HH IY
IH	it	IH T
IY	eat	IY T
JH	gee	JH IY
JH	gee	JH IY
K	key	K IY
L	lee	L IY
M	me	M IY
N	knee	N IY
NG	ping	P IH NG
OW	oat	OW T
OY	toy	T OY
P	pee	P IY
R	read	R IY D
S	sea	S IY
SH	she	SH IY
T	tea	T IY
TH	theta	TH EY T AH
UH	hood	HH UH D
UW	two	T UW
V	vee	V IY
W	we	W IY
Y	yield	Y IY L D
Z	zee	Z IY
ZH	seizure	S IY ZH ER

Source 1

Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- Different lexicons / standards: **IPA**, ARPABET

Phoneme	Example	Translation
AA	odd	AA D
AE	at	AE T
AH	hut	HH AH T
AO	ought	AO T
AW	cow	K AW
AY	hide	HH AY D
B	be	B IY
CH	cheese	CH IY Z
D	dee	D IY
DH	thee	DH IY
EH	Ed	EH D
ER	hurt	HH ER T
EY	ate	EY T
F	fee	F IY
G	green	G R IY N
HH	he	HH IY
IH	it	IH T
IY	eat	IY T
JH	gee	JH IY
JH	gee	JH IY
K	key	K IY
L	lee	L IY
M	me	M IY
N	knee	N IY
NG	ping	P IH NG
OW	oat	OW T
OY	toy	T OY
P	pee	P IY
R	read	R IY D
S	sea	S IY
SH	she	SH IY
T	tea	T IY
TH	theta	TH EY T AH
UH	hood	HH UH D
UW	two	T UW
V	vee	V IY
W	we	W IY
Y	yield	Y IY L D
Z	zee	Z IY
ZH	seizure	S IY ZH ER

Source 1

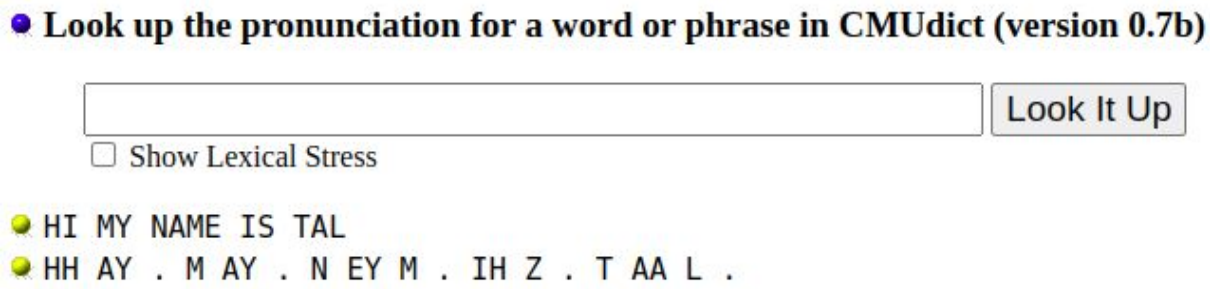
Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- Different lexicons / standards: **IPA**, ARPABET
  - IPA lexicon has 44 phonemes, ARPABET has 46.
  - In English there are 65 phonemes, usually people compress it to 39 phonemes.

Source 1

Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- There are some lexicons, such as the CMU dictionary that breaks each word to it's phoneme sequence. Online link

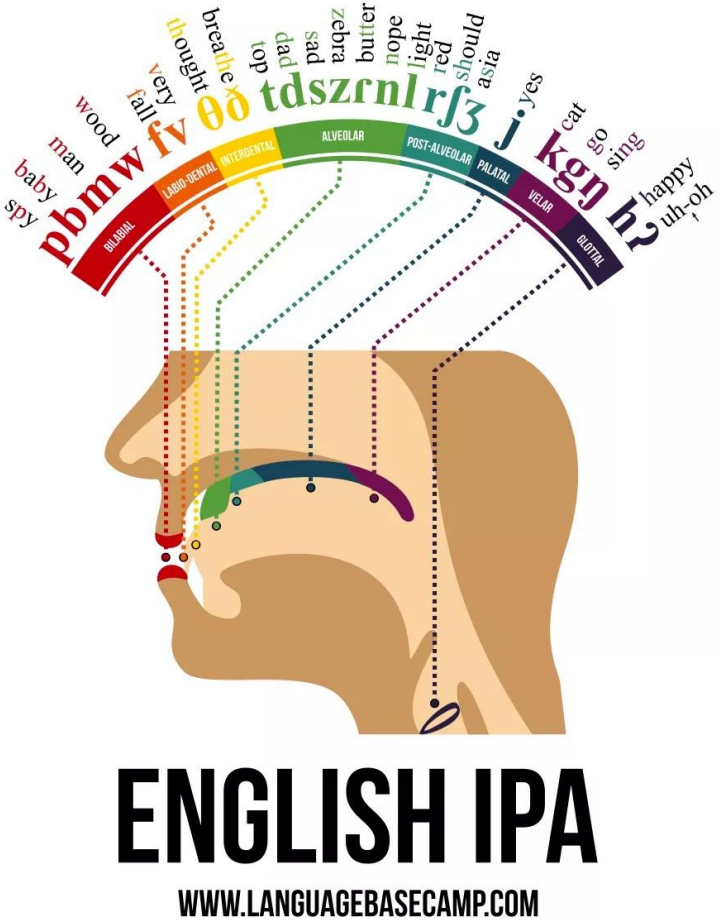


Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- There are some lexicons, such as the CMU dictionary that breaks each word to it's phoneme sequence. Online link
- Use this link to read IPA phoneme sequence

Phonetics

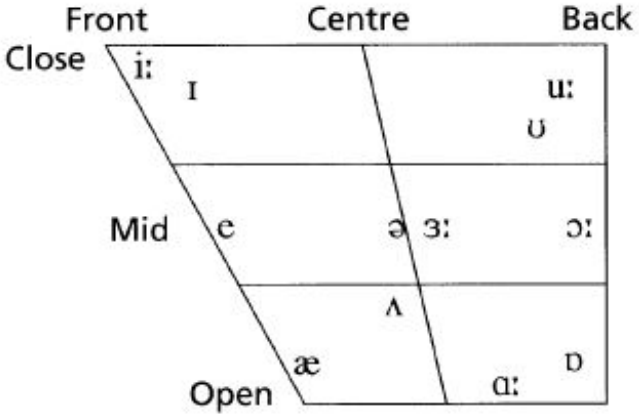
- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- The articulator has a different setup for each phoneme



Source 1

Phonetics

- **Phoneme:** the smallest phonetic unit that has meaning in the language. Represented by slashes by convention.
  - /K/ /IY/ /P/ Vs /D/ /IY/ /P/
- The articulator has a different setup for each phoneme



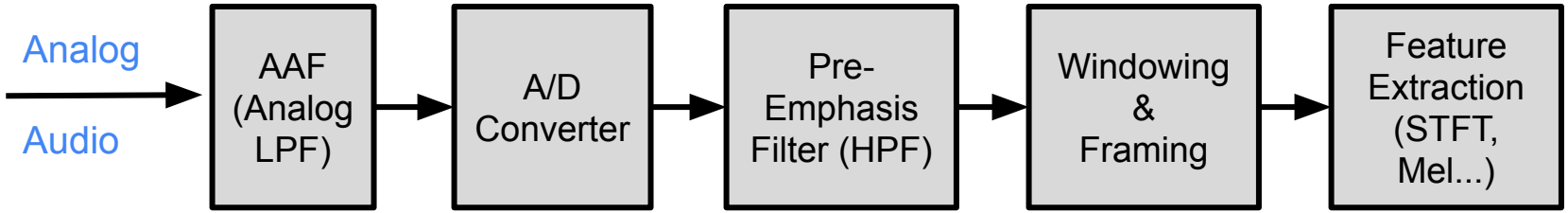
Source 1

Overview

- Motivation
- Communication
- Anatomy & Speech production system
- Phonetics
- **Acoustic/Speech Features**
- Traditional Speech Signals Analysis
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression

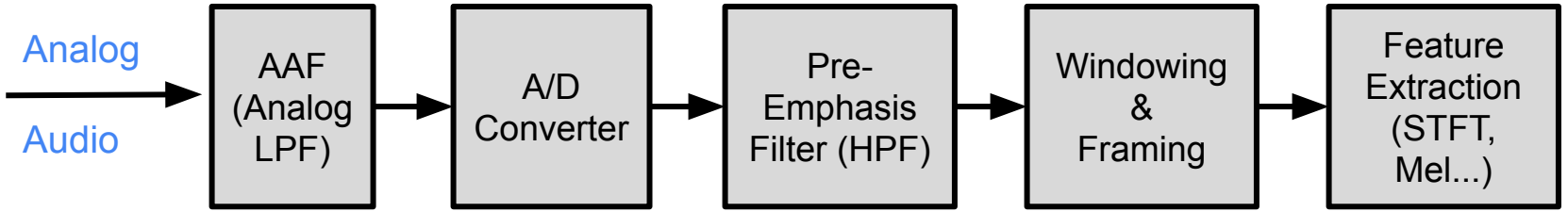
Speech & Audio Signal Processing

Speech Signal Processing- General Pipeline



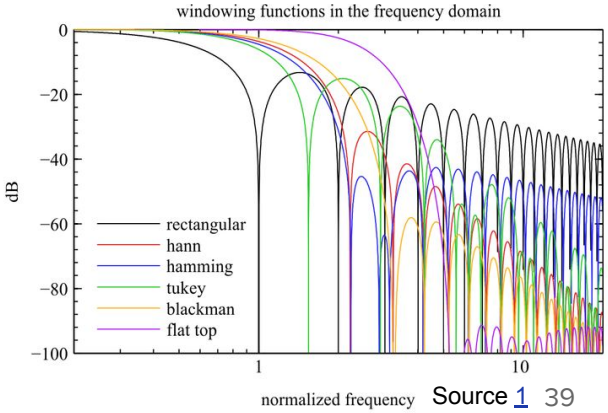
- **Analog audio** input reaches the microphone
- **AAF:** Anti-aliasing filter (AAF) is applied (the analog audio is on infinite frequency response, hence we need to truncate it to a finit one, done in an analog manner because we once we will sample the signal we will have aliasing)
- **A/D convertor:** After applying the AAF, we then convert to digital- both magnitude (amplitude- bits) and sampling (discrete- time).
- **DC Removal:** Usually we subtract mean to remove biases due to acquisition problems

Speech Signal Processing- General Pipeline

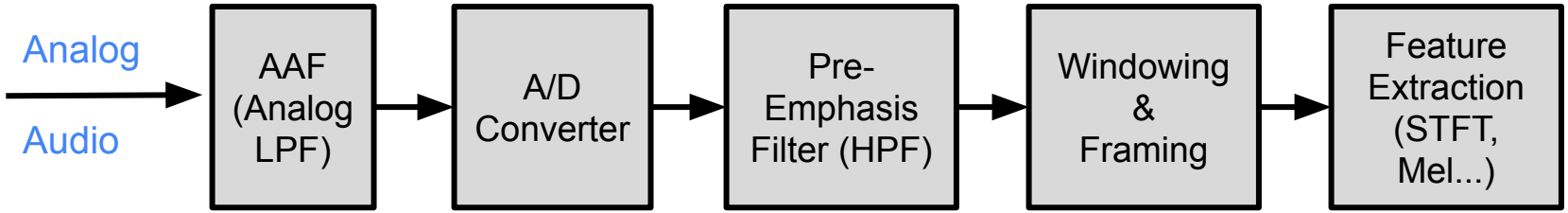


- **Pre-emphasis Filter:** The natural attenuation that arises from voice source is about -12 dB/octave. Pre-emphasis makes higher frequencies of voiced sounds more apparent
- **Windowing:**
  - Non-stationary signals are processed in short frames (assumed stationary) that are overlapping with each other.
  - Typical frame length ~ 20-40 msec, overlapping ~ 30-75 %
- **Framing:** Windowing reduces the effect of the spectral artefacts that arise from discontinuities at the frame endpoints. Typically Hamming window is used

$$H(Z) = 1 - \alpha Z^{-1}$$
$$y[n] = x[n] - \alpha x[n-1]$$
$$\alpha \approx 1$$

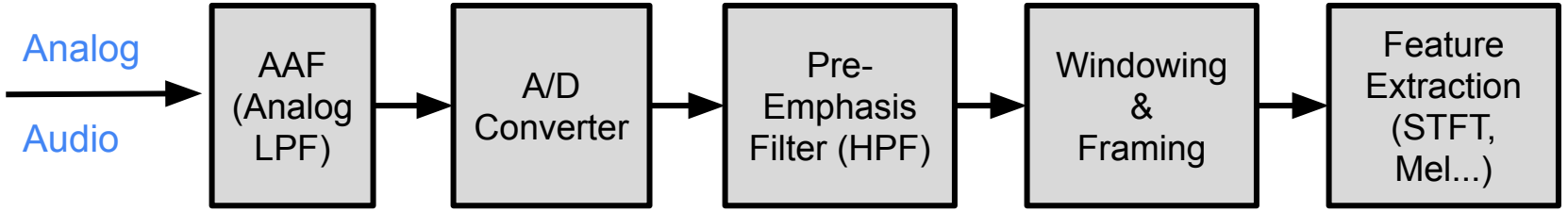


Speech Signal Processing- General Pipeline



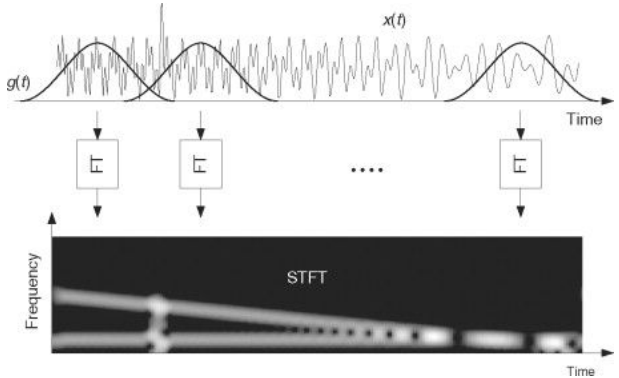
- **Feature extraction:**
  - A link to torch.audio tutorial for playing/manipulating audio

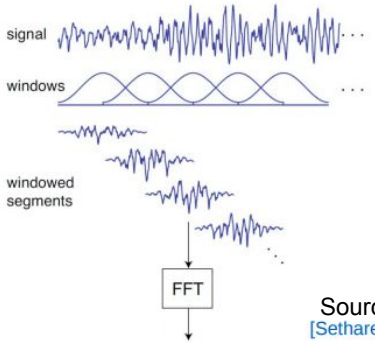
Speech Signal Processing- General Pipeline



- **STFT:** The most well-known approach to time-frequency analysis makes use of nonparametric, Fourier-based spectral analysis applied to each of the short segments- an operation referred to as the Short-Time Fourier Transform (STFT). In this approach, the definition of the Fourier transform is modified so that a sliding time window w[n] is included, and defines each time segment to be analyzed, thus resulting in a two-dimensional function.

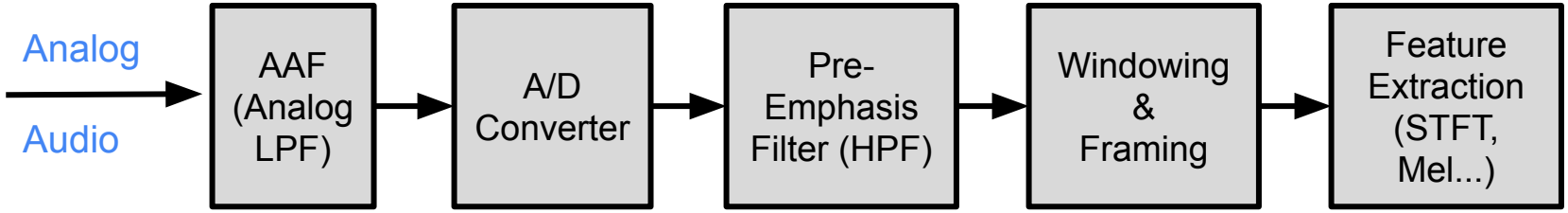
$$X_n\left(e^{j\hat{w}}\right) = \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\hat{w}m}$$



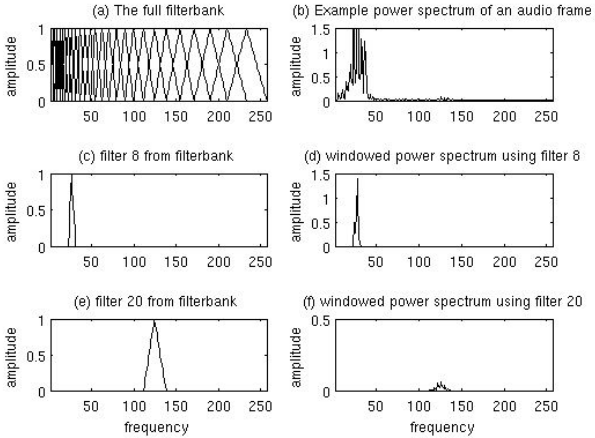


Source 1 [Sethares, 2007]

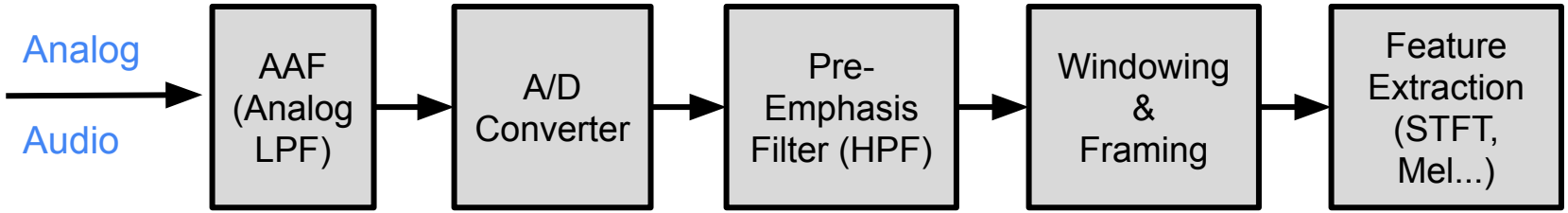
Speech Signal Processing- General Pipeline



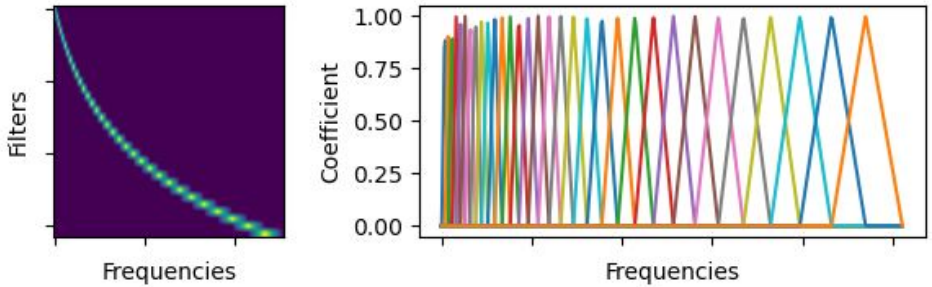
- **Mel Spectrogram:**
  - Linear projects. Where the projection matrix is Nfilters X Mag STFT size.
  - Typical value: 40-80 filters@16KHz.



Speech Signal Processing- General Pipeline

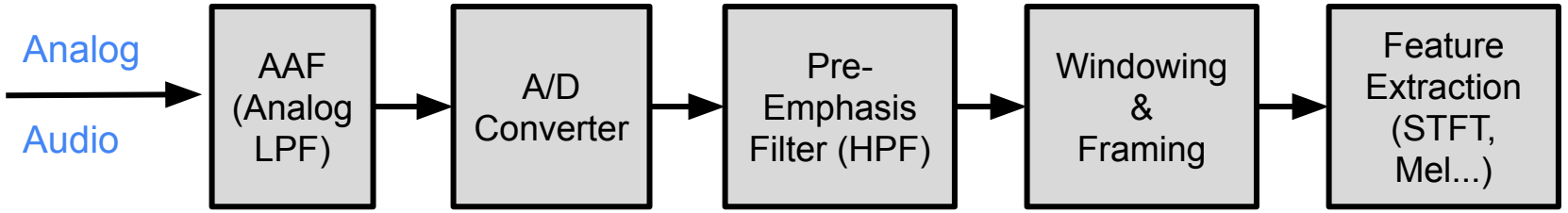


- **Mel Spectrogram:**
  - Linear projects. Where the projection matrix is Nfilters X Mag STFT size.
  - Typical value: 40-80 filters@16KHz.
  - Compression of upper frequencies (according to mel scale- powered by perceptual motivation)

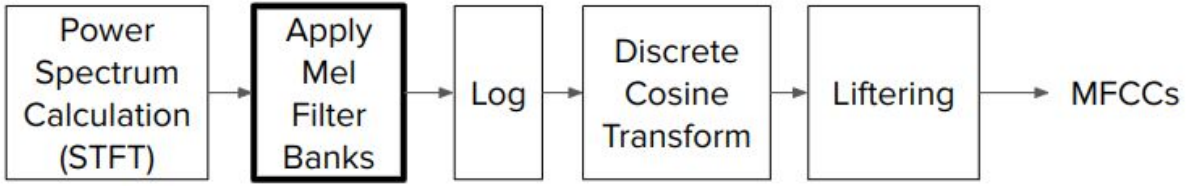


DSP for Audio, Tal Rosenwein

Speech Signal Processing- General Pipeline

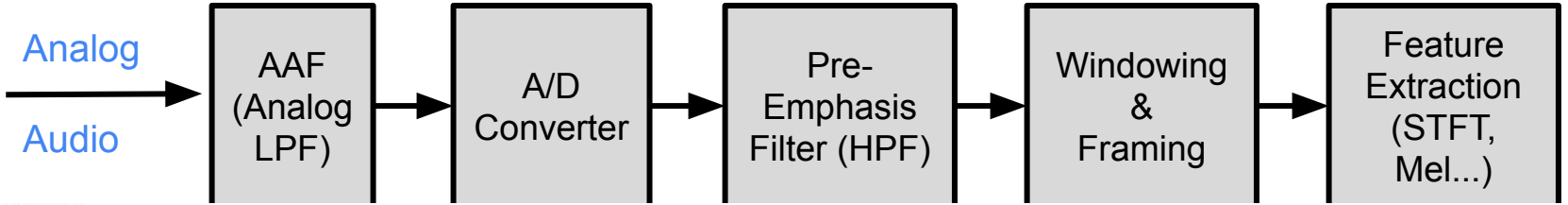


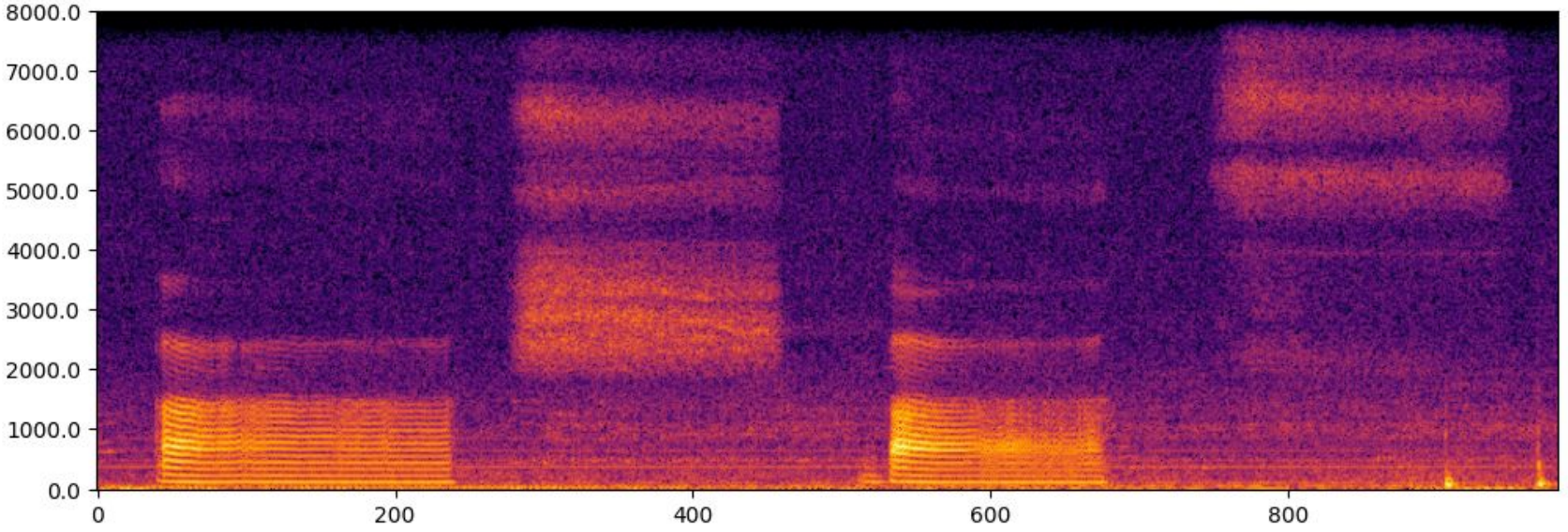
- **MFCCs (Mel Frequency Cepstral Coefficients)**
  - After receiving Mel filter banks, we apply a series of operations and get the MFCCs.
  - Usually 13 coefficients.
  - Along with delta and delta-delta (temporal derivatives) which results in 39 coeffs.
  - They were widely used due to several reasons, such as interpretation. Will not elaborate as they are currently not in use anymore (only in HuBERT for initialization of the clusters during pre-training).



DSP for Audio, Tal Rosenwein

Speech Signal Processing- General Pipeline



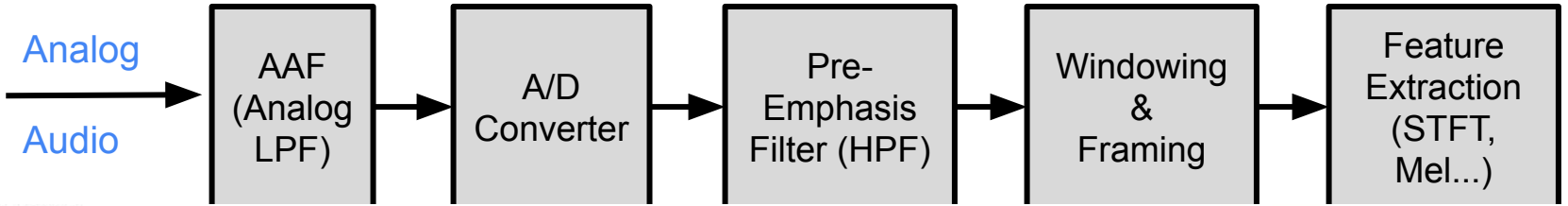


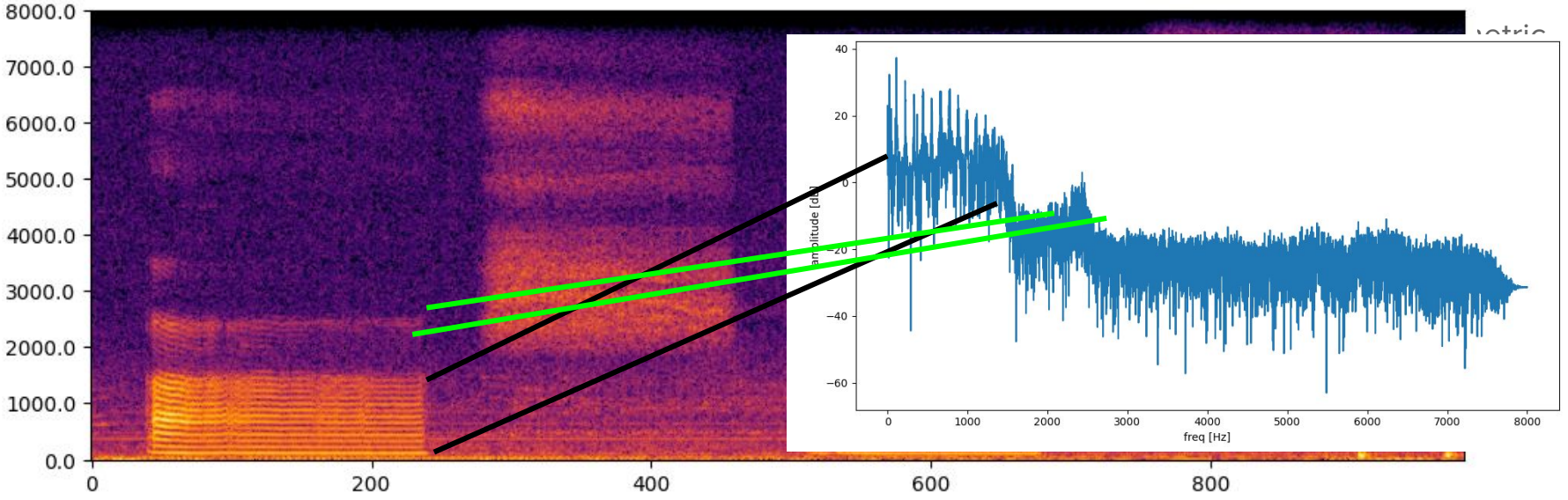
ıetric,
to as
ɔrm is
to be

http
shc
http
TF

[Sethares, 2007]

Speech Signal Processing- General Pipeline





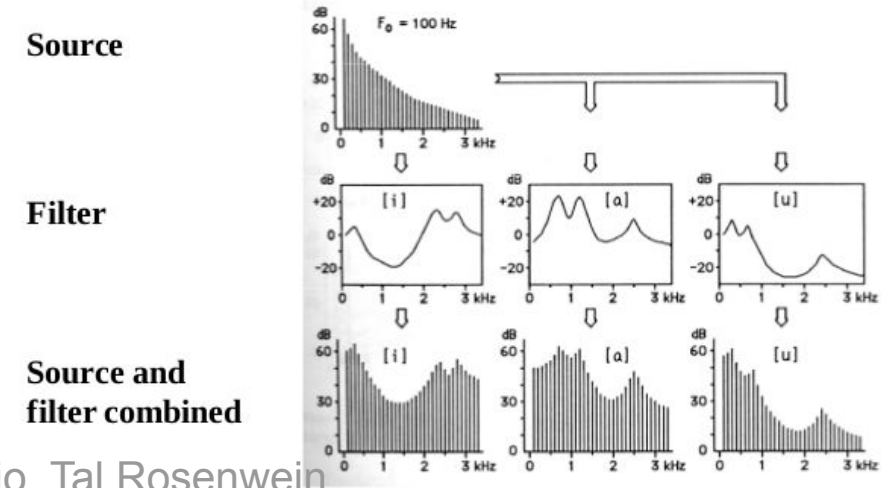
http
shc
http
TF

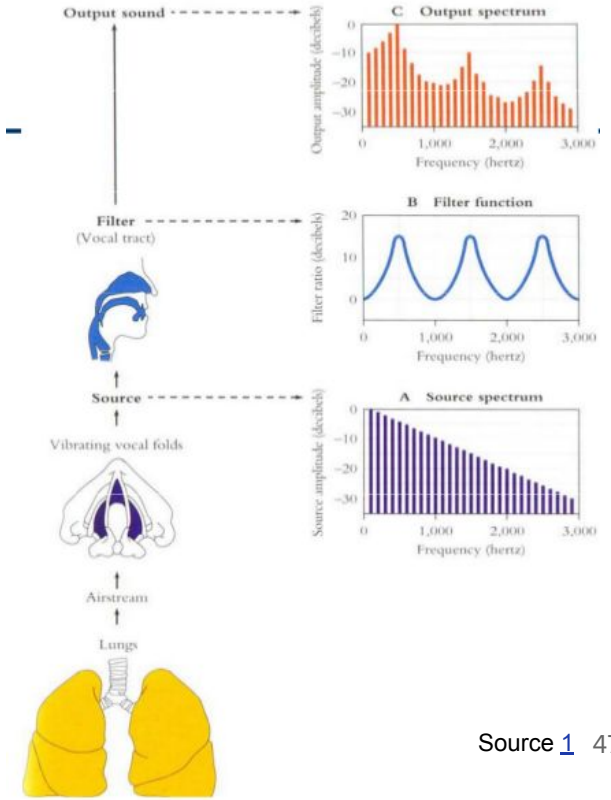
[Sethares, 2007]

Speech Production- Source-Filter Model

- The resulting speech spectrum S(z) is a combination of the source spectrum U(z) and the vocal tract transfer function H(z):

$S(z) = U(z)H(z)$	$\leftrightarrow$	$s[n] = u[n] * h[n]$
Frequency domain		Time domain



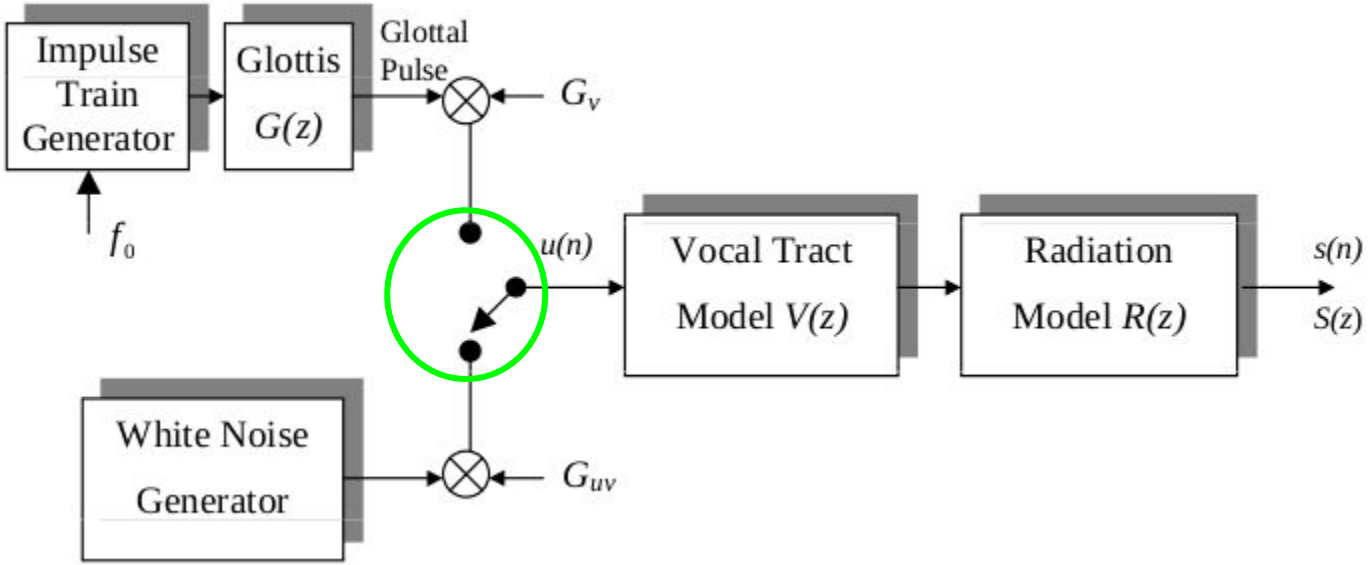


Source 1

Speech Production- Source-Filter Model

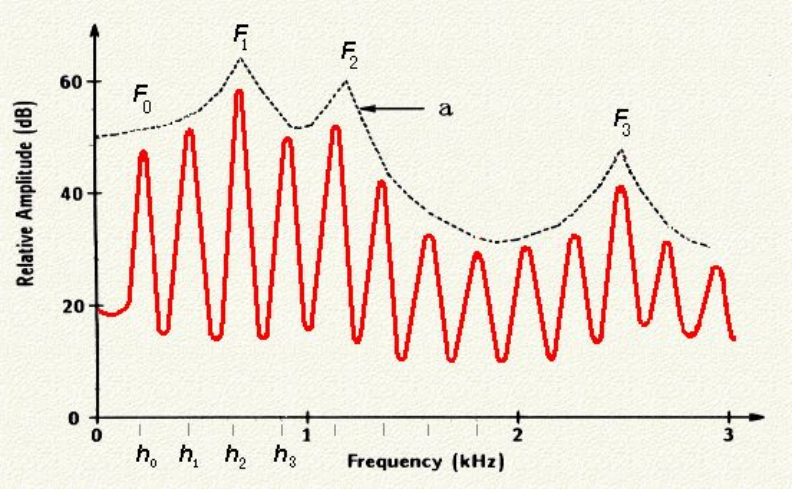
Voiced

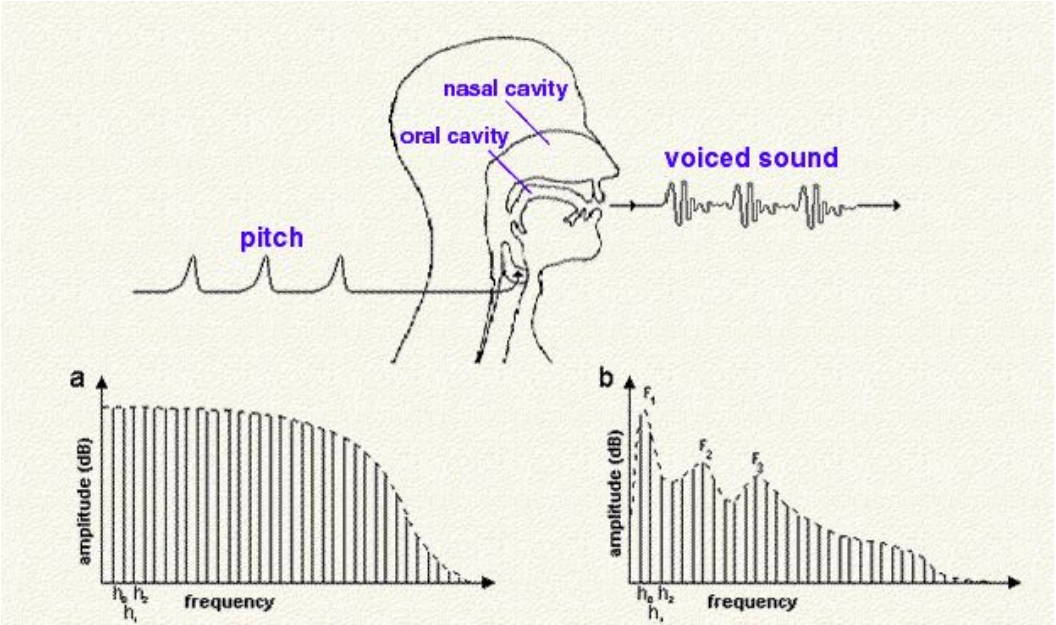
Un-Voiced



Speech Signals Analysis

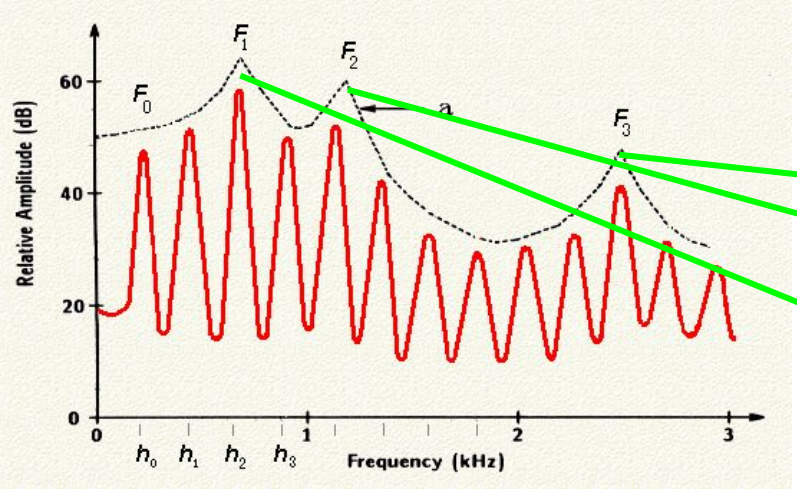
- Pitch, Formants and Spectral Envelope

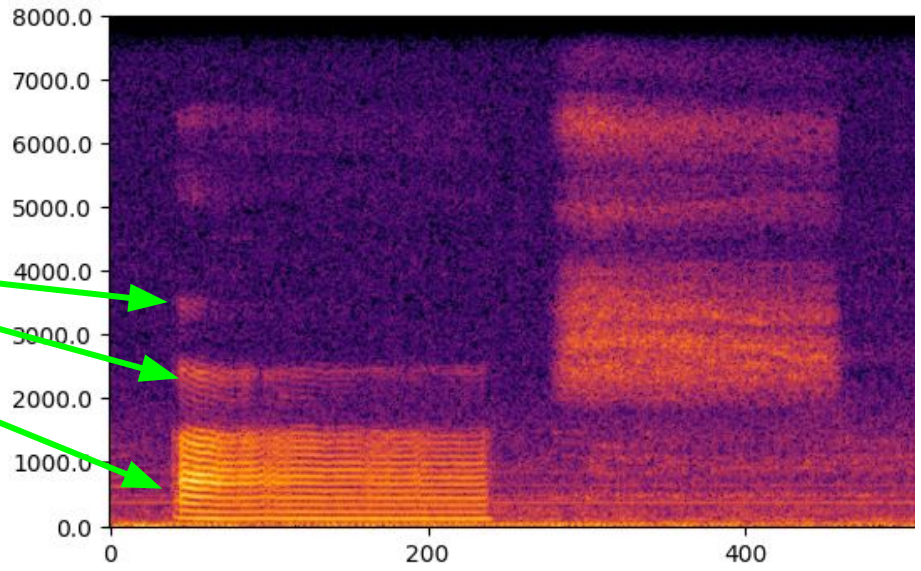




Speech Signals Analysis

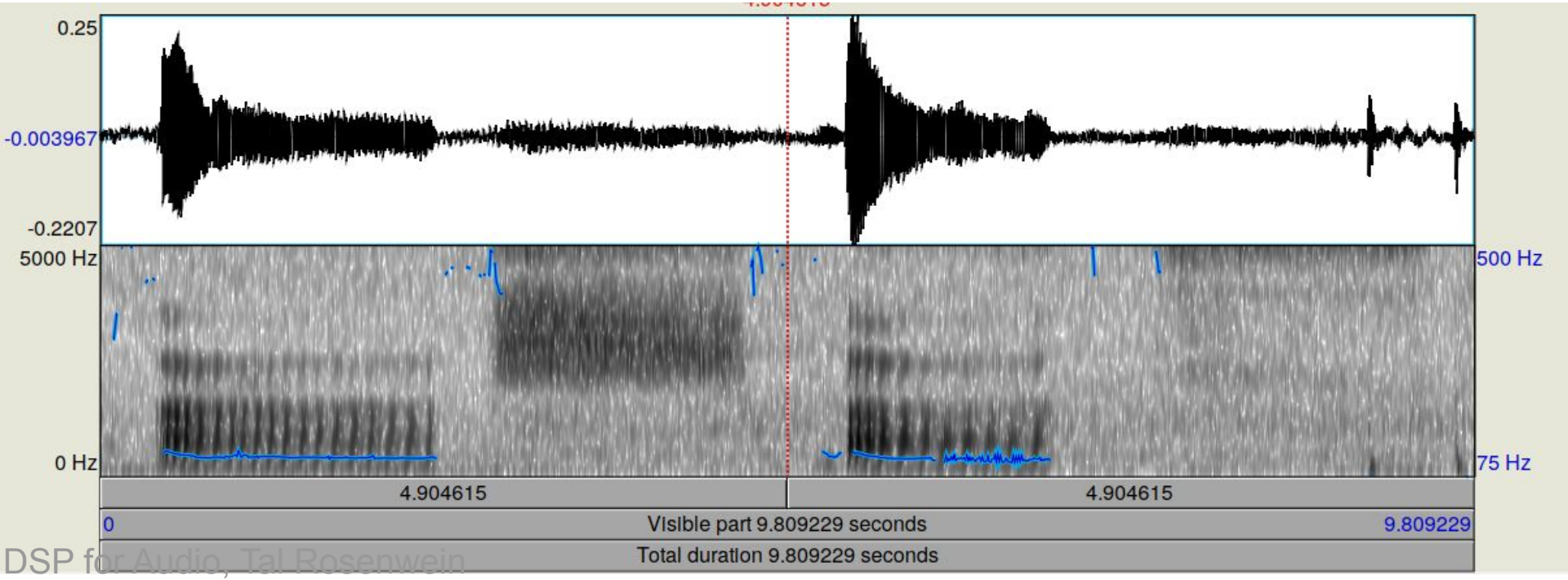
- Pitch, Formants and Spectral Envelope





Speech Signals Analysis: Pitch Detection

- Various algorithms, such as autocorrelation



Speech Signals Analysis: Spectral Envelope

- Linear Predictive Coding (LPC)
- Parametric method for representing the **spectral envelope** in a compressed from (was the basis of VoIP).The process is done on frames due to non-stationarity property of the speech signal.

Assuming that the stationary signal s[n], where n=1,2,...,N, is the output of the system which can be described by (Eq. 1.14) [3]

$$s[n] = -\sum_{i=1}^{p} a_i^* s[n-i] + G^* u[n] \qquad 1.14$$

Where a_i^* are the linear predictive coefficients which generated the system, u[n] is the input in the n'th sample and G^* is the gain of the system.

Speech Signals Analysis: Spectral Envelope

- Linear Predictive Coding (LPC)
- Parametric method for representing the **spectral envelope** in a compressed from (was the basis of VoIP).
- The process is done on frames due to non-stationarity property of the speech signal.

$$s[n] = -\sum_{i=1}^{p} a_i^* s[n-i] + G^* u[n] \qquad 1.14$$

From Eq. 1.14 one can see that s[n] is given a linear combination of its past values summed with the input of the system u[n]. Applying the Z transform on both sides of Eq. 1.14 yields (Eq. 1.15)

$$H(z) = \frac{S(z)}{U(z)} = \frac{G^*}{1 + \sum_{i=1}^{p} a_i^* z^{-i}} \qquad 1.15$$

Speech Signals Analysis: Spectral Envelope

- Linear Predictive Coding (LPC)
- Parametric method for representing the **spectral envelope** in a compressed from (was the basis of VoIP).The process is done on frames due to non-stationarity property of the speech signal.

We expect that as the model order will increase the prediction error will decrease (Eq. 1.22).

$$\lim_{p\to\infty} E[e^2[n]] = 0 \qquad 1.22$$

Though, as we increase the number of parameters, we will get an estimation which is closer to the real values. This can become a disadvantage when we want to model "typical pattern" of the phonemes. Since this model will be modeling the non-relevant artifacts as well (including the Pitch). Therefore we would like to have model order that emphasizes the relevant formants.

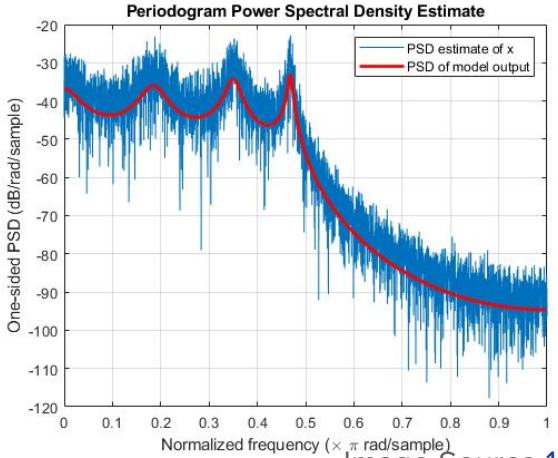
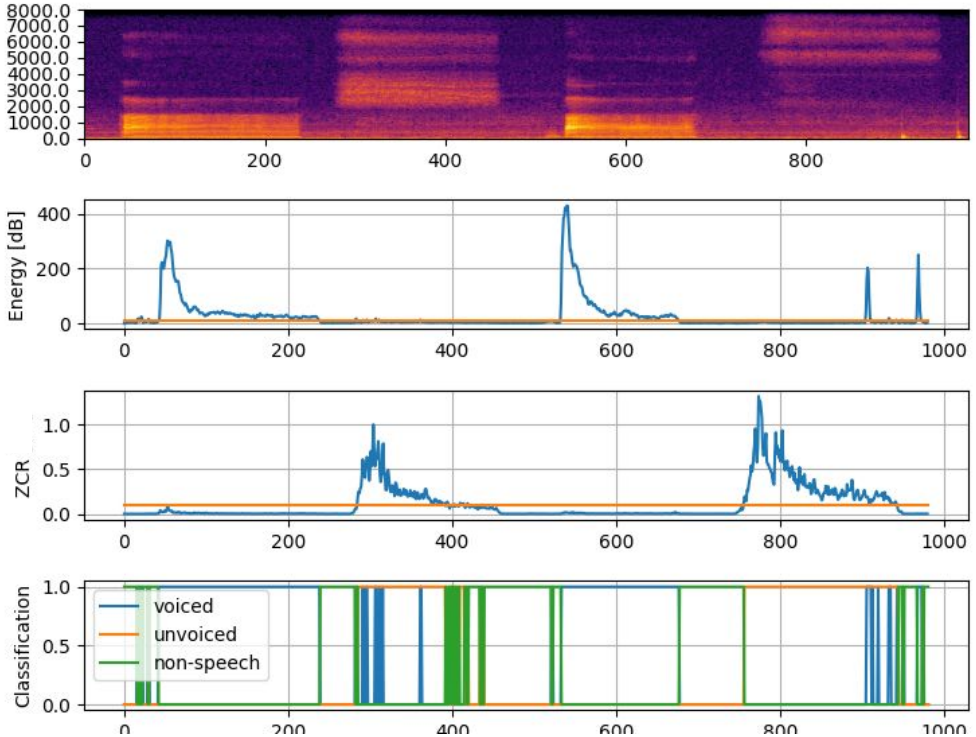


Image Source 1

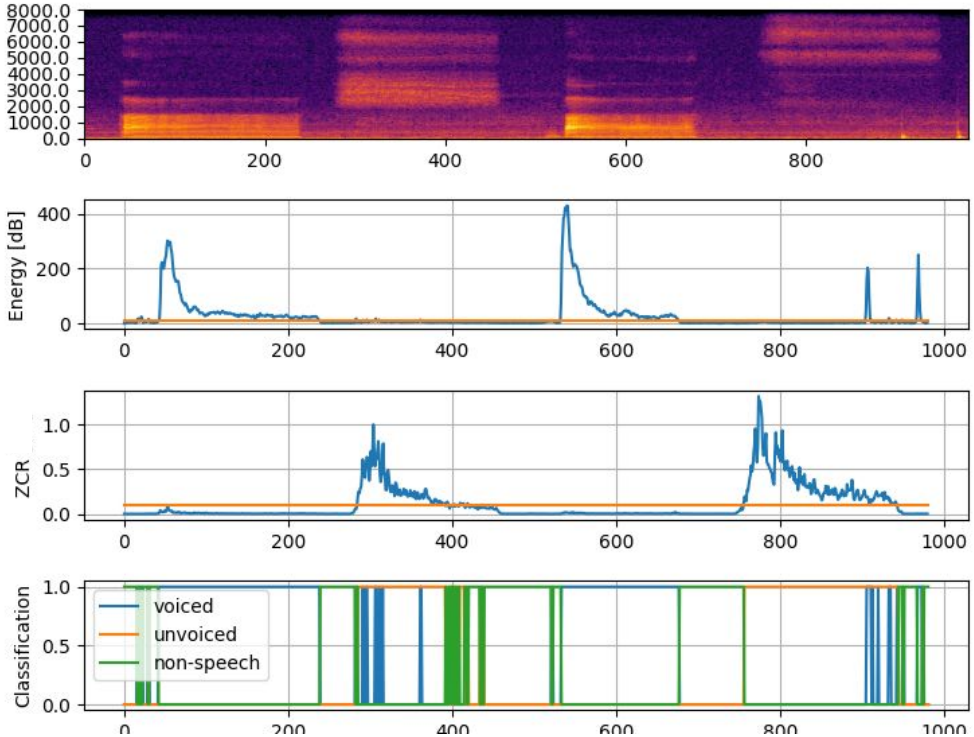
Speech Signals Analysis: Classification

- Naive Classification:
  - Speech Vs non-speech: Energy
  - Unvoiced: Zero Crossing Rate (ZCR)
  - Voiced: The rest



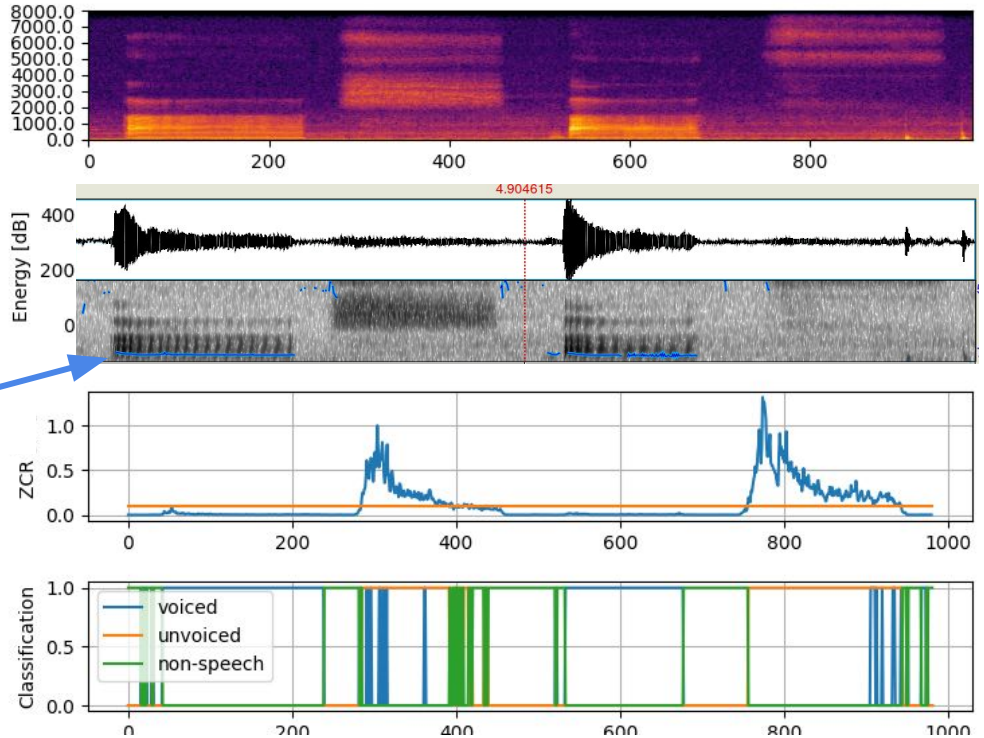
Speech Signals Analysis: Classification

- Naive Classification:
  - Speech Vs non-speech: Energy
  - Unvoiced: Zero Crossing Rate (ZCR)
  - Voiced: The rest
- But what happens in noisy environments:
  - Pitch- voiced



Speech Signals Analysis: Classification

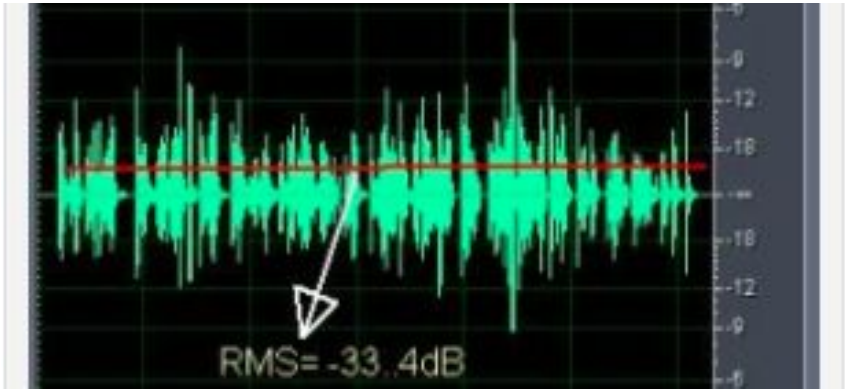
- Naive Classification:
  - Speech Vs non-speech: Energy
  - Unvoiced: Zero Crossing Rate (ZCR)
  - Voiced: The rest
- But what happens in noisy environments:
  - Pitch- voiced



Speech Signals Analysis: Energy & RMS

- Sometimes we want to represent some “trends” of signal, but its average has a (zero) constant value.
- 2 accepted measures for an ‘alternative’ average: RMS and Energy
  - RMS: Root-Mean-Squared. Most books define this as the “amount of AC power that produces the same heating effect as an equivalent DC power”
  - Energy: defined as the area under the sq
- Usually calculated in **chunks with hop**.

$$E_s = \sum_{n=-\infty}^{\infty} |x(n)|^2 \qquad RMS = \sqrt{\frac{1}{n}\sum_i x_i^2}$$



Overview

- Motivation
- Communication
- Anatomy & Speech production system
- Phonetics
- Acoustic/Speech Features
- **Traditional Speech Signals Analysis**
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression

Applications

Applications

Challenges:

1. Delayed signal / packet lost - Time stretching
2. Noisy environment - Noise reduction
3. Speakers with multiple gains - Auto gain control
4. Audio Clipping - Audio Compression
5. Connectivity issues - Variable bitrate vocoders

Image Source

Applications

Challenges:

1. **Delayed signal / packet lost -** **Time stretching**
2. Noisy environment - Noise reduction
3. Speakers with multiple gains - Auto gain control
4. Audio Clipping - Audio Compression
5. Connectivity issues - Variable bitrate vocoders

Image Source

Applications: Time Stretching

- Many applications require to align input and output audio, as in soundtracks. Or we want to hear a YouTube video in a different rate than it was recorded.
- Hence, we would like to change the speed of the wav file
- Doing the naive way:

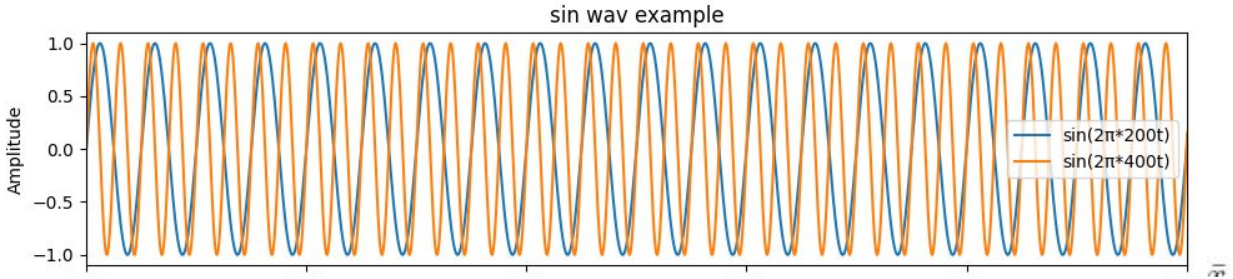
$$f(x) \rightarrow \Im(w) = \int_{-\infty}^{\infty} f(x)e^{-jwx}dx$$

$$f(ax) \rightarrow \frac{1}{|a|}\Im(\frac{w}{a})$$

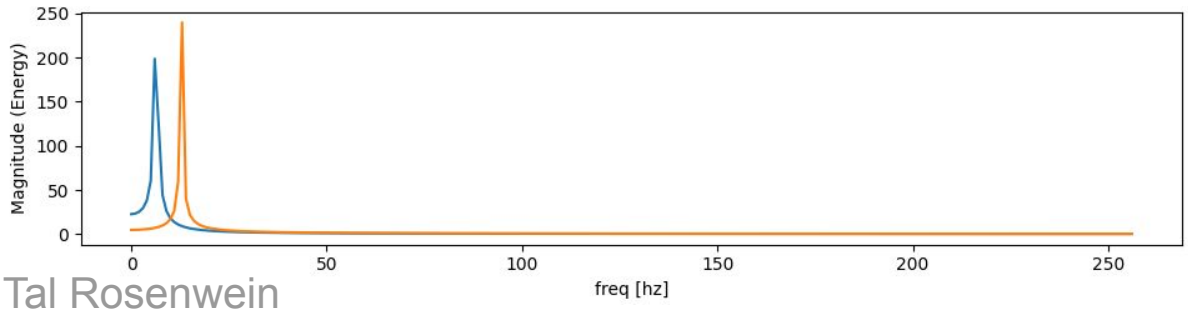
$$\Im(f(ax)) = \int_{-\infty}^{\infty} f(ax)e^{-jwx}dx = |\bar{x} = ax, dx = \frac{d\bar{x}}{a}| = \int_{-\infty}^{\infty} f(\bar{x})e^{-j\frac{w}{a}\bar{x}}\frac{1}{a}d\bar{x}$$

Applications: Time Stretching

- Doing the naive way will change the pitch

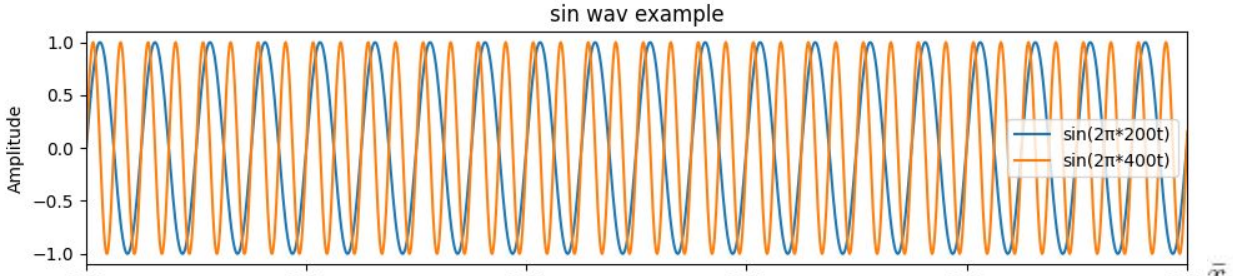


Speeding up the audio → period is shorter → increasing frequency

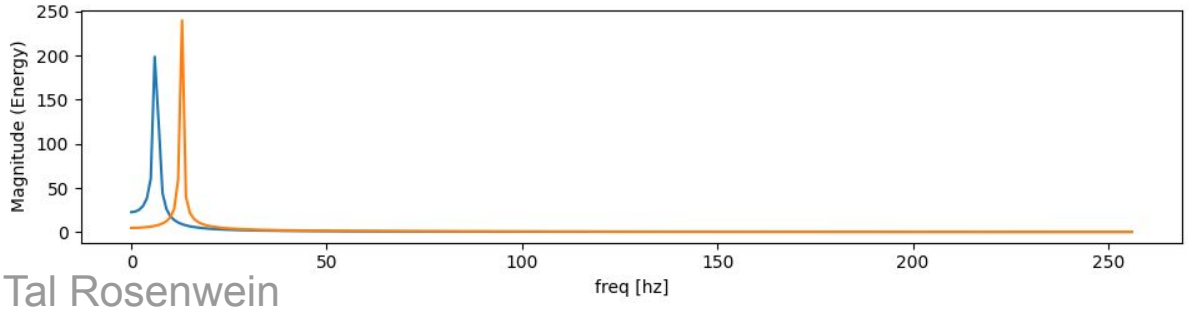


Applications: Time Stretching

- Doing the naive way will change the pitch

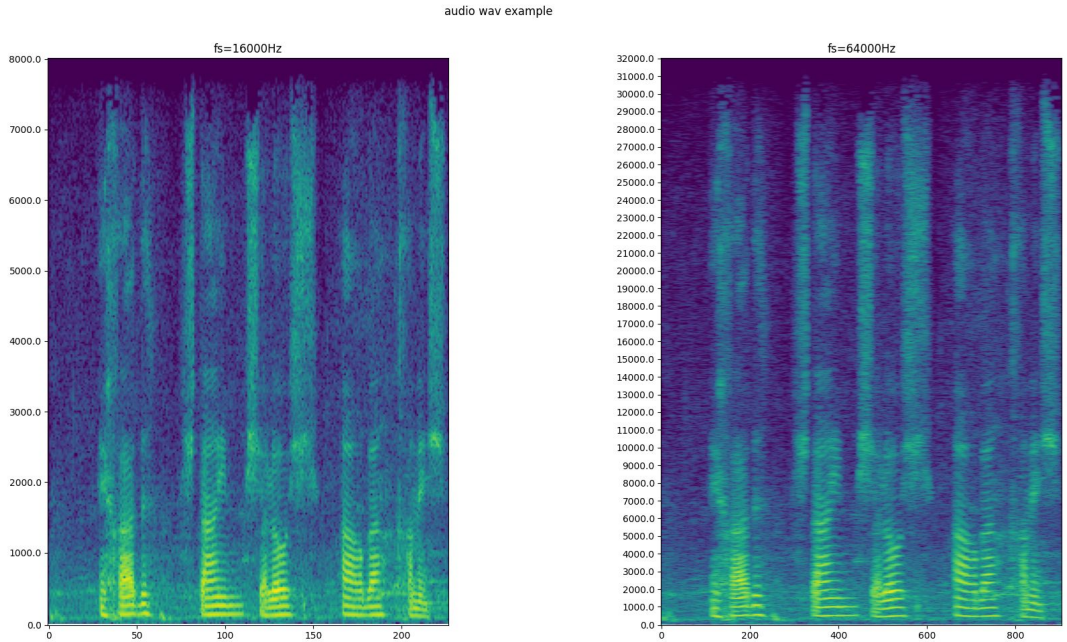


Speeding up the audio → period is shorter → increasing frequency



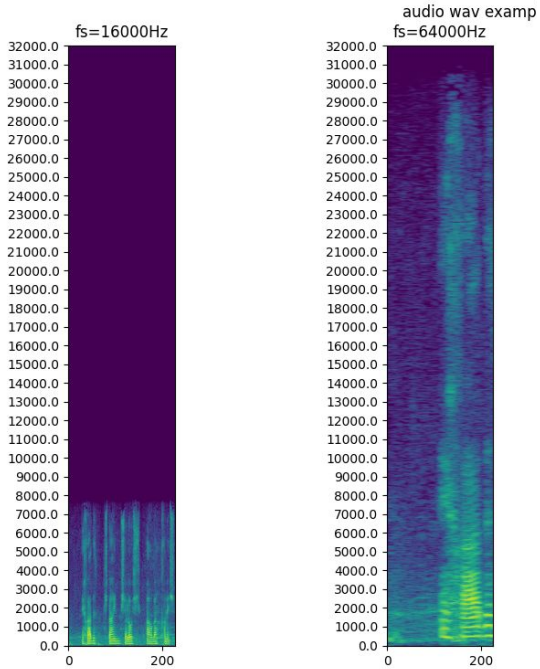
Applications: Time Stretching

- Doing the naive way will change the pitch



Applications: Time Stretching

- Doing the naive way will change the pitch



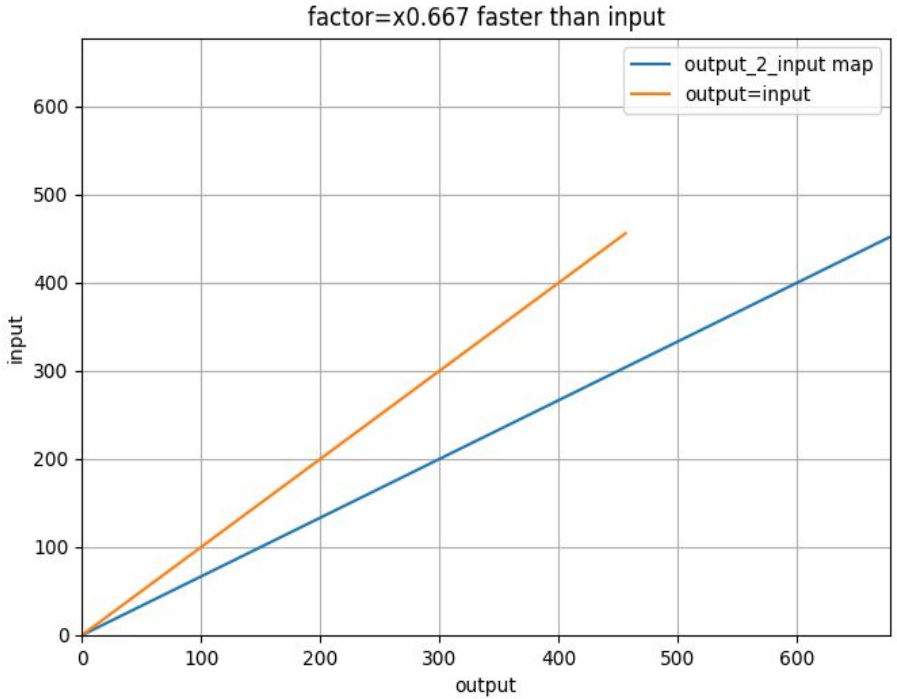
Applications: Time Stretching

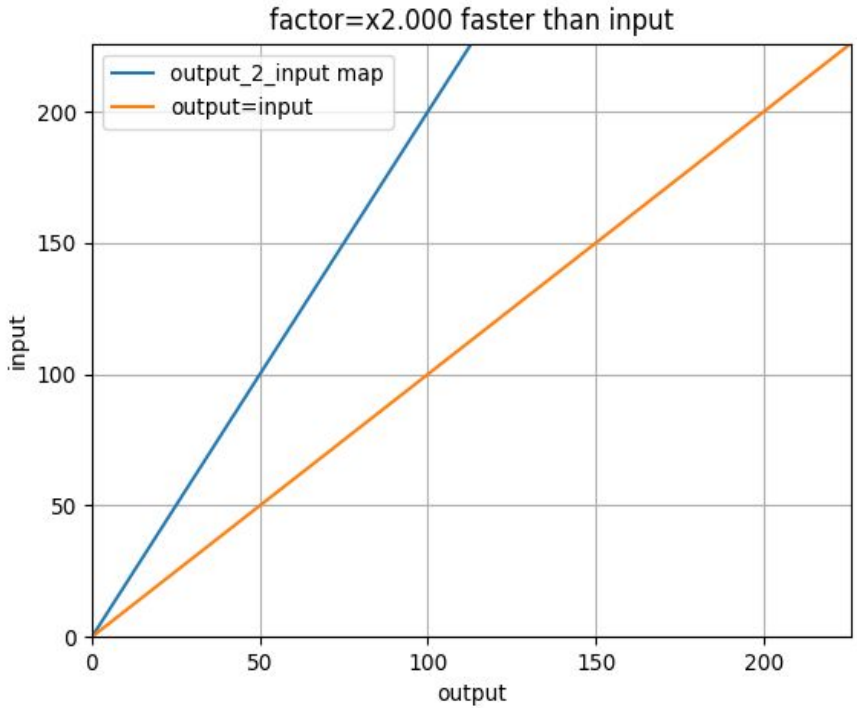
- We need to find **an approximation** that will stretch while keeping the pitch without too much distortion.

Applications: Time Stretching

- We need to find **an approximation** that will stretch while keeping the pitch without too much distortion.
- **Phase Vocoder**
  - **Set the alignment between input and output times**
    - Must be monotonic w.r.t the input
    - Can be fixed / non-fixed alignment

Applications: Time Stretching



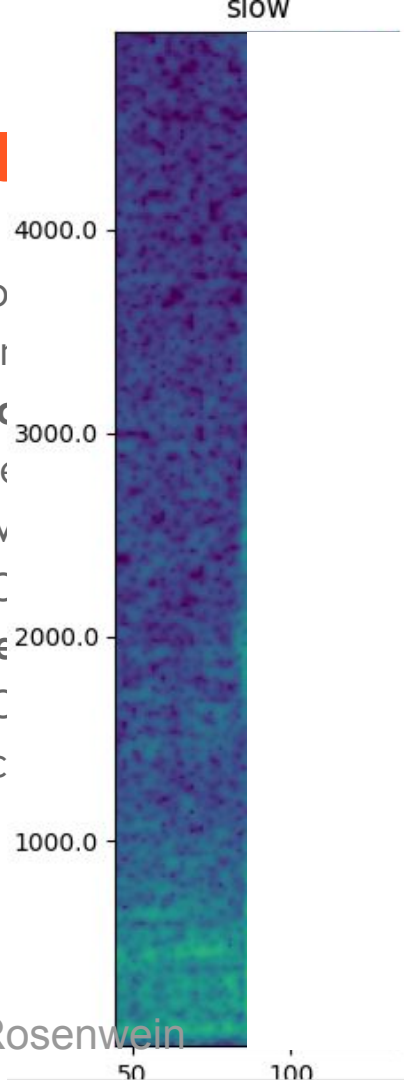


Applications: Time Stretching

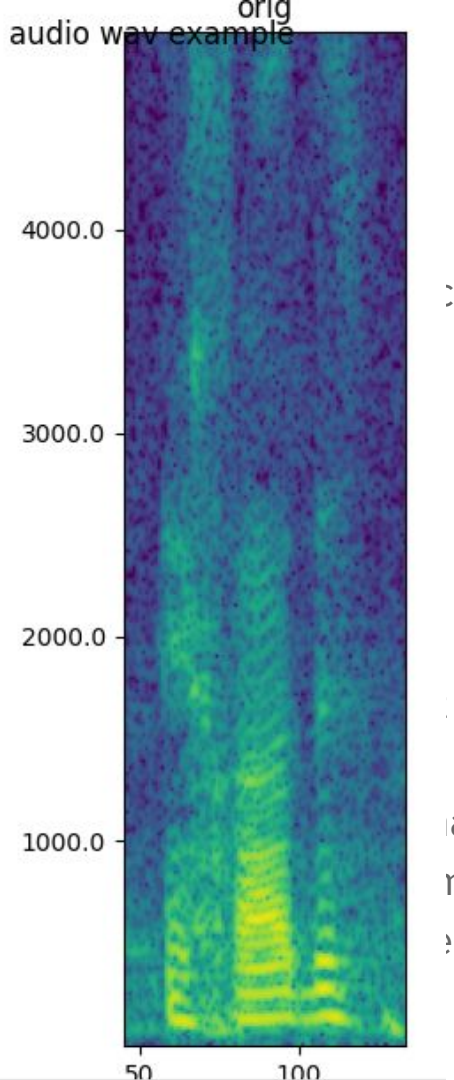
- We need to find **an approximation** that will stretch while keeping the pitch without too much distortion.
- **Phase Vocoder** (phase_vocoder_variable_example)
  - Set the alignment between input and output times
    - Must be monotonic w.r.t the input
    - Can be fixed / non-fixed alignment
  - **Iterate over the output times:**
    - Current value indicates what is the input index that the current output should consider.
      - **Mag:**Find nearest neighbour within the input vector for that time, and interpolation between adjacent frames in the spectrogram

Applicat

- We need to
  much distor
- **Phase Voc**
  - Set the
    - M
    - C
  - **Iterate**
    - C
      c



out_t=90



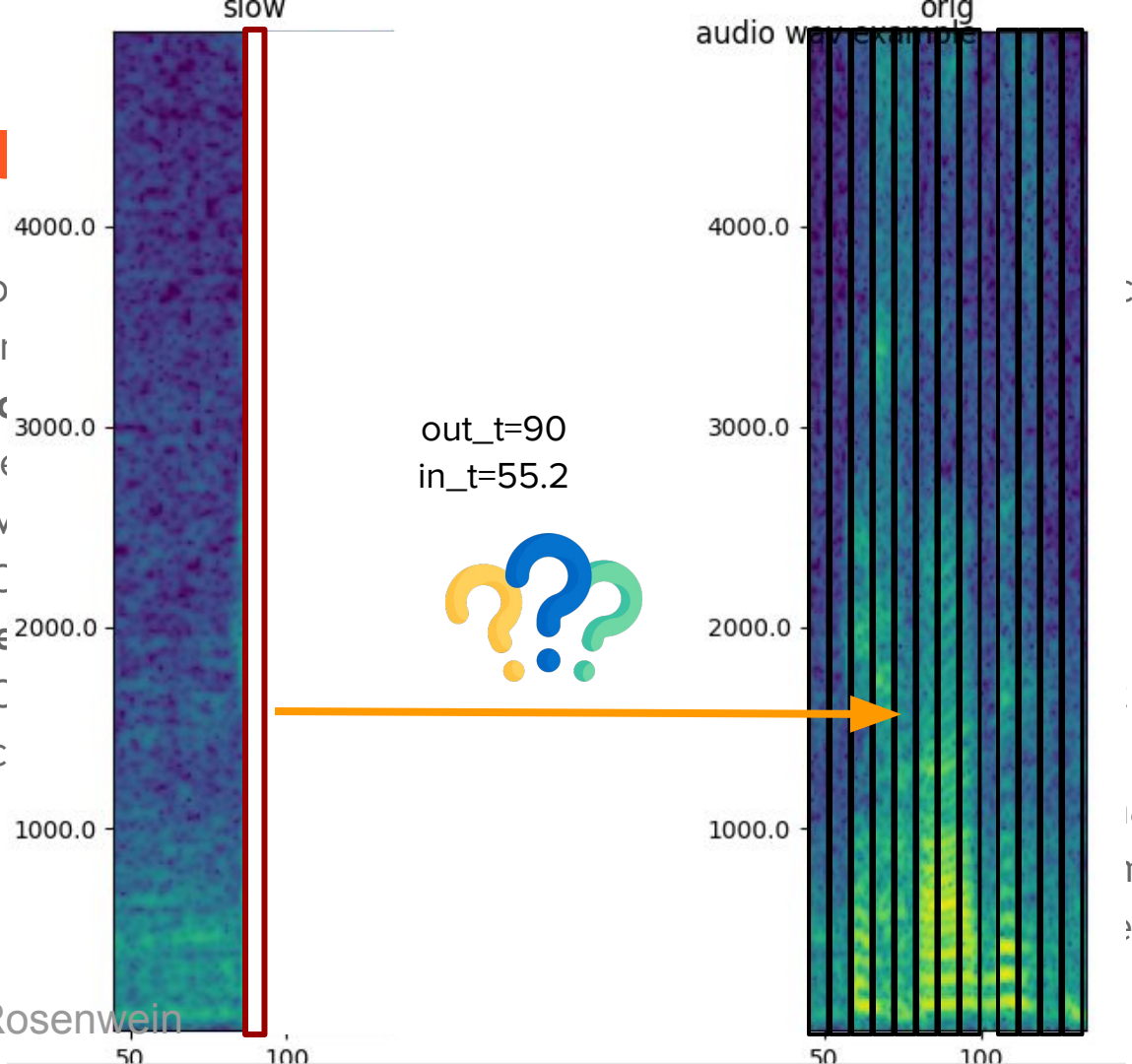
ch without too

output should

at time, and
n
ighbours values.

Applicat

- We need to
  much distor
- **Phase Voc**
  - Set the
    - M
    - C
  - **Iterate**
    - C
      c



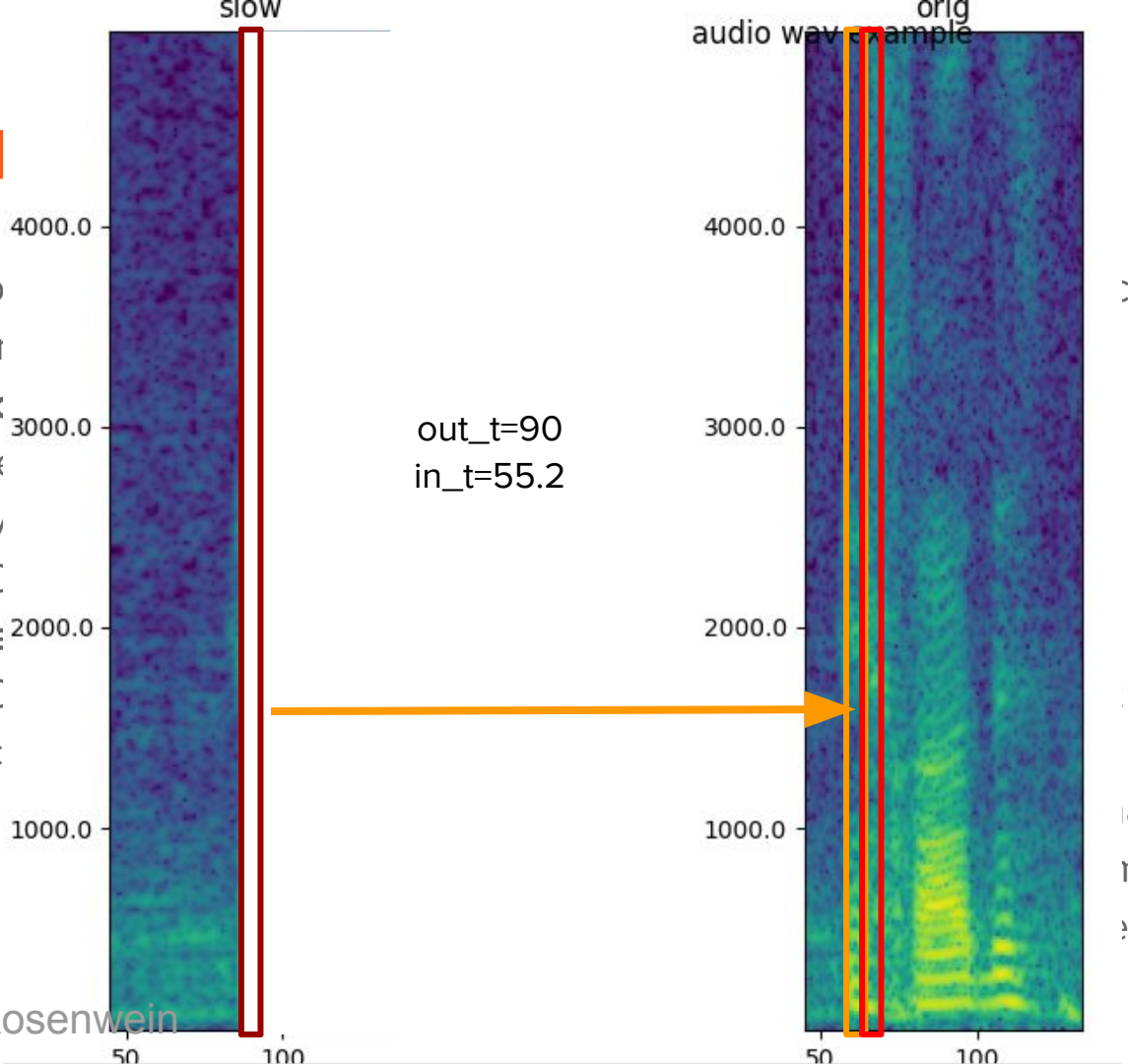
ch without too

output should

at time, and
n
ighbours values.

Applicat

- We need to
  much distor
- **Phase Voc**
  - Set the
    - M
    - C
  - **Iterate**
    - C
      c



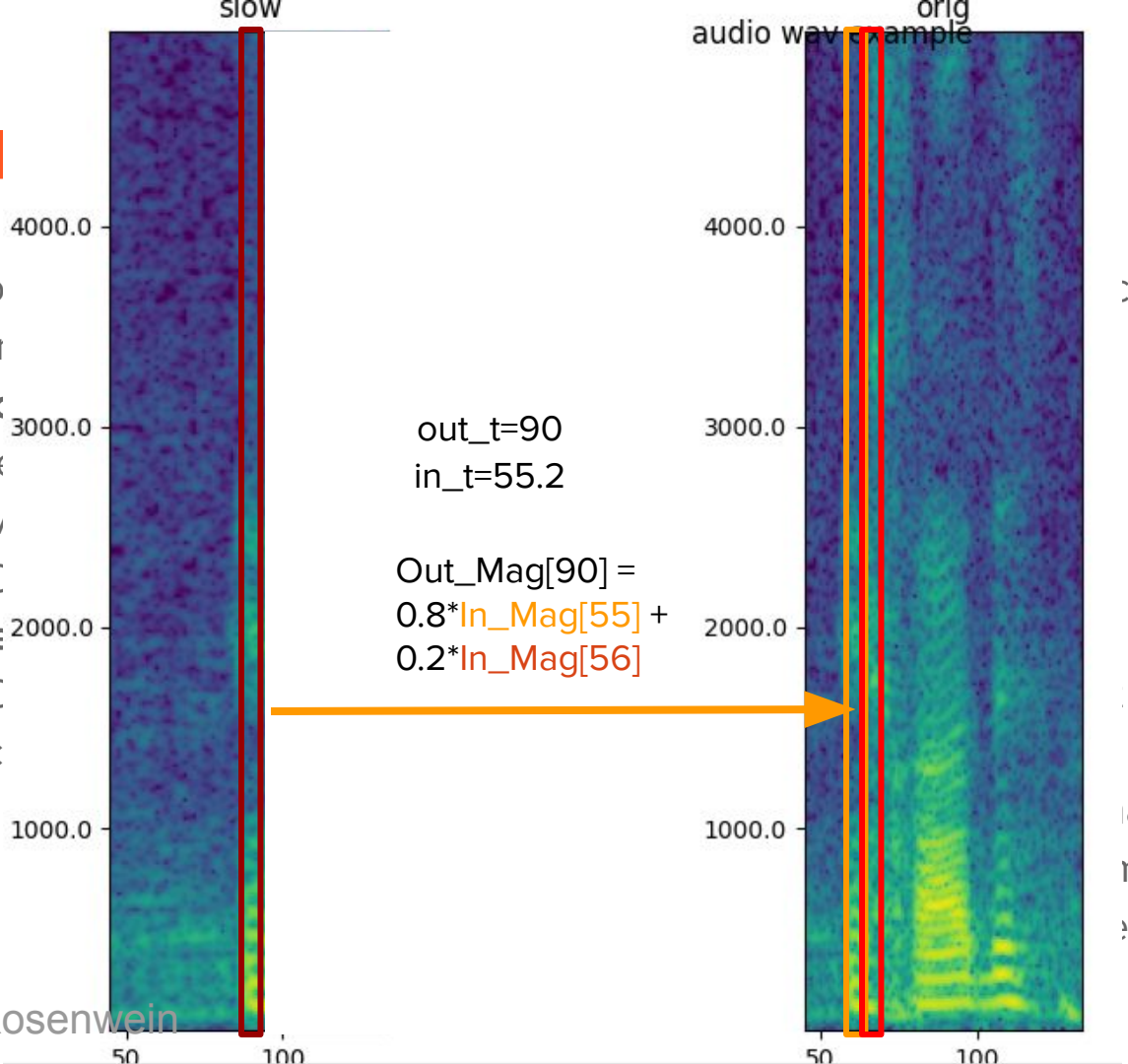
ch without too

output should

at time, and
n
ighbours values.

Applicat

- We need to
  much distor
- **Phase Voc**
  - Set the
    - M
    - C
  - **Iterate**
    - C
      c



ch without too

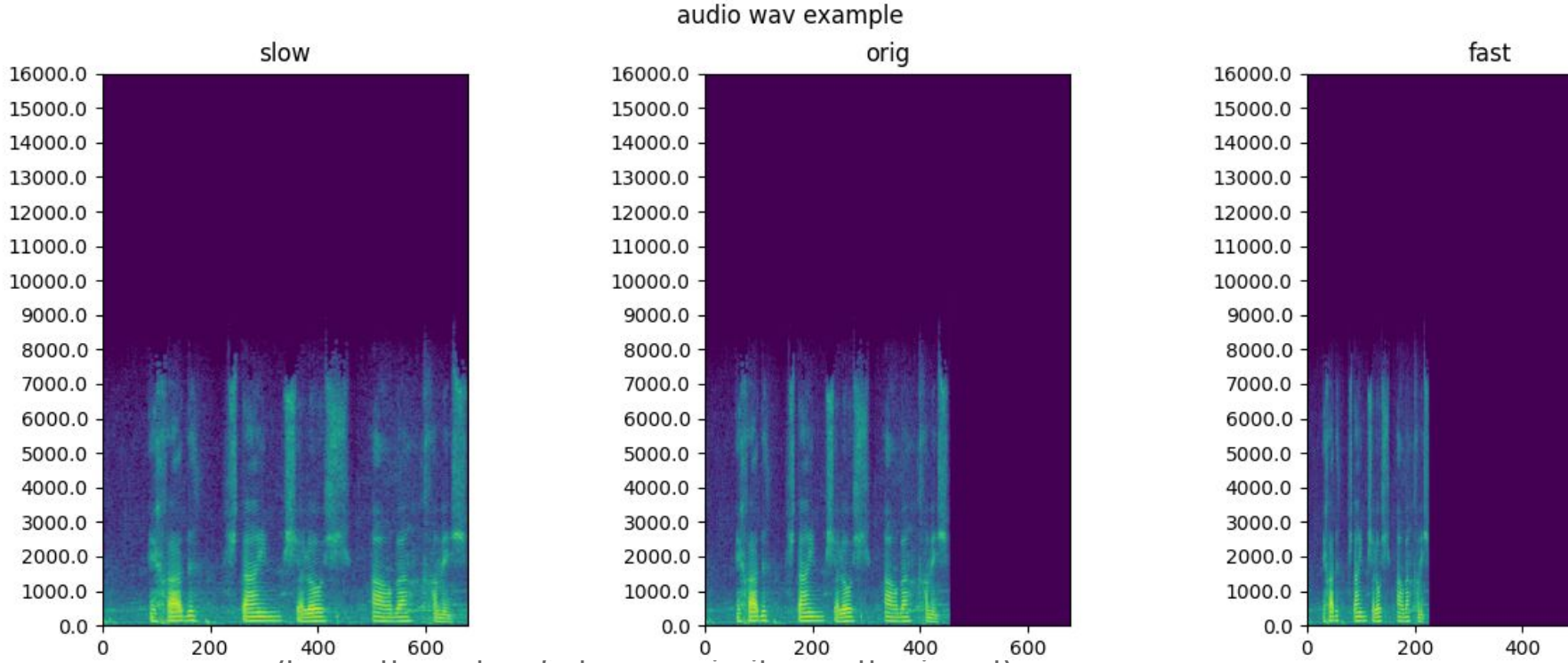
output should

at time, and
n
ighbours values.

Applications: Time Stretching

- We need to find **an approximation** that will stretch while keeping the pitch without too much distortion.
- **Phase Vocoder** (phase_vocoder_variable_example)
  - Set the alignment between input and output times
    - Must be monotonic w.r.t the input
    - Can be fixed / non-fixed alignment
  - **Iterate over the output times:**
    - Current value indicates what is the input index that the current output should consider.
      - **Mag:**Find nearest neighbour within the input vector for that time, and interpolation between adjacent frames in the spectrogram
      - **Phase:** Add prev value to the diff between the nearest neighbours values. (keep the rate of change similar as the input)
      - Append to prev value to form the output STFT.

Applications: Time Stretching



- (keep the rate of change similar as the input)
- Append to prev value to form the output STFT.

Applications: Time Stretching

- We need to find **an approximation** that will stretch while keeping the pitch without too much distortion.
- **Phase Vocoder**
  - Example
- Learnable methods (ScalerGAN)

Applications

Challenges:

1. Delayed signal / packet lost - Time stretching
2. **Noisy environment - Noise reduction**
3. Speakers with multiple gains - Auto gain control
4. Audio Clipping - Audio Compression
5. Connectivity issues - Variable bitrate vocoders

Image Source

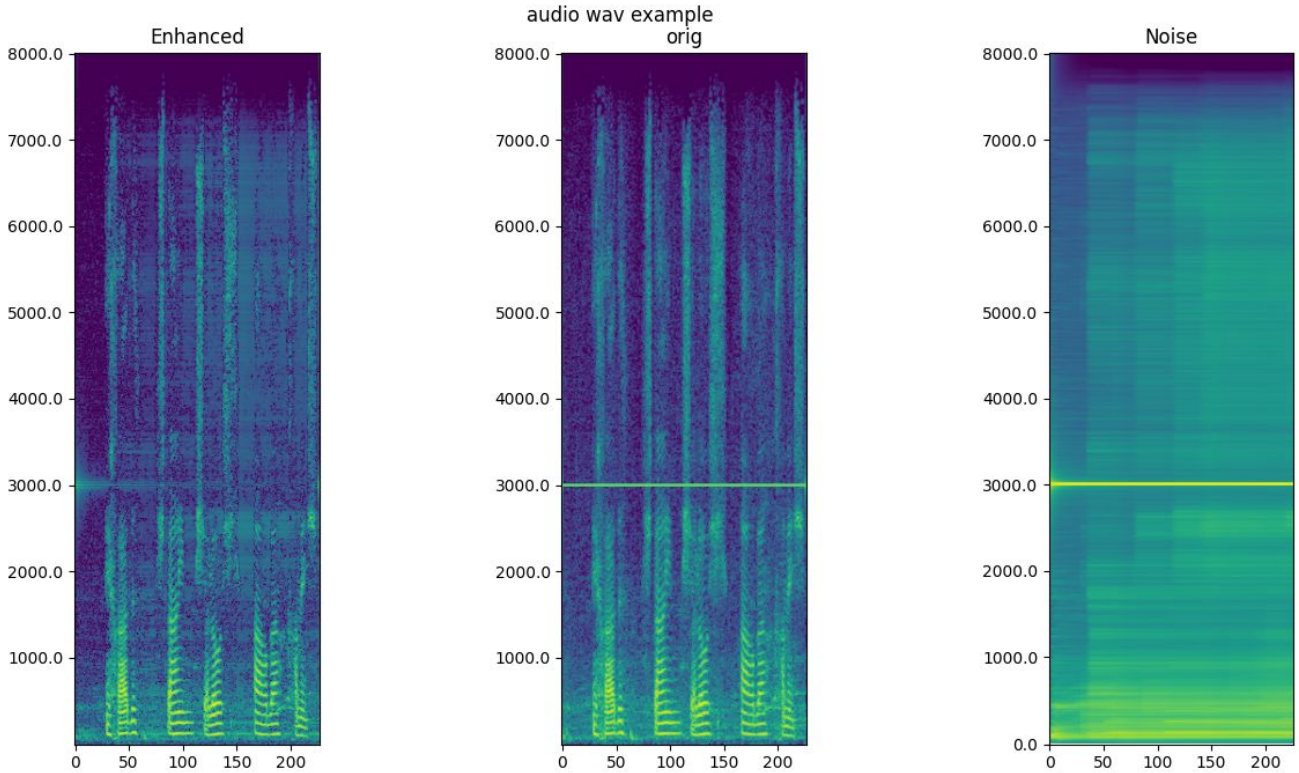
Applications: Speech Enhancement

- When we record audio it is noisy as the we are not in studio.
- We would like to remove background noise from the acquired signal

$$y[n] = x[n] + s[n]$$

- Where: **y[n]** is the acquired signal, **x[n]** is the desired signal, and **s[n]** is stationary background noise signal

Applications: Speech Enhancement

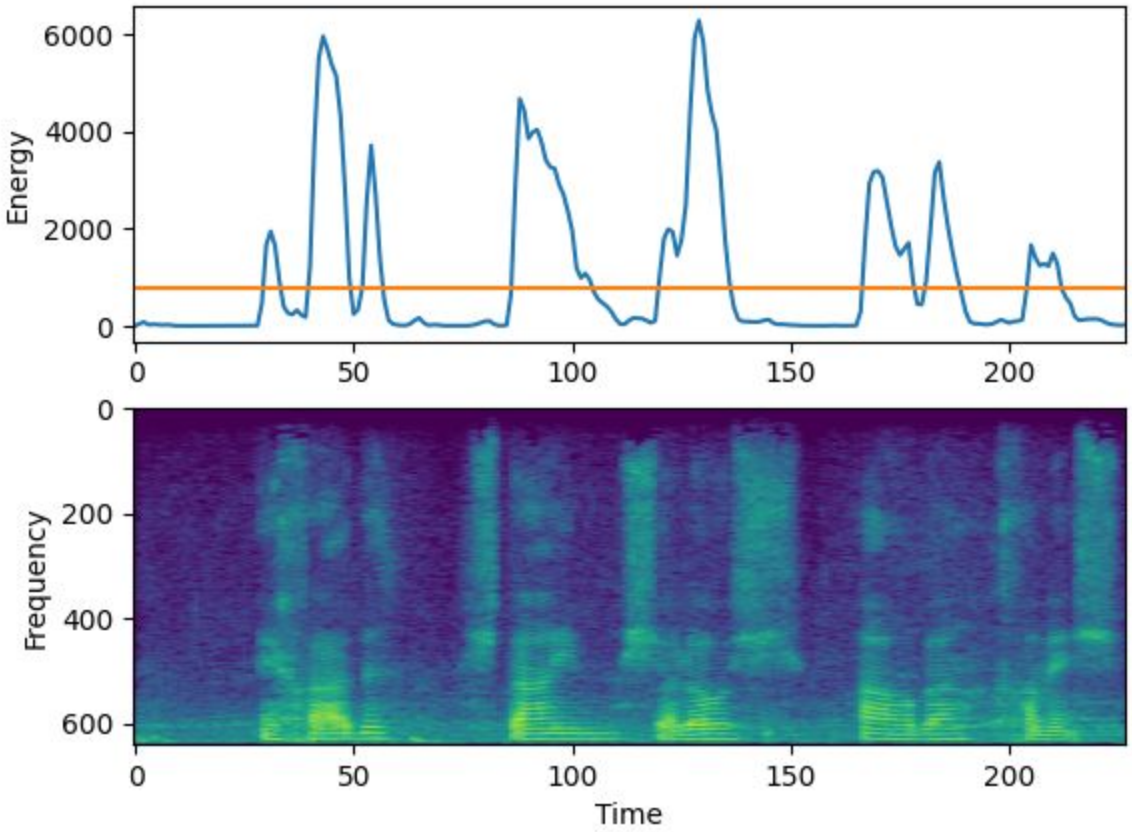


Applications: Speech Enhancement

- Approach: Subtract the estimated noisy magnitude from the acquired signal.
- Therefore, we need to estimate the noise:
  - **Locate segments / frames where there is no speech (using VAD)**
  - Aggregate statistics to a buffer: (determine buffer size)
  - Average the buffer to form the noise estimation (assuming non stationary signals will be averaged out)
  - Subtracting the noise estimation from the input signal y[n]

Applica

- Approach
- Therefore
  - **Loca**
  - Aggr
  - Aver
    be a
  - Subt

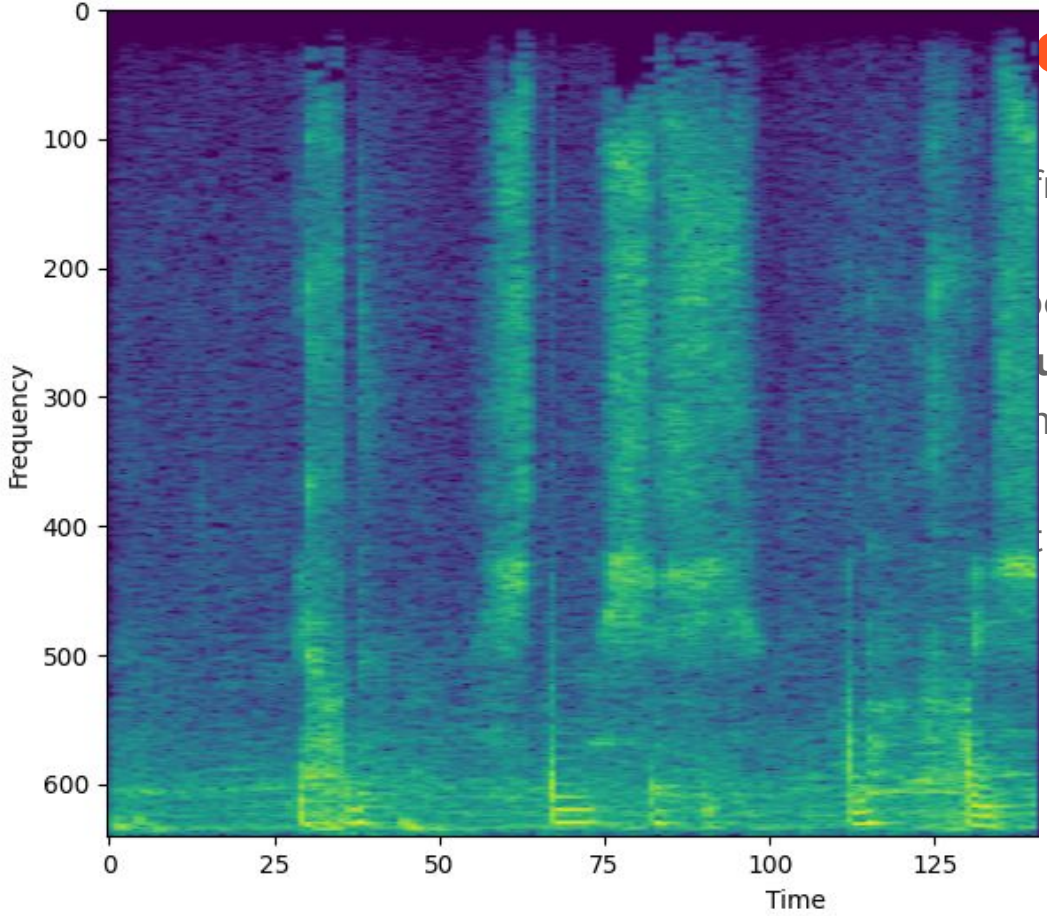


al.

nary signals will

Applications: Speech Enhancement

- Approach: Subtract the estimated noisy magnitude from the acquired signal.
- Therefore, we need to estimate the noise:
  - Locate segments / frames where there is no speech (using VAD)
  - **Aggregate statistics to a buffer: (determine buffer size)**
  - Average the buffer to form the noise estimation (assuming non stationary signals will be averaged out)
  - Subtracting the noise estimation from the input signal y[n]



cement

rom the acquired signal.

eech (using VAD)

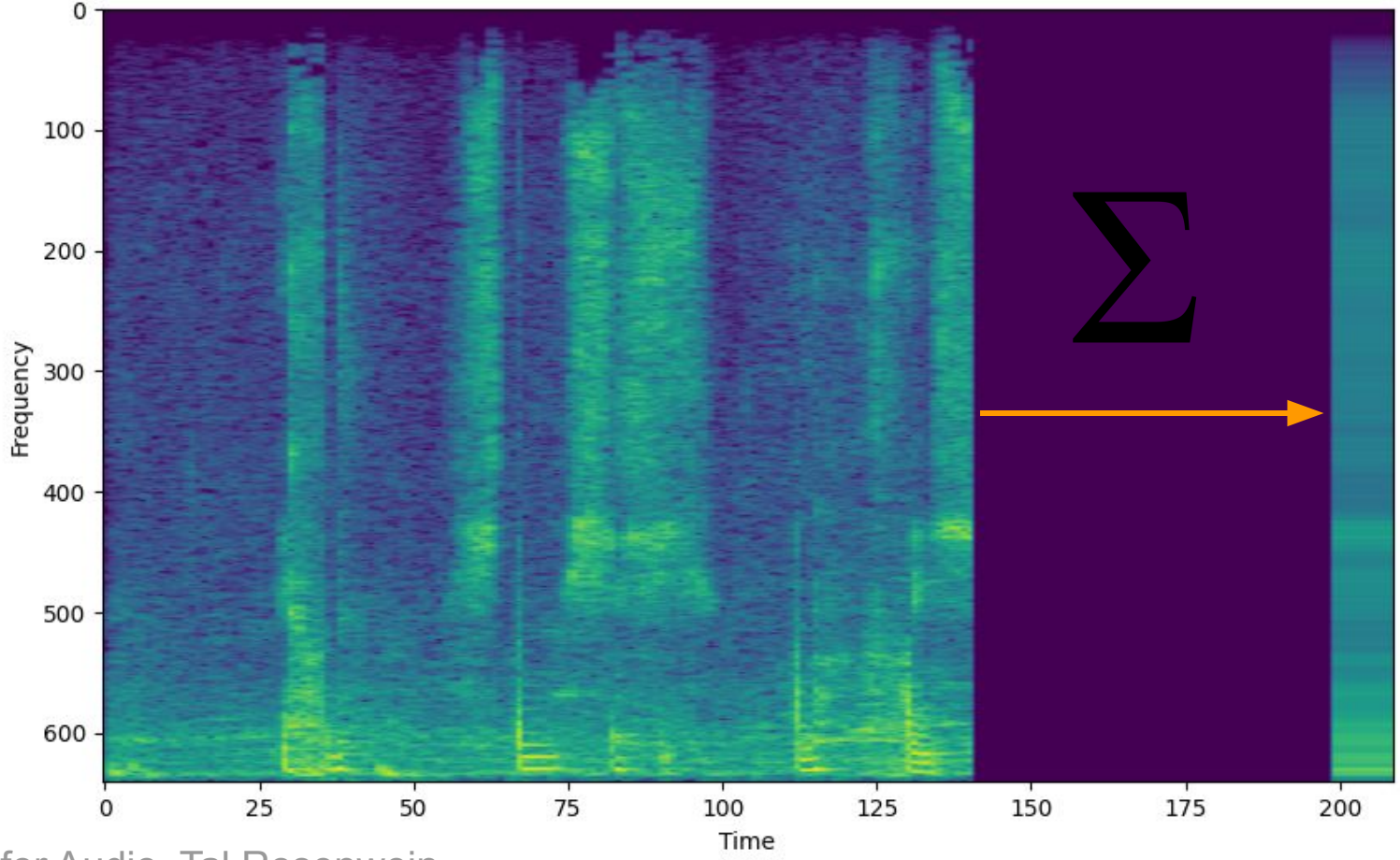
uffer size)

(assuming non stationary signals will

signal y[n]

Applications: Speech Enhancement

- Approach: Subtract the estimated noisy magnitude from the acquired signal.
- Therefore, we need to estimate the noise:
  - Locate segments / frames where there is no speech (using VAD)
  - Aggregate statistics to a buffer: (determine buffer size)
  - **Average the buffer to form the noise estimation (assuming non stationary signals will be averaged out)**
  - Subtracting the noise estimation from the input signal y[n]


0
100
200
300
400
500
600
Frequency
0
25
50
75
100
125
150
175
200
Time
Σ

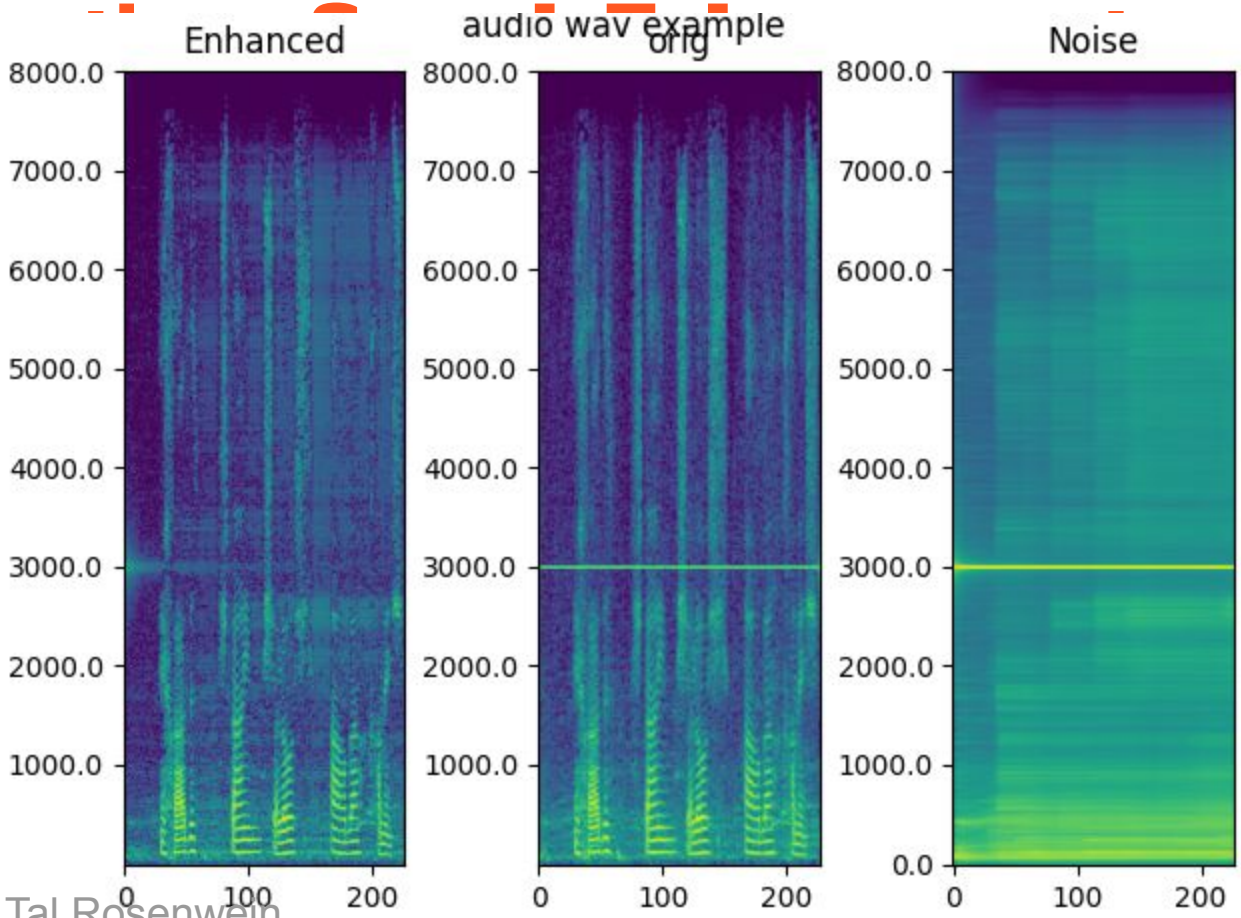
ary signals

Applications: Speech Enhancement

- Approach: Subtract the estimated noisy magnitude from the acquired signal.
- Therefore, we need to estimate the noise:
  - Locate segments / frames where there is no speech (using VAD)
  - Aggregate statistics to a buffer: (determine buffer size)
  - Average the buffer to form the noise estimation (assuming non stationary signals will be averaged out)
  - **Subtracting the noise estimation from the input signal y[n]**

Appli

- Appro
- There
  - I
  - A
  - A
  - S



nal.

onary signals will

Applications: Speech Enhancement

- Python and Audacity examples
- Challenges:
  - Updating vs freezing the noise estimation vector: pros and cons
  - Musical noises may appear when filtering is too aggressive:

$$o[n] = \alpha y[n] + (1 - \alpha)\bar{x}[n]$$

- Learnable methods (DEMUCS)

Applications

Challenges:

1. Delayed signal / packet lost - Time stretching
2. Noisy environment - Noise reduction
3. **Speakers with multiple gains - Auto gain control**
4. Audio Clipping - Audio Compression
5. Connectivity issues - Variable bitrate vocoders

Image Source

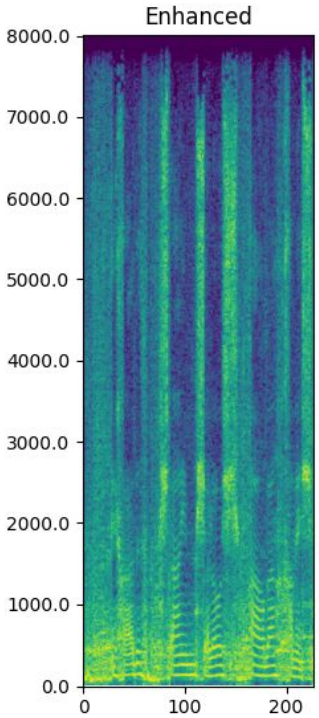
Applications: Auto Gain Control

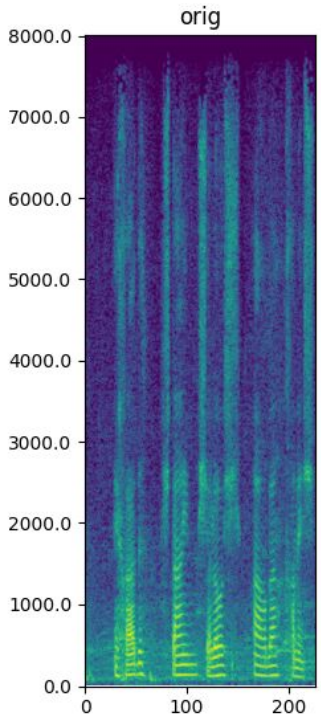
- In many cases we would like to have a constant “volume” regardless of how the speaker is speaking.
  - Our algorithm is sensitive to RMS
  - Our users are sensitive to RMS
- Hence we need to determine the desired RMS and normalize the audio accordingly
  - Applying the same factor to all frames will result in a constant amplification
  - So we need to do it frame-by-frame

Applications: Auto Gain Control

audio wav example

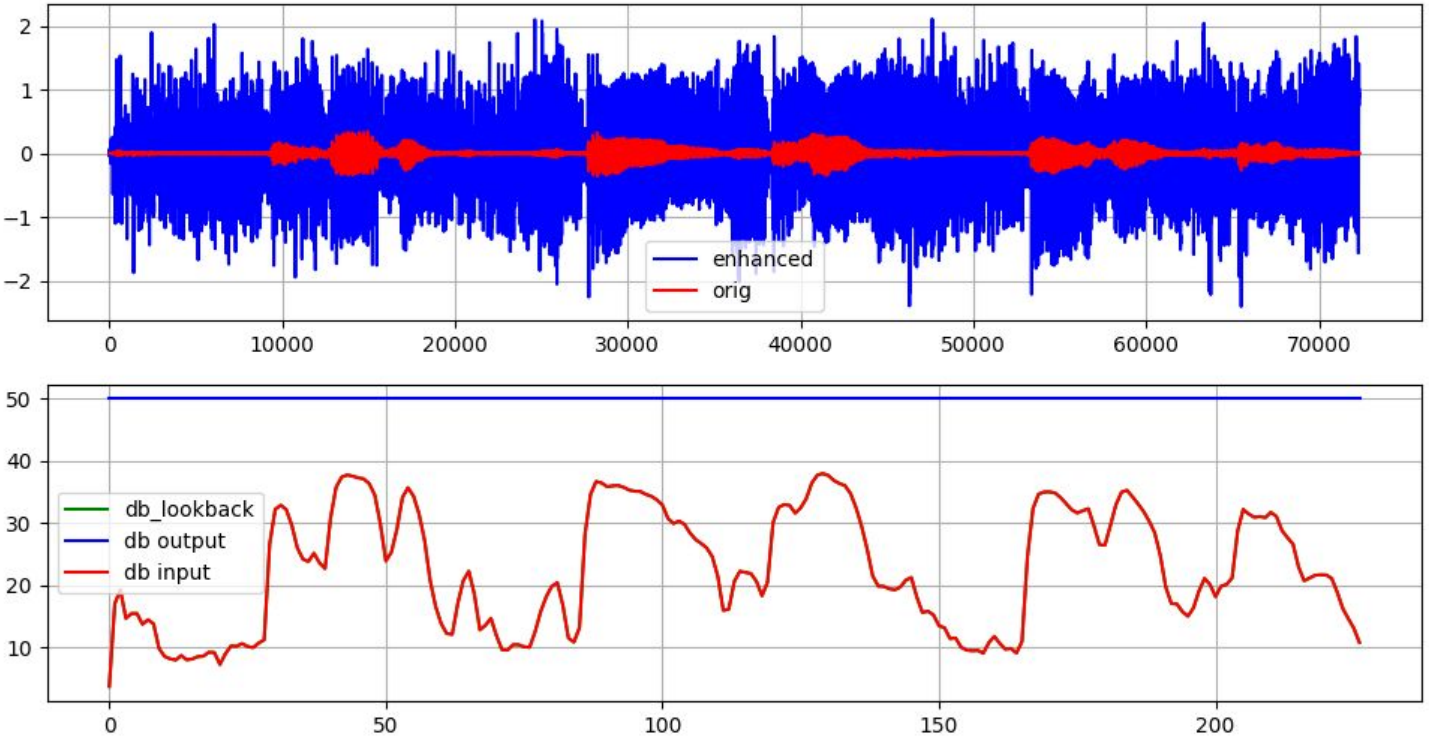
- In many cases v of how the speaker is speaking.
  - Our algori
  - Our users
- Hence we neec dio accordingly
  - Applying t nplification
  - So we nee





Applications: Auto Gain Control

- In r … speaker is s …
  - …
  - …
- He … gly
  - …
  - …



Applications: Auto Gain Control

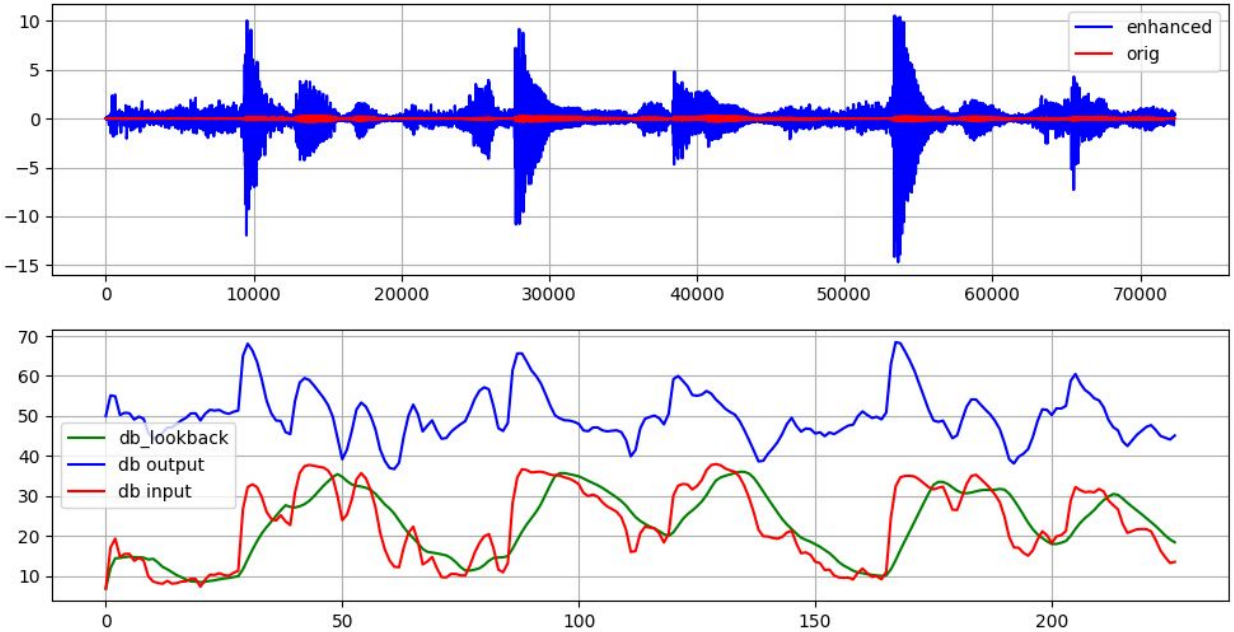
- In many cases we would like to have a constant “volume” regardless of how the speaker is speaking.
  - Our algorithm is sensitive to RMS
  - Our users are sensitive to RMS
- Hence we need to determine the desired RMS and normalize the audio accordingly
  - Applying the same factor to all frames will result in a constant amplification
  - So we need to do it frame-by-frame- “pumping” effect

Applications: Auto Gain Control

- In many cases we would like to have a constant “volume” regardless of how the speaker is speaking.
  - Our algorithm is sensitive to RMS
  - Our users are sensitive to RMS
- Hence we need to determine the desired RMS and normalize the audio accordingly
  - Applying the same factor to all frames will result in a constant amplification
  - So we need to do it frame-by-frame- “pumping” effect
  - So we will aggregate stats

Applications: Auto Gain Control

- In
  is
  - C
  - C
- He
  - C
  - C
  - C

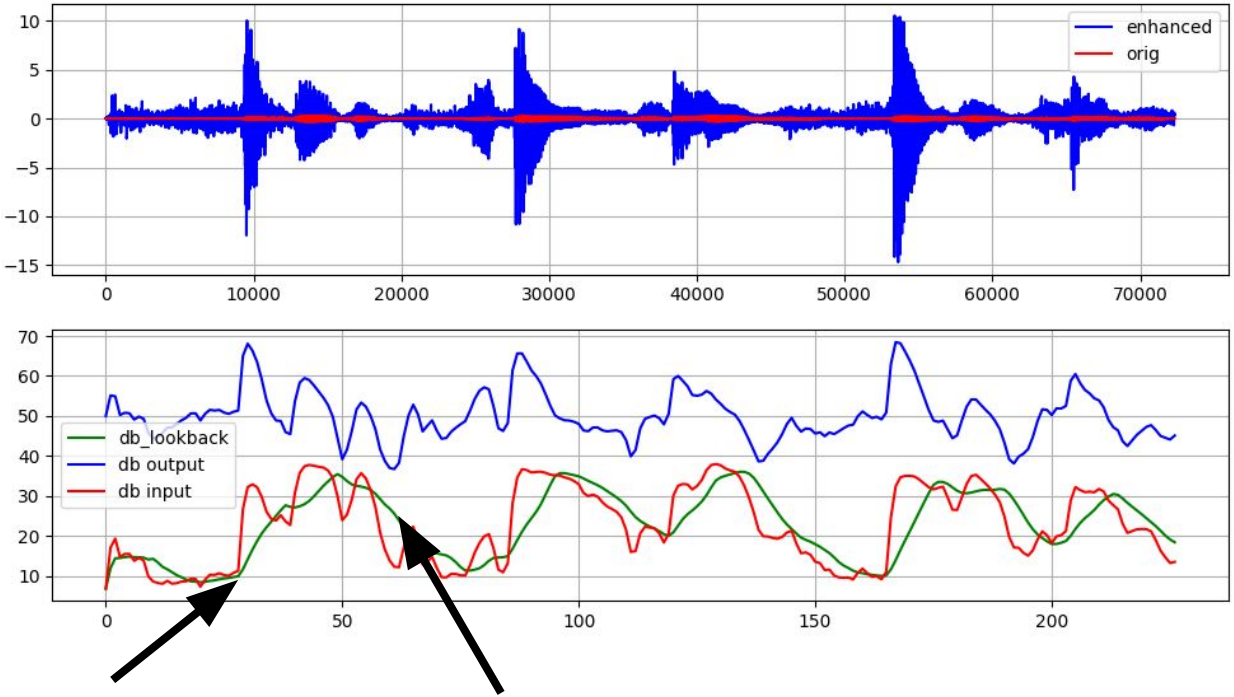


speaker

gly

Applications: Auto Gain Control

- In
  is
  - C
  - C
- He
  - C
  - C
  - C



Attack (Red > Green)

Release (Red < Green)

Applications: Auto Gain Control

- In many cases we would like to have a constant “volume” regardless of how the speaker is speaking.
  - Our algorithm is sensitive to RMS
  - Our users are sensitive to RMS
- Hence we need to determine the desired RMS and normalize the audio accordingly
  - Applying the same factor to all frames will result in a constant amplification
  - So we need to do it frame-by-frame- “pumping” effect
  - So we will aggregate stats
- Additional challenges:
  - Overflow: results in clipping (compression, sigmoid)
  - Do not amplify noise: determine energy threshold (noise floor)
  - Smooth transient noises: Attack and release: using hard decision / soft decision (exponential decay).

Applications

Challenges:

1. Delayed signal / packet lost - Time stretching
2. Noisy environment - Noise reduction
3. Speakers with multiple gains - Auto gain control
4. **Audio Clipping - Audio Compression**
5. Connectivity issues - Variable bitrate vocoders

Applications: Audio Compression

- In many cases we would like to have a different dynamic range
  - People with hearing loss
  - Audio editing- emphasis some parts or making audio sound better

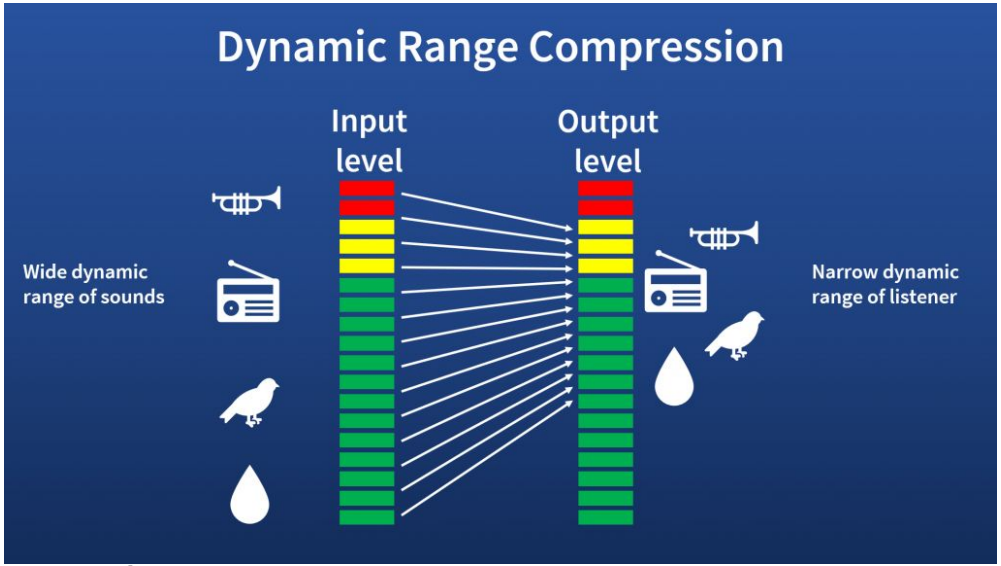


Image Source

Applications: Audio Compression

- In many cases we would like to have a different dynamic range
  - People with hearing loss
  - Audio editing- emphasis some parts or making audio sound better
- We need to determine the mapping between the input and output gains
  - Attack and release: when / how fast do we want to apply compression: only on non-transient parts.

Applications

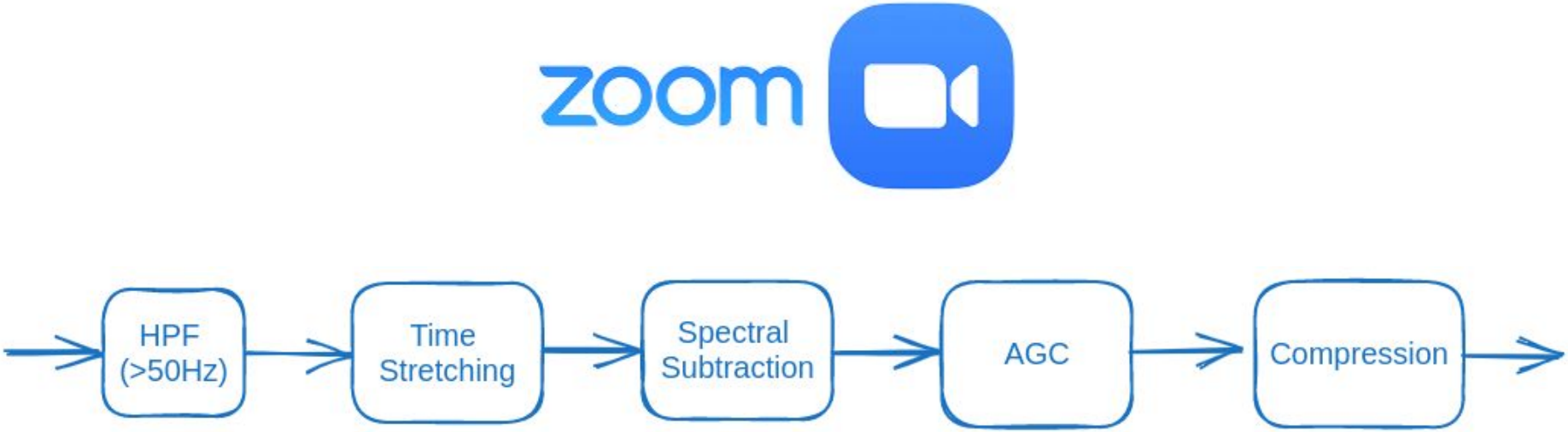


Image Source

Applications

Challenges:

1. Delayed signal / packet lost - Time stretching
2. Noisy environment - Noise reduction
3. Speakers with multiple gains - Auto gain control
4. Audio Clipping - Audio Compression
5. **Connectivity issues - Variable bitrate vocoders**

Image Source

Overview

- Motivation
- Communication
- Anatomy & Speech production system
- Phonetics
- Acoustic/Speech Features
- Traditional Speech Signals Analysis
  - Time Stretching
  - Speech Enhancement- Spectral Subtraction
  - Auto Gain Control (AGC)
  - Audio Compression