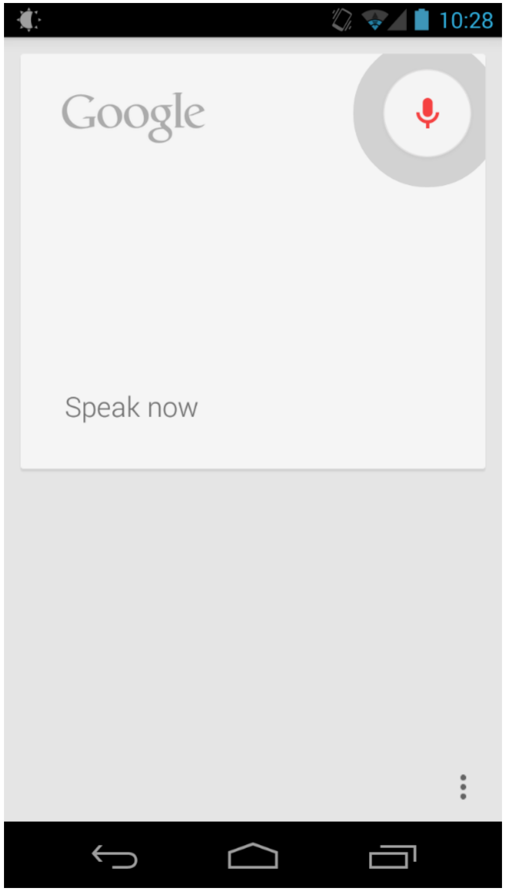
ETH zürich

Speech recognition in systems for human-computer interaction

Ubiquitous Computing Seminar FS2014

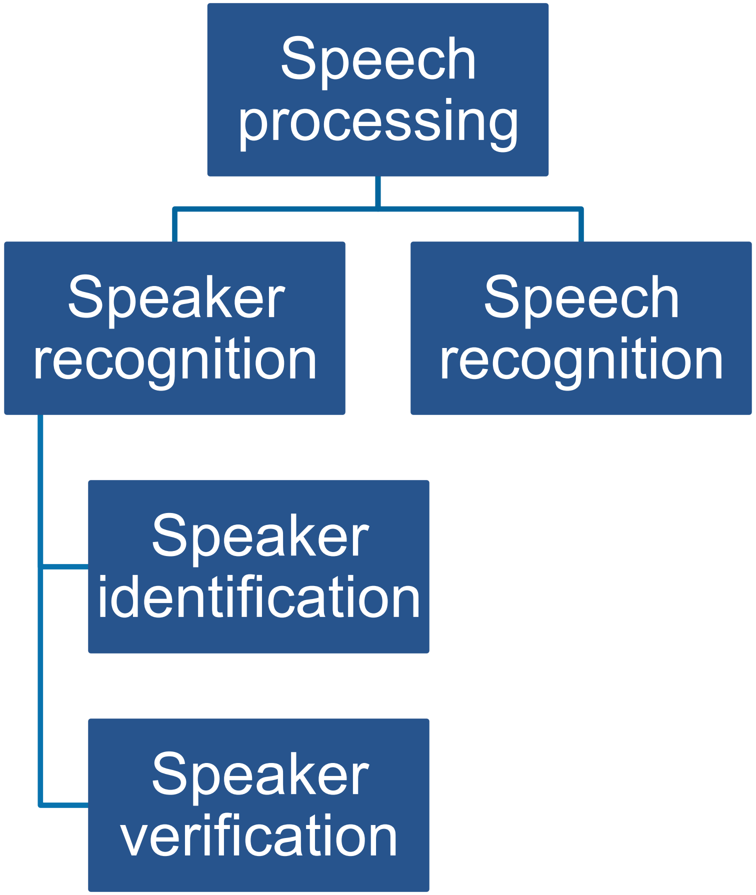
Niklas Hofmann

Why speech recognition?



Source: http://www.freepixels.com/index.php?action=showpic&cat=20&pic | Google Voice Search Android

Speech processing



Speaker verification

- User claims identity
- Binary decision
  - Either identity claim is correct
  - or «access» denied
- Enrollment
- Text dependent vs. independent

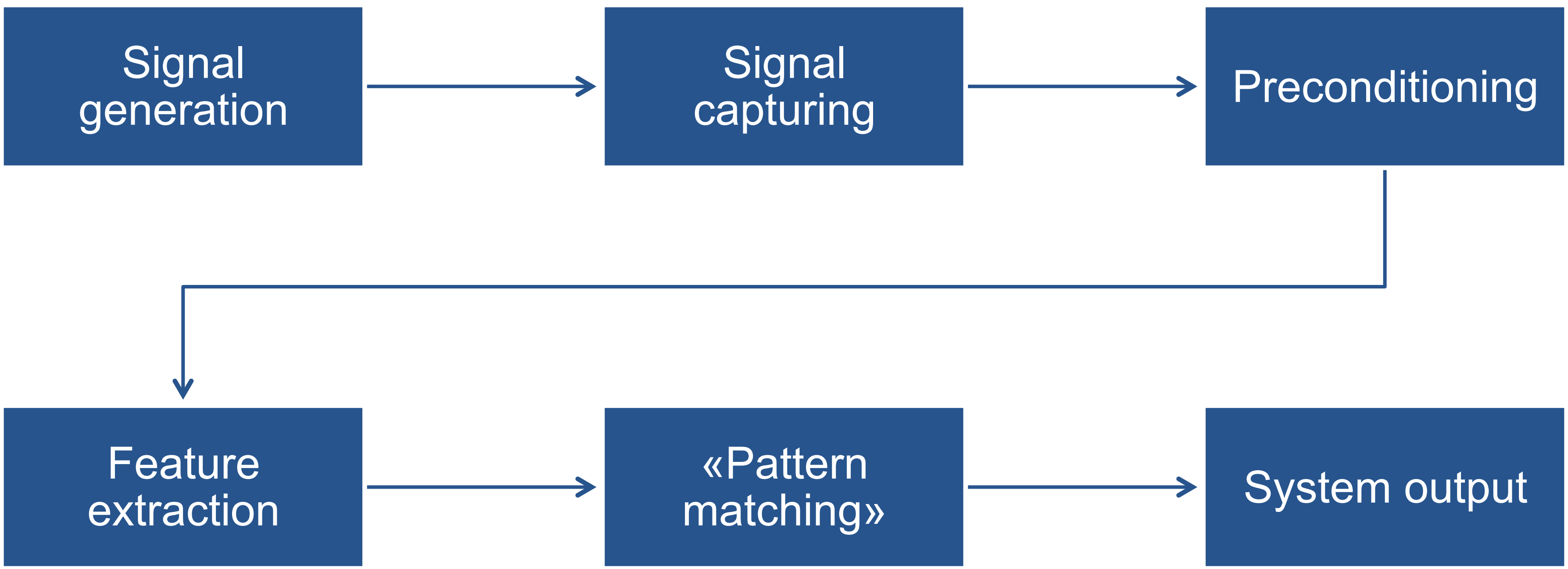
Speaker identification

- No apriori identity claim
- Enrollment
- Open vs. closed group
- Text dependent vs. independent

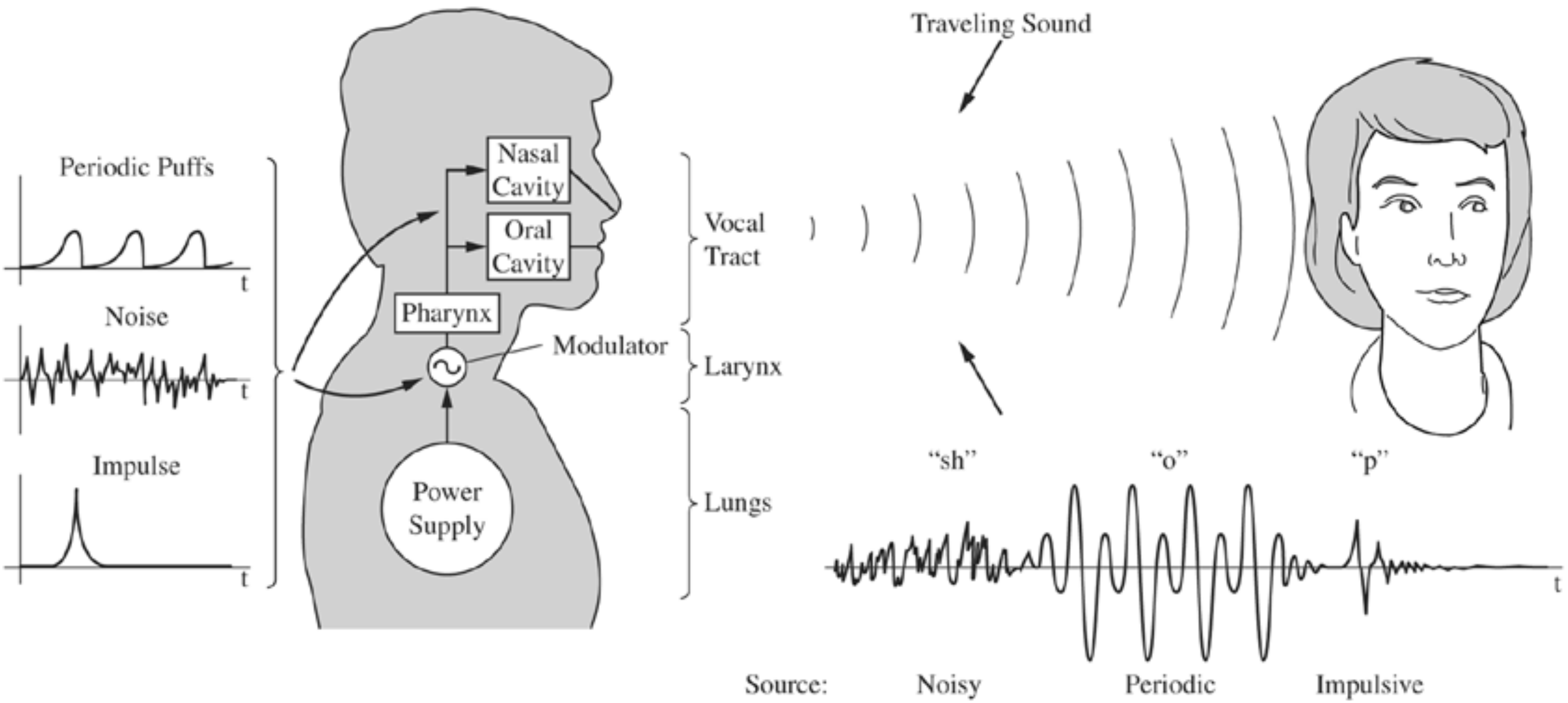
Speech recognition

- Recognize spoken language
- Speaker independent vs. dependent
- Restricted input vs. «speech-to-text»
- No predefined usage
  - Commands
  - Data input
  - Transcription

Speech processing stages



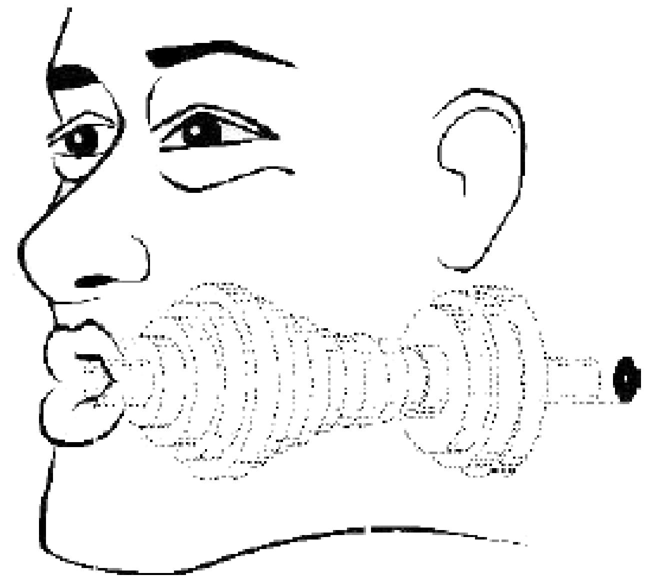
Signal generation



Source: Discrete-time speech signal processing | T. Quatieri | 2002

Signal generation

- Simplified vocal tract
- Time invariant for a short time
- Source modeled as
  - Periodic signal
  - Noise
- Speech as overlay of source and resonance



Source: Introduction to Speech Processing | Ricardo Gutierrez-Osuna | CSE@TAMU 2011

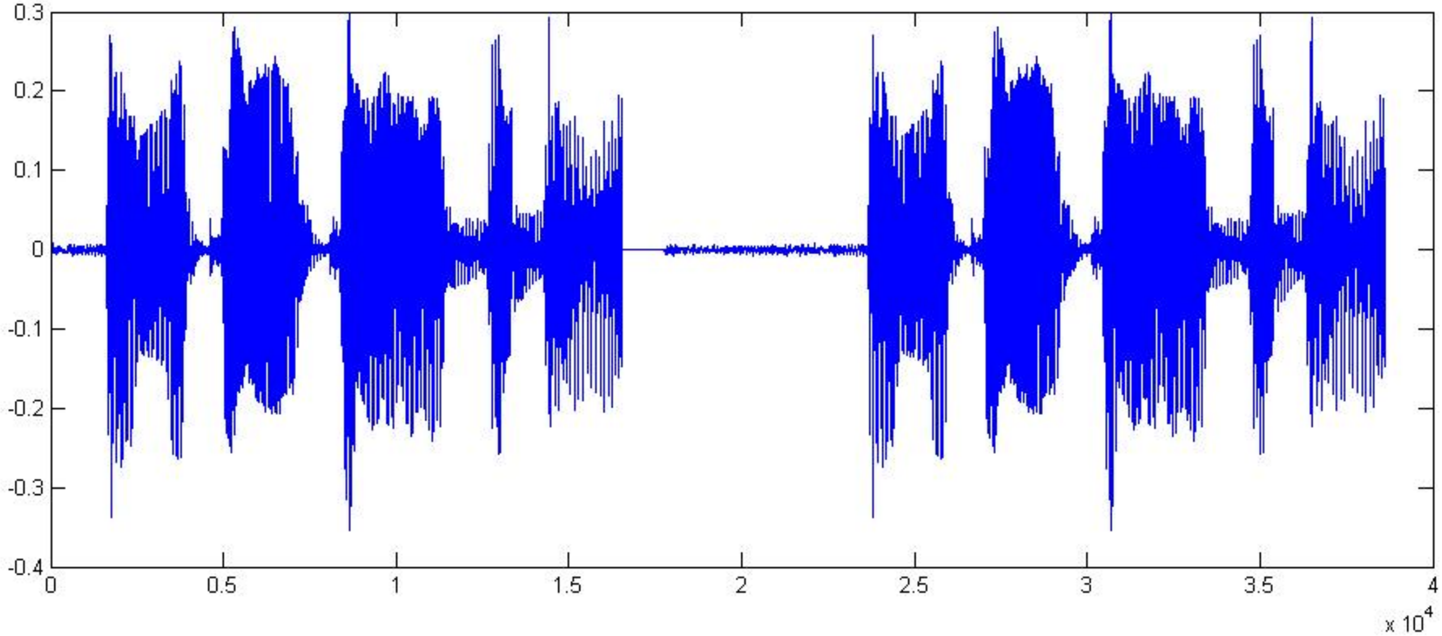
Signal capturing / preconditioning

- Microphone
  - Bandwidth
  - Quality (better quality → easier to detect features)
- Ambience
  - Noise
  - Echo
- Start / Endpoint detection
- Normalization
- Emphasize relevant frequencies
  - Similar to human hearing

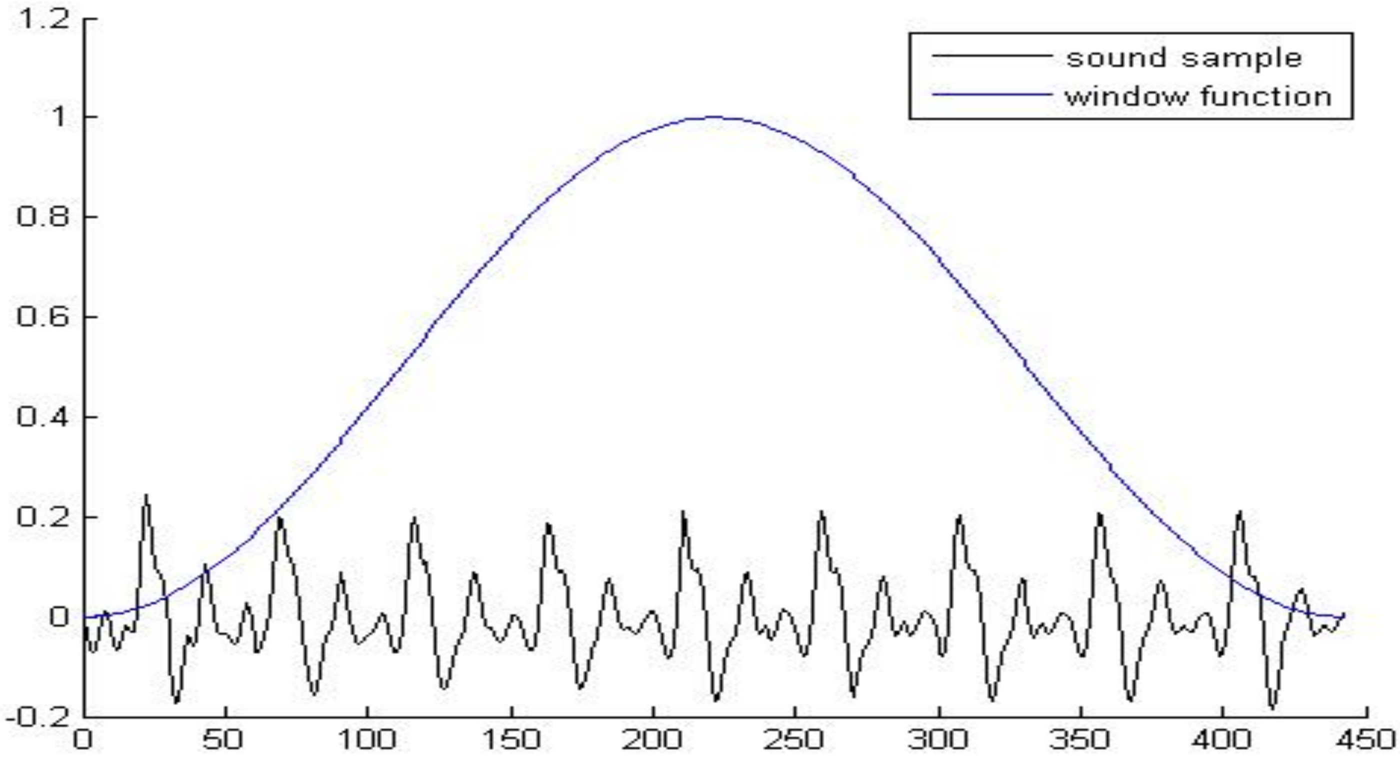
Feature extraction

- Signal framing
  - Vocal tract static for small frame (20-40ms)
- Performed on either
  - Waveform
  - Spectrum
  - Ceptstrum
  - Mix of all
- Techniques used
  - Linear Prediction
  - Cepstral Coefficients

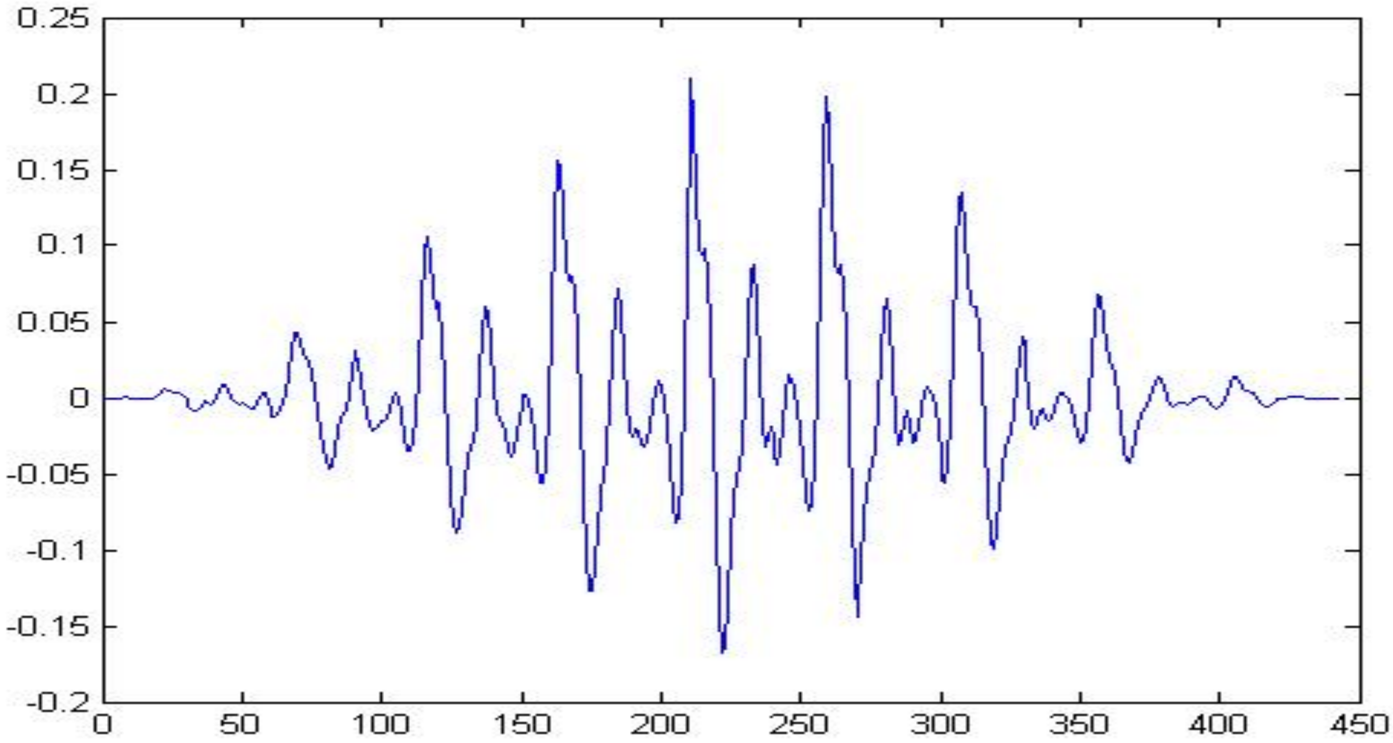
Framing



Framing


1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
50
100
150
200
250
300
350
400
450
sound sample
window function

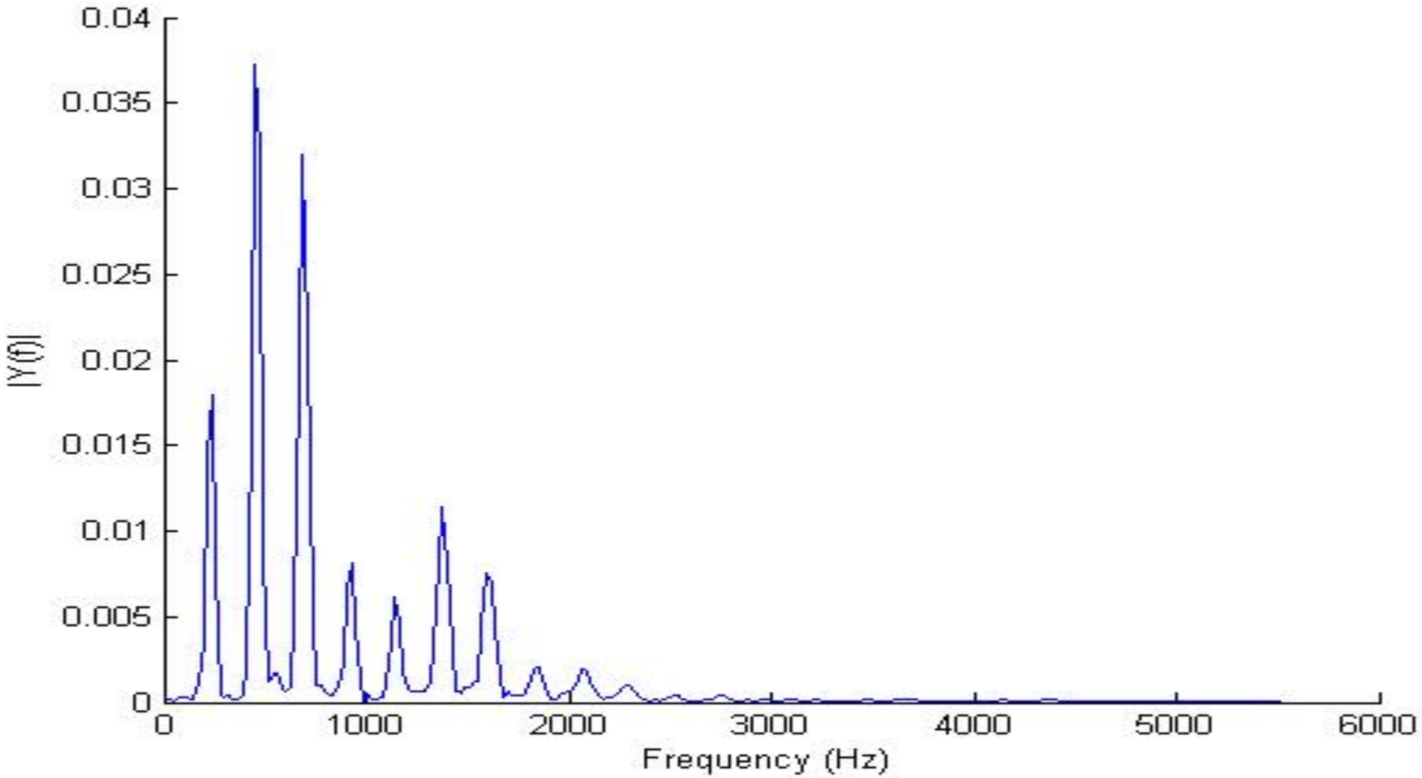
Waveform


0.25
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
0
50
100
150
200
250
300
350
400
450

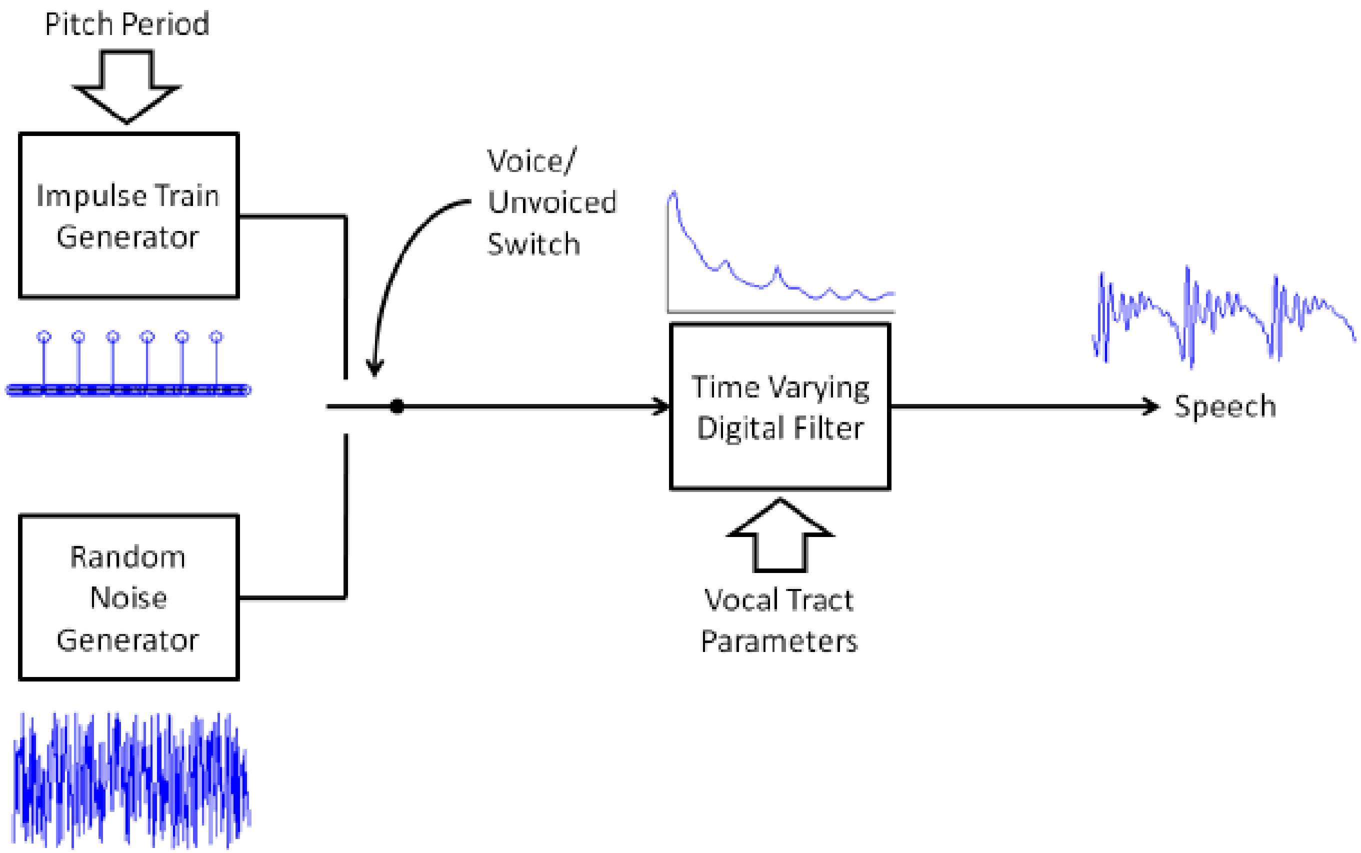
Spectrum

- Transform sample from time domain to frequency domain
- Invention of FFT very helpfull (1965)
- Gives insight in periodicity of a signal
- Sensitive to framing (→ window functions)

Spectrum

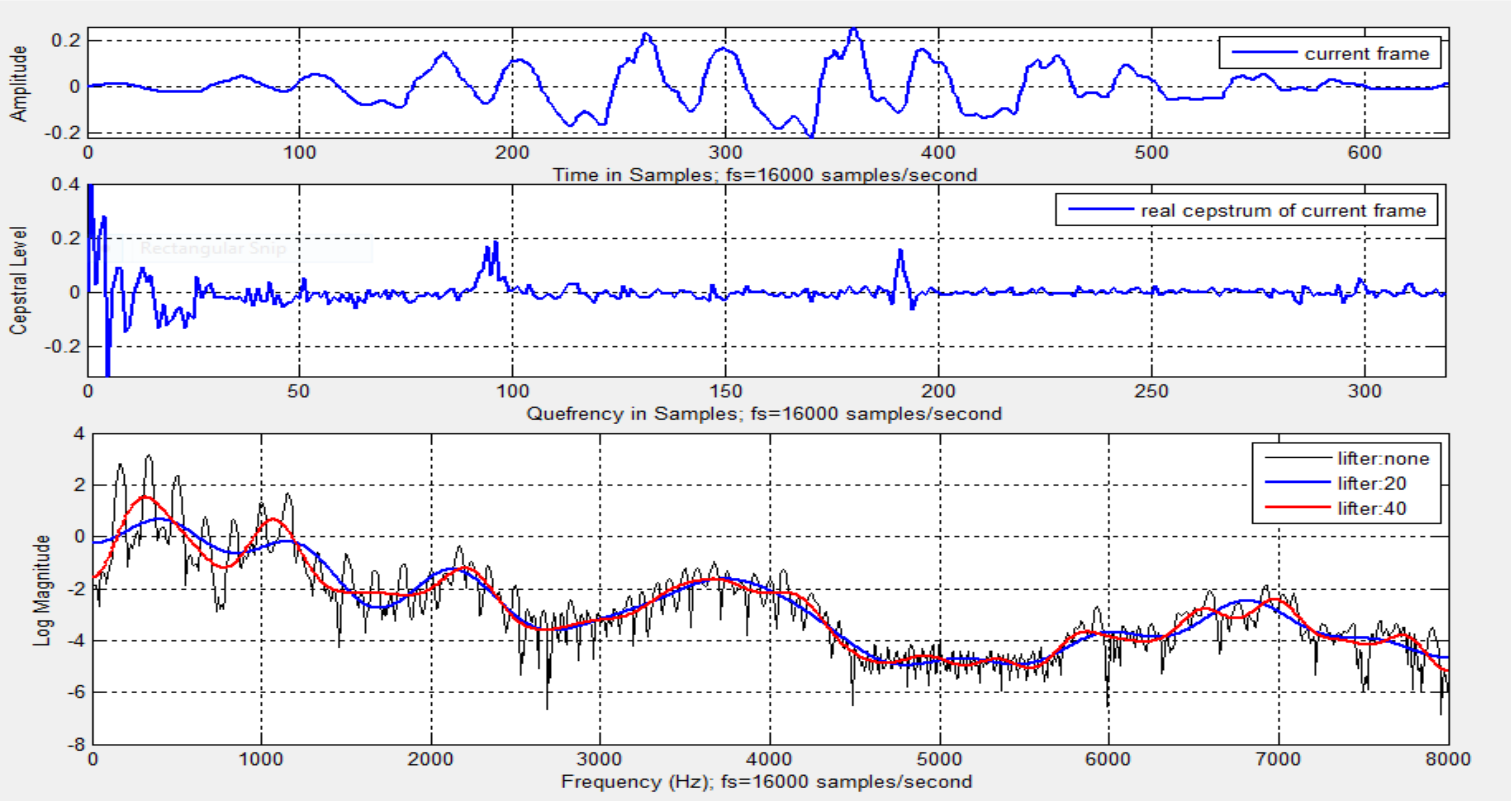

0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
|Y(f)|
0
1000
2000
3000
4000
5000
6000
Frequency (Hz)

Linear prediction



Source: Linear Prediction | Alan O Cinnéide | Dublin Institute of Technology | [2008]

Cepstral coefficients



«Pattern matching»

- «Detect» speech units (phonemes / words) out of series of feature vectors
- Two main ideas
  - Template matching
    - «Simple» matching
    - Dynamic time warping
  - Statistical
    - Hidden markov model

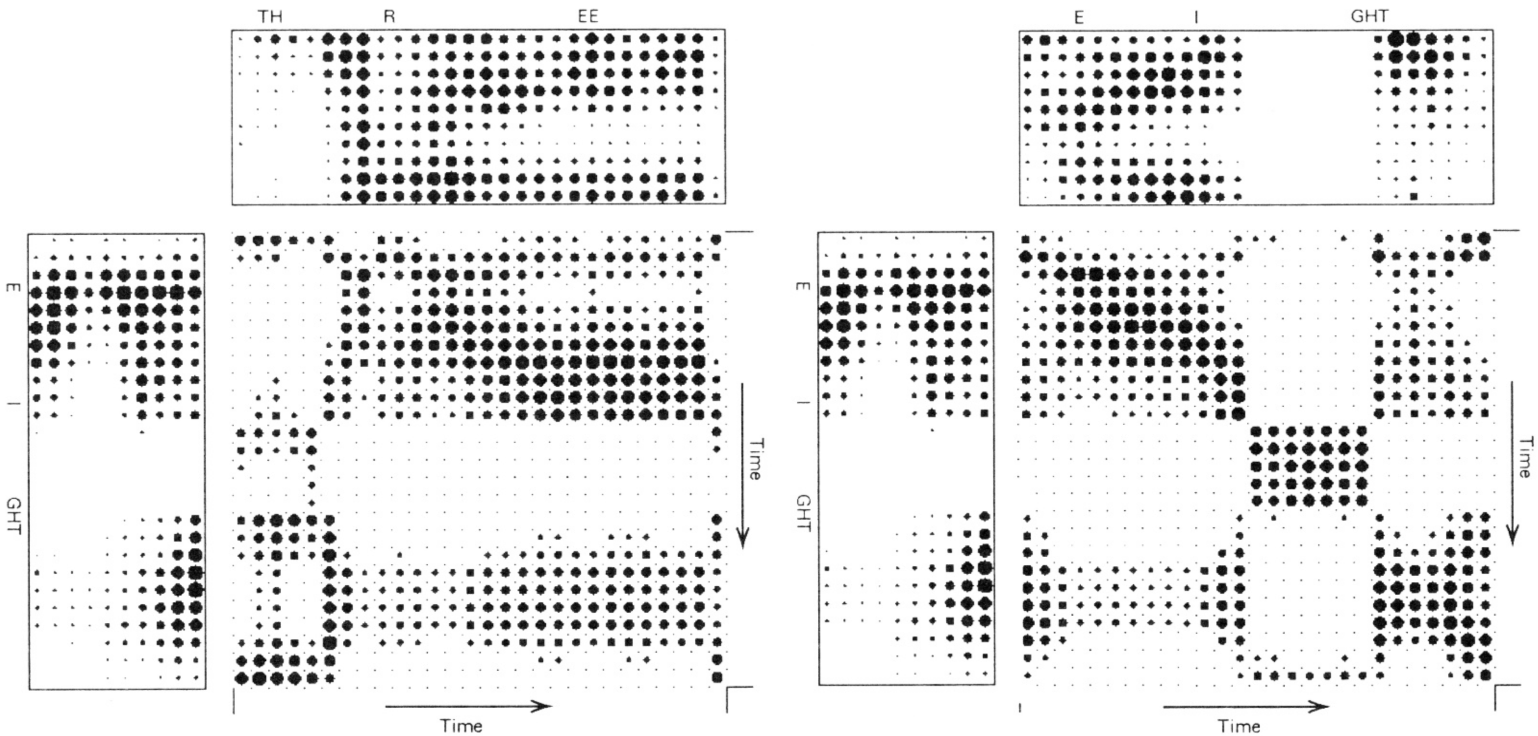
«Simple» matching

- Calculates distance from sample to template
- Simple to implement
- Assumes sample and template of same length / speed
  - Very sensitive to different speech patterns (length, pronounciation)
- No widespread use anymore

Dynamic time warping (DTW)

- Tries to «correct» slower/faster sample with respect to template
- Uses metrics to disallow too much «warping»
- Still calculates «distance» between sample and template

Dynamic time warping (DTW)



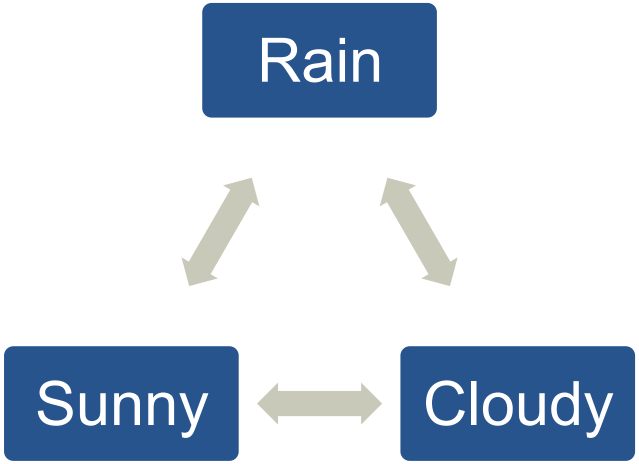
Source: Speech Synthesis and Recognition | John Holmes and Wendy Holmes | [2nd Edition]

Hidden markov model (HMM)

- Models speech as process with hidden states and observable features
- Each unit (e.g. word) matched to own process
- Gives probability that sample generated from a certain process
- Described by:
  - Set of n States S_n
  - State transition matrix A
  - (probability density function for the observations for each state, b_i)

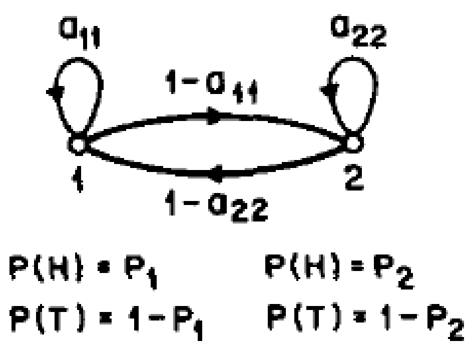
Hidden markov model (HMM)

- Example: Weather
  - State 1: rain / snow
  - State 2: cloudy
  - State 3: sunny



Hidden markov model (HMM)

- State not necessarily mapped to observation
  - Multiple observations possible in one state
  - Each observation has different probability to be seen
  - E.g. Series of «head» and «tails» can be generated by single coin or by two or more different coins (we do not know which coin is thrown when)



O = H H T T H T H H T T H ...

S = 2 1 1 2 2 2 1 2 2 1 2 ...

Source: Tutorial on Hidden Markov Models | L. R. Rabiner | 1989

Applying HMM to speech recognition

- Idea: generate one HMM per word
  - Very complex for longer words
  - Recognition of words not in training set impossible/improbable
- Divide word into subunits (phonemes)
  - E.g. Cat → /k/ + /a/ + /t/
- Train one HMM per phoneme (~45 for english)
- Chain HMM together to recognize words / sentences

Applying HMM to speech recognition

- One possible model:
  - 1 State for transition in: /sil/ → /a/
  - 1 State for the middle: /a/
  - 1 State for transition out: /a/ → /sil/
- Phoneme level HMM still not accurate enough
- Context can alter sound of phoneme
- Use context dependent models

Applying HMM to speech recognition

- Triphone: e.g. Cat
  - First triphone: /sil/ → /k/ → /a/
  - Second triphone: /k/ → /a/ → /t/
  - Third triphone: /a/ → /t/ → /sil/
- Solves context sensitivity but high computation cost:
  - 45 phoneme → $45 \uparrow 3 = 91125$ different models (not all needed)

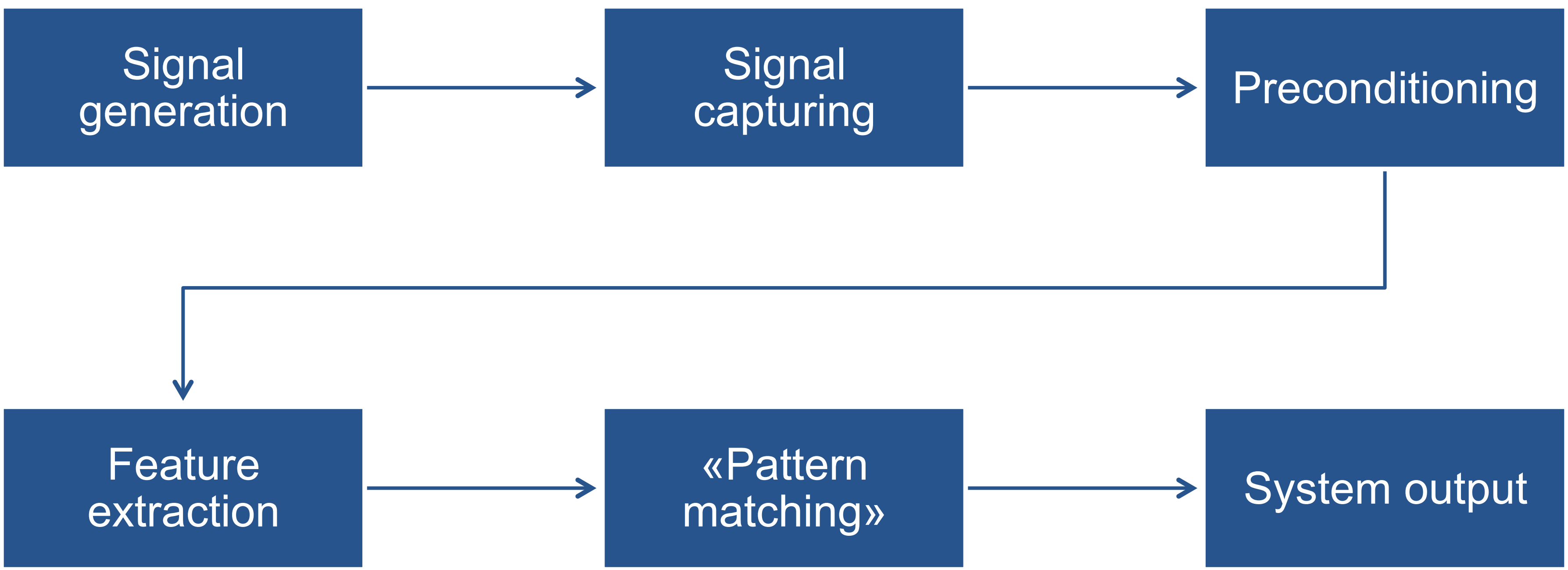
DTW vs HMM

- Performed with 16 speakers (8:8 male:female)
- Utterance of digits 0 – 9
- Also compared linear prediction to cepstral coefficients

Types of features	Frame length	Recognition Accuracy in % using DTW	Recognitio n Accuracy in % using HMM
LPC	13	69	86
LPC+Δ+Δ^2	39	76	91
MFCC	13	77	90
MFCC+Δ+Δ^2	39	86	94

Source: Comparison of DTW and HMM | S. C. Sajjan | 2012

Speech processing stages



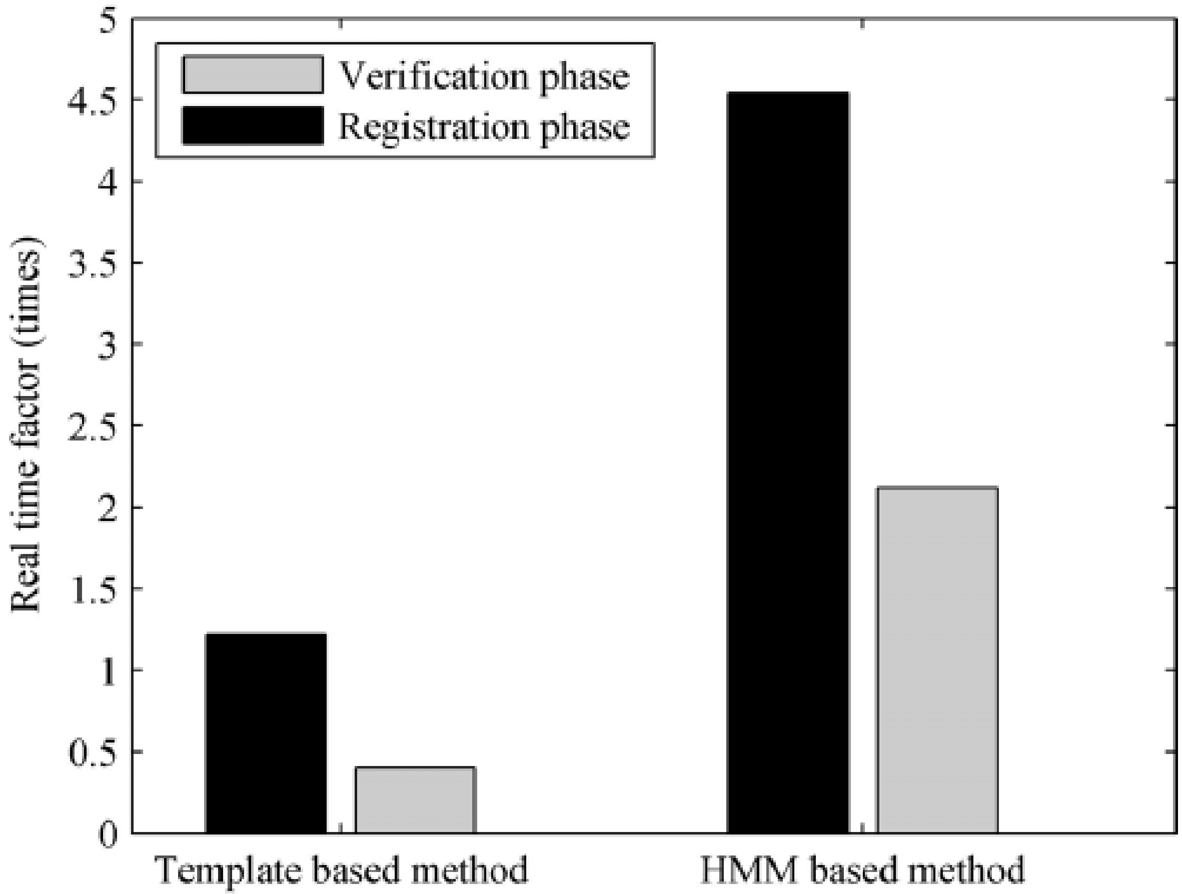
Speech recognition on mobile devices

- Limited power supply
  - Prevent frequent unneeded activation of system
- Limited storage
  - Tradeoff between size and performance of speech and language models
- Limited computing power
  - Tradoff between accuracy and speed
- Long training undesirable

Performance on mobile device

- Comparison of DTW to HMM on mobile device (2009)
  - 500 MHz CPU
- Detection of keywords of specific user
- Data set of 30 people
  - 7 females and 23 males
  - Speaking 6 words (4-11 phonemes)
  - Each word repeated 10 times

Real time factor



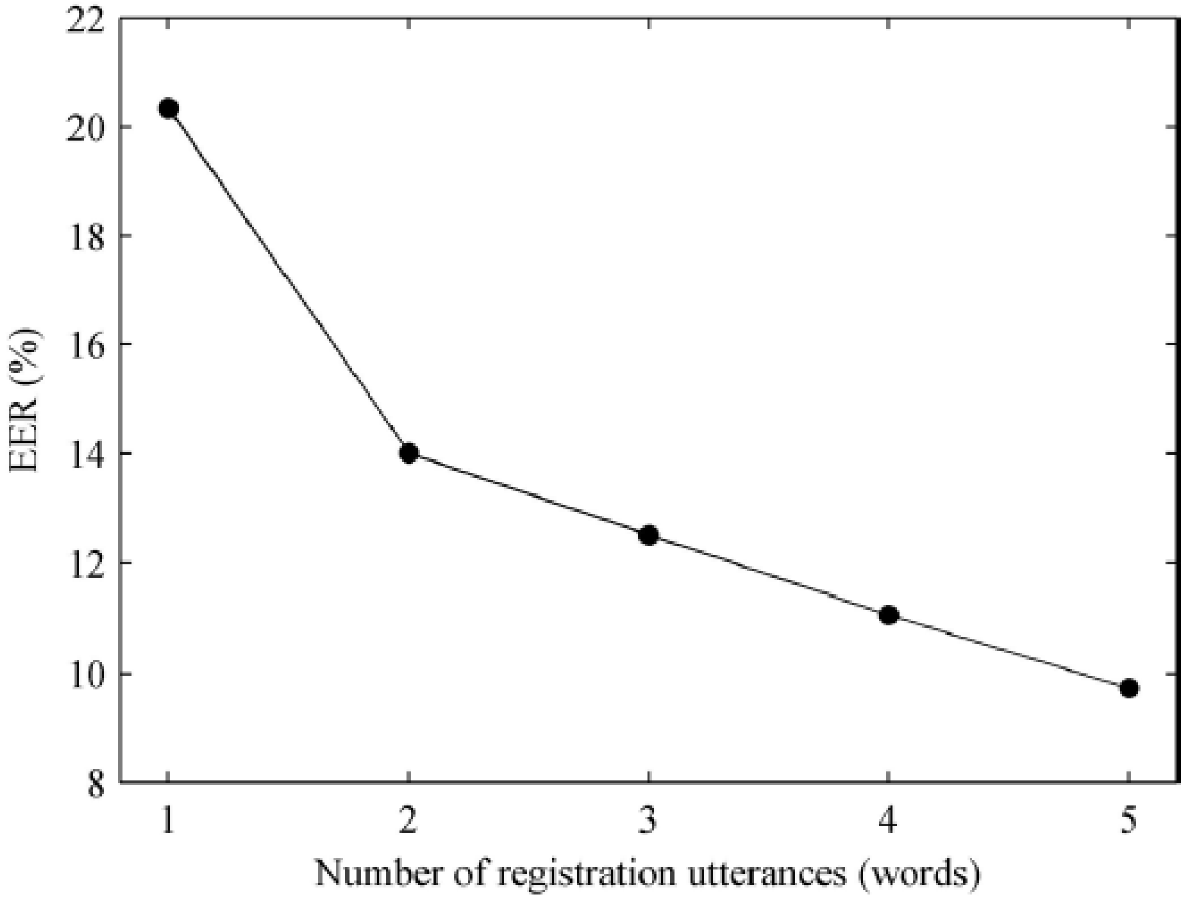
Source: Voice Trigger System | H. Lee, S. Chang, D. Yook, Y. Kim | 2009

Error rate

- Meassured «equal error rate»
  - Acceptance threshold set to get equal
    - False posivite rate
    - False negative rate

- Dynamic Time warping: ~14% error rate
- Hidden Markov model: down to ~9% error rate
  - Heavily dependent on ammount of training data

Hidden markov model



Source: Voice Trigger System | H. Lee, S. Chang, D. Yook, Y. Kim | 2009

What about modern cloud based systems?

- Multiple «consumer grade» systems deployed
  - 2008 Google Voice Search for Mobile App on iPhone
  - 2011 Apple launches Siri on iOS
  - 2011 Google adds Voice Search to Google.com

A closer look on Google Voice Search

- Experiments done with 39-dimensional LP-cepstral coefficients
- Uses triphone system
- Relies heavily on a language model to decrease computation and increase accuracy

Language model

- Learned from typed search queries on google.com
  - Trained on over 230 billion words
- Also accounts for different locales

Training Locale		Test Locale	
	USA	GBR	AUS
USA	**0.7**	1.3	1.6
GBR	1.3	**0.7**	1.3
AUS	1.3	1.1	**0.7**

(Out-Of-Vocabulary rate : percentage of words unknown to the language model)

Source: Google Search by Voice: A case study | Google Inc.

A look into the future

- Modern capabilities of computers enable more complex systems than ever
- Rediscovery of artificial neural networks

- But problem still not solved:
  - No automatic transcription of dialog

Thank you