Human-Computer Interaction

Session 9
Speech Interaction, Recognition & Synthesis

Interaction paradigms: machines as

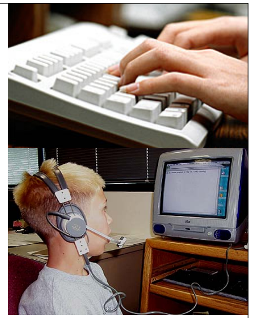
tools ➜ operate

User has to operate the computer continuously

1. user operates the machine
2. machine performs local problem-solving task
3. machine gives feedback to user
4. goto 1.

User-centered design as process to build usable tools

Interaction paradigms: machines as



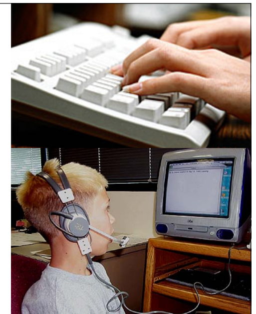
tools ➜ operate

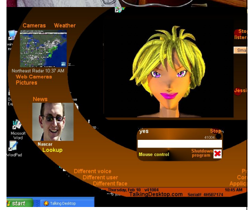
smart tools ➜ instruct

Tools for tasks, but smarter and more autonomous such that the user can instruct them efficiently and easily

- task intelligence ➔ commands on abstract levels concerning more complex procedures or (sub-)tasks
- interaction intelligence ➔ commands in more fashion (language, multimodal)

Interaction paradigms: machines as





tools ➜ operate

smart tools ➜ instruct

assistants ➜ converse

User sets goal, delegates subtasks to system

- autonomous execution, possibly proactive suggestions by the system
- reciprocal conversation to clarify and specify subtasks

Interaction paradigms: machines as

tools ➜ operate

smart tools ➜ instruct

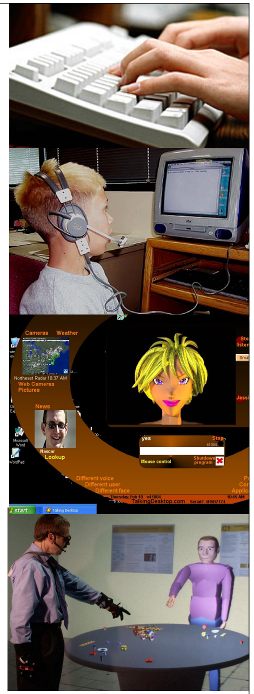
assistants ➜ converse

companions ➜ collaborate

User and system cooperate on par

both can negotiate tasks, raise goals, distribute subtasks, ask for support

additional social dimension (mutual appreciation, trust, empathy, etc.)



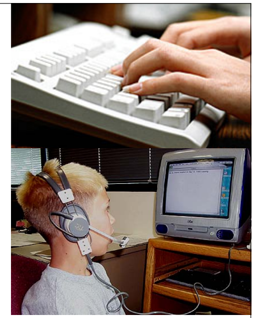
Interaction paradigms: machines as

tools ➜ operate

smart tools ➜ instruct

task intelligence ➔ commands on abstract levels concerning more complex procedures or (sub-)tasks

interaction intelligence ➔ commands in more fashion: spoken language (speech), multimodal interaction, gestures, etc.



Speech interaction

More and more used today...

- on the desktop, e.g. dictate
- on the phone, e.g. ticket booking, pizza ordering

Ongoing research on...

- natural language processing
- mobile devices & robots
- automotive interaction
- Virtual Reality
- ...

Overview: machines as...

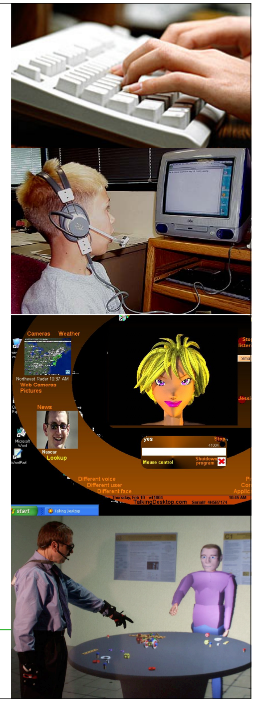
tools ➜ operate

smart tools ➜ instruct

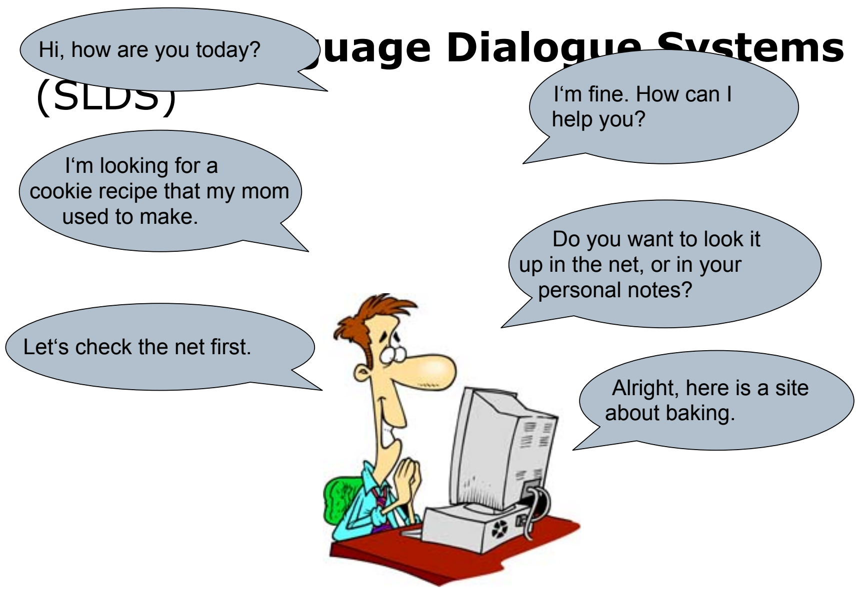
Spoken Language Dialogue Systems

assistants ➜ converse

companions ➜ collaborate



uage Dialogue Systems (SLDS)



Spoken Language Dialogue Systems (SLDS)

A system that allows a user to speak his queries in natural language and receive useful spoken responses from it

Provides an interface between the user and a computer-based application that permits spoken interaction in a "relatively natural manner"

But in practice.... (example)

Levels of sophistication

Touch-tone replacement

System Prompt: "For checking information, press or say one."
Caller Response: "One."

Directed dialogue

System Prompt: "Would you like checking account information or rate information?"
Caller Response: "Checking", or "checking account," or "rates."

Natural language

System Prompt: "What transaction would you like to perform?"
Caller Response: "Transfer 500 dollars from checking to savings."

Levels of sophistication

-Flexibility/
+Robustness

Controlled language — limited vocabulary, simple grammar (e.g. command language)

↓

Natural language — huge vocabulary, complex grammar, grammatical variation, ambiguities, unclear sentence boundaries, omissions, word fragments

↓

Natural dialogue — turn-taking, initiative switch, discourse grounding, restarts, interruptions, interjections, speech repairs

+Flexibility/
-Robustness

What‘s involved in language interaction?

Phonology & Phonetics
: speech sounds and their usage

Morphology
: components and structure of words

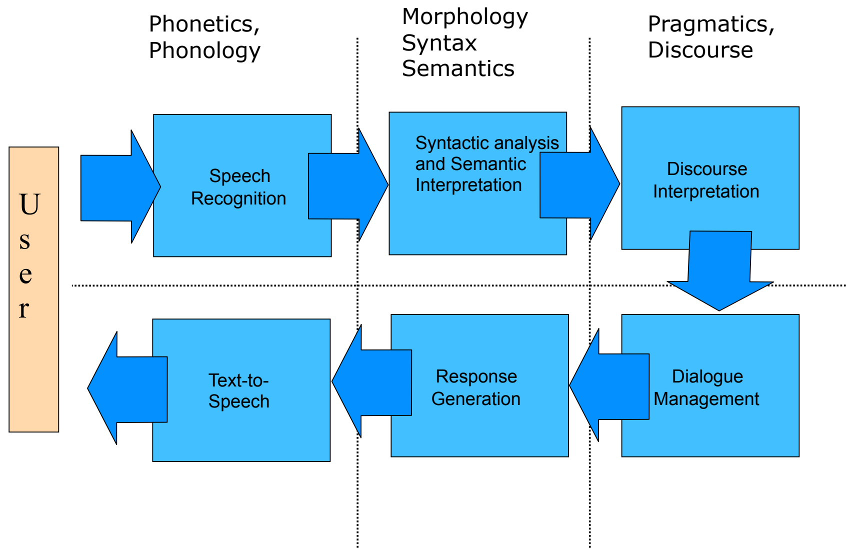
Syntax
: structural relationship between words & phrases

Semantics
: meaning of words (lexical) and word combinations (compositional)

Pragmatics
: language use in context in order to accomplish things (said: „I‘m cold“ → meant: „shut the window“)

Discourse
: larger meaningful connection across linguistic units

Classical structure of SLDS



Classical structure of SLDS

Speech Recognition
: Decode the sequence of feature vectors into a sequence of *words*.

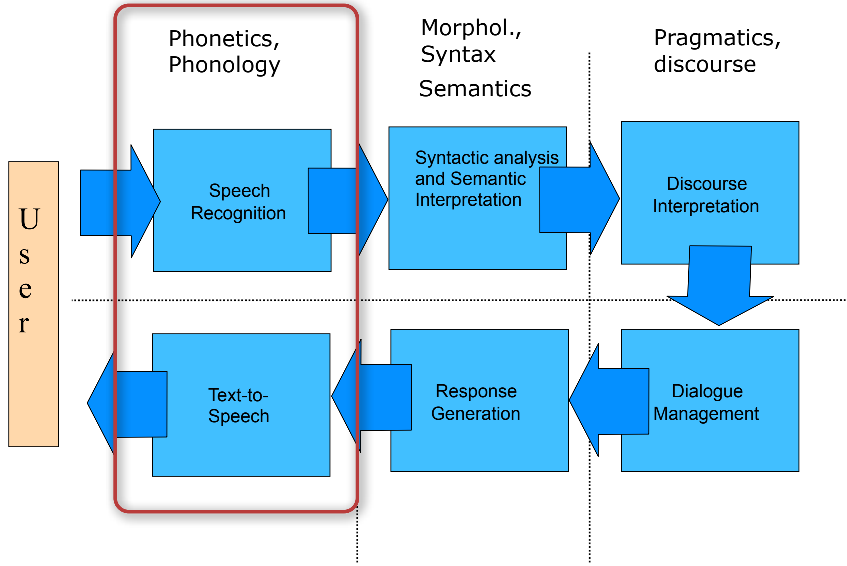
Syntactic Analysis and Semantic Interpretation
: Determine the utterance *structure* and the *meaning* of the words.

Discourse Interpretation
: Understand what the *utterance means* and what the user *intends* by putting it in *context*.

Dialogue Management
: Determine *how to respond* properly to the user intentions.

Response Generation
: Turn communicative act(s) into a *natural utterance.*

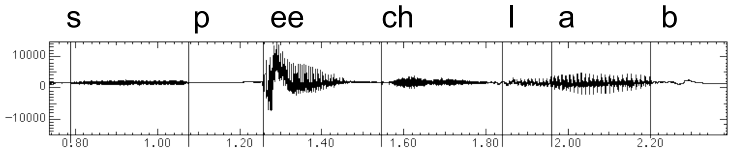
Text-to-speech
: Turn the words into *synthetic speech.*



Starting and end point: acoustic waves

Human speech generates a wave

A wave for the words "speech lab":



Phonetics

study of speech sounds

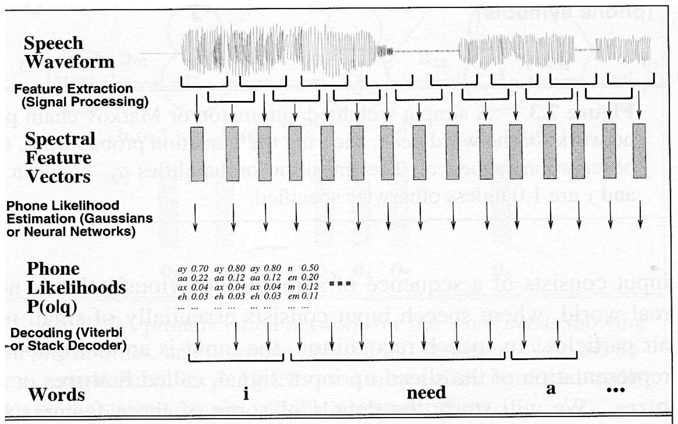
- *Phone* (*segment*) = speech sound (e.g. „[t]")
  - *vowels*, *consonants*
- *Allophone:* different pronounciations of a phone
  - [t] in „tunafish" → aspirated, voicelessness thereafter
  - [t] in „starfish" → unaspirated
- *Diphone*, *triphone*, … = combination of phones
- *Syllables* = made up of vowels and consonants, not always clearly definable („syllabification problem")
- *Prominence* = *Accented* syllables that stand out
  - Louder, longer, pitch movement, or combination
- *Lexical stress* = accented syllable if word is accented
  - „CONtent" (noun) vs „conTENT" (adjective)

Phonology

study of the ways that sounds are used to make meaning

- *Phoneme* = smallest meaning-distinguishing, but *not meaningful* articulatory unit
  - Phones [b] (`bill´) and [ph] (`pill´) discriminate two meanings → different phonemes /b/ und /p/
  - Subsume different elemental sounds under one phoneme, e.g. [p] in `spill´ and [ph] in `pill´ → /p/
- *Phonological rules* = relation between phoneme and its allophones
- Every language has ist own set of phonemes and rules
  - ~40 German phonemes: /p/, /t/, /k/ (plosives); /m/, /n/, /ŋ/ (nasals); /a:/, /a/, /e:/, /ɛ/ (vowels); …

Speech recognition (at a glance)

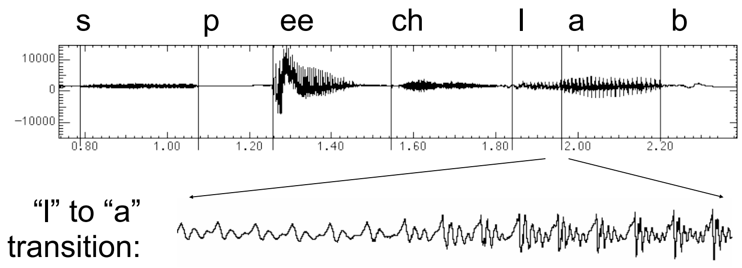


Schematic architecture for a (simplified) speech recognizer.

(Jurafsky & Martin, 2000)

Waveform

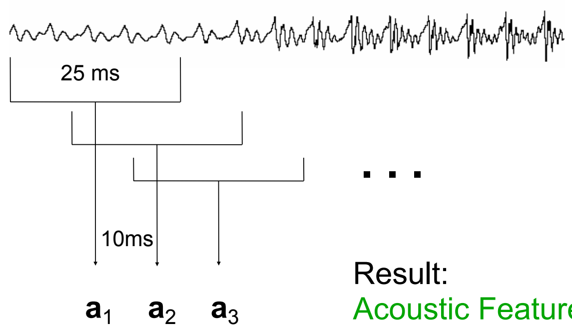
A wave for the words "speech lab" looks like:



Acoustic Sampling

10 ms frame (= 1/100 second)

~25 ms window around frame to smooth signal processing



Result:
Acoustic Feature Vectors

Speech Recognition Problem

The recognition problem: Find most likely sequence **w** of "words" given the sequence of acoustic observation vectors **a**

Use Bayes' law to create a generative model

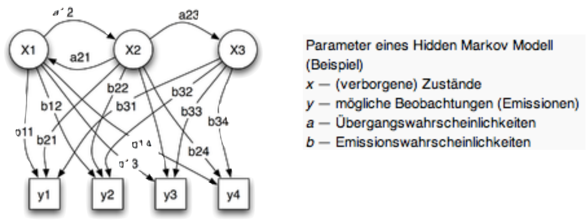
- holds: P(*a*,*b*) = P(*a*|*b*) P(*b*) = P(*b*|*a*) P(*a*)
- Joint probability of *a* and *b* = a priori probability of *b* times the probability of *a* given *b*
- *Bayes rule*: P(*b*|*a*) = P(*a*|*b*) P(*b*) / P(*a*)

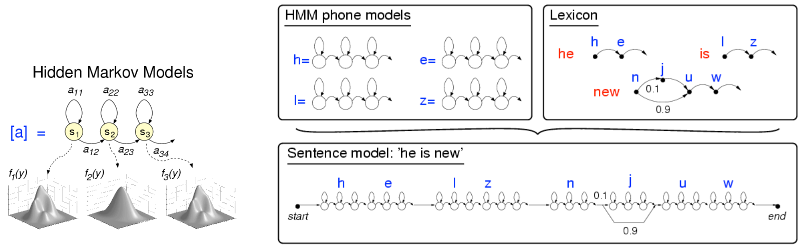
Applied to recognition problem:

- acoustic model: P(**a**|**w**) (→ HMMs for sub-word units)
- language model: P(**w**) (→ Grammars, etc.)
- $\mathrm{ArgMax}_{\mathbf{w}}\ P(\mathbf{w}|\mathbf{a}) = \mathrm{ArgMax}_{\mathbf{w}}\ P(\mathbf{a}|\mathbf{w})\ P(\mathbf{w}) / P(\mathbf{a})$

  $\sim \mathrm{ArgMax}_{\mathbf{w}}\ P(\mathbf{a}|\mathbf{w})\ P(\mathbf{w})$

Hidden Markov Models





Distinguishing features of ASRs

Speaker

- independent vs. dependent
- adapt to speaker vs. non-adaptive

Speech

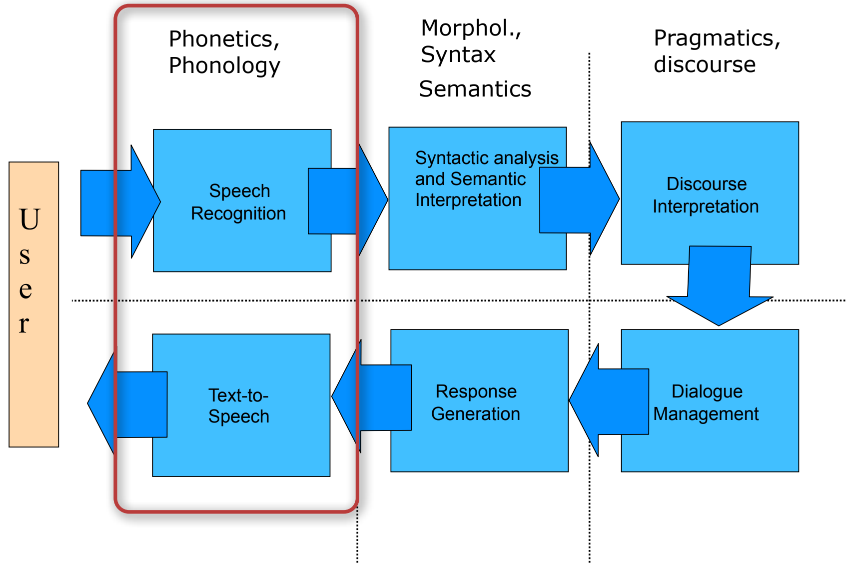
- recognize words vs. verify word hypotheses
- continuous vs. discrete (single words)
- spontaneous vs. read speech
- large vocabulary (2K-200K) vs. limited (2-200)

Acoustics

- noisy environment vs. quiet environment
- high-res microphone vs. phone vs. cellular

Performance

- real time, low vs. high latency
- incremental results vs. final results



Text-to-speech synthesis

Inverse problem: mapping text to wave forms

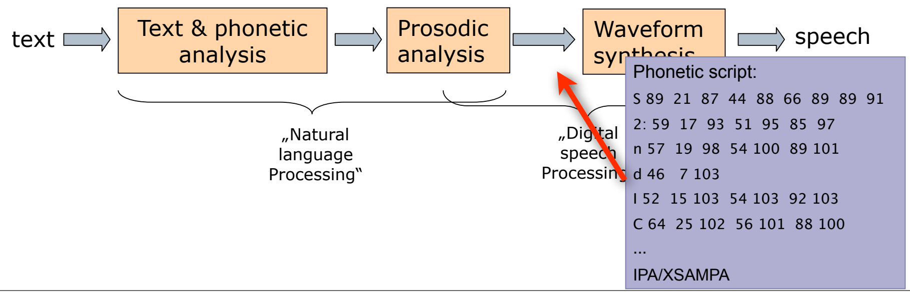
Simplest (and most common) solution

- record prompts spoken by a (trained) human
- produces human quality voice
- limited to number of prompts that can be recorded
- extensions using cut-and-paste or template filling

Text-to-speech synthesis

More flexible synthesis, commonly in three general steps:

1. analyse text and select sound segments
2. determine prosody and how to set it over segments
3. create acoustic waveform signal (synthesis)



Which segments?

Co-articulation = change in segments due to movement of articulators in neighboring segments

Phonemes?
- problematic due to co-articulatory effects

Allophones?
- Variants of a phoneme in specific contexts
- *Example*: Phoneme /p/ → [p] in spill and [ph] in pill

Diphones („Zweilautverbindungen")?
- Diphones start half-way thru 1st phone and end half-way thru 2nd
- critical phone transition is contained in the segment, need not be calculated by synthesizer
- *Example*: diphones for German word „Phonetik": f-o, o-n, n-e, e-t, t-i, i-k

Einheitenlänge	Einheit	#Einheiten (Englisch)	#Regeln	Qualität
kurz	Allophone	60-80	hoch	gering
↓	Diphone	$<40^2$-65^2	↓	↓
	Triphone	$<40^3$-65^3		
	Halbsilben	2K		
	Silben	11K		
	Doppelsilben	<11K2		
	Wort	100K-1.5M		
	Phrasen	∞		
lang	Satz	∞	gering	hoch

Source: E. Andre

Phonetic analysis
from words to segments

Word	Pronunciation
goose	[gus]
geese	[gis]
hedgehog	[ˈhɛdʒ.hɔg]
hedgehogs	[ˈhɛdʒ.hɔgz]

Look up words/wordforms in a pronunciation dictionary
- e.g. CMUdict: ~125.000 wordforms
- + primary stress, secondary stress
  http://www.speech.cs.cmu.edu/cgi-bin/cmudict

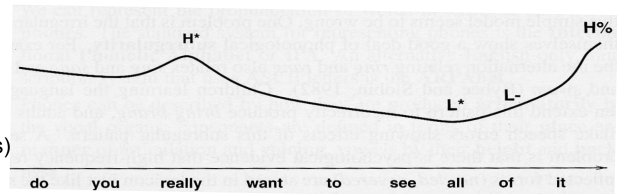
always a lot of unknown words: use letter-to-sound rules
- MITalk (1987): 10.000 rules repository: p – [p]; ph – [f]; phe – [fi]; phes – [fiz]; … … …
- Festival: rules account for co-articulation: [c h] + any consonant = `k´, else `ch´ (`christmas´ vs. `choice´)
- Usually machine learned from large data sets

Prosodic analysis
from words+segments to boundaries, accent, F0, duration

TTS systems need to create proper prosody by adapting...

1. prosodic phrasing/boundaries:
   - Break utterances into units
   - Punctuation and syntactic structure useful, but not sufficient
2. duration of segments:
   - Predict duration of each segment
   - Helps to create prominence
3. intonation/accents on/over segments:
   - Predict accents: which syllables should be accented?
   - Realize as F0 contour („pitch") with special form for accents

ToBI Notation
(Tones and Break Indices)



Waveform synthesis

from segments, f0, duration to waveform

- Signal modeling
  - **Rule-based**
    - Formant synthesis
  - **Data-based**
    - Diphone concat.
    - Unit selection
    - HMM-based
- Physiological modeling, **Articulatory synthesis**

Start with acoustics, rules to create formants

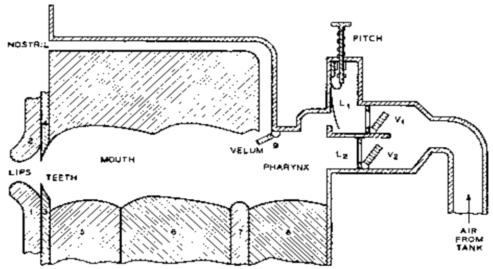
Use databases of stored speech to assemble new utterances

Model movements of articulators and acoustics of the human vocal tract

Articulatory synthesis

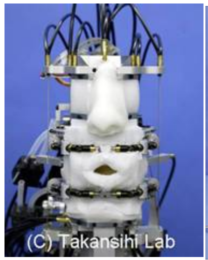
based on physical or (nowadays) computational models of the human vocal tract and the articulation processes occurring there

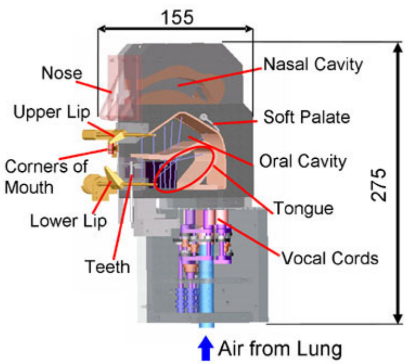
used to be deficient and computationally too demanding, but nowadays get better and used more often

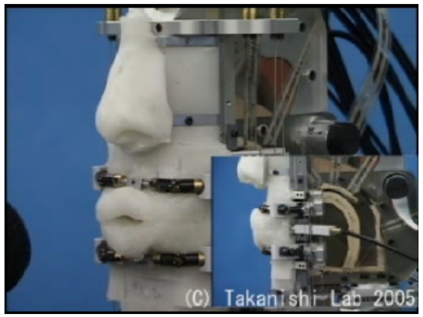


Articulatory synthesis

Talking robots WT-4, WT-5
Waseda University, Tokyo





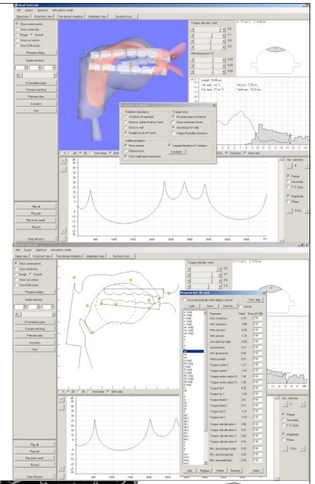


„sasisuseso"

Articulatory synthesis

- Example:
  http://www.vocaltractlab.de

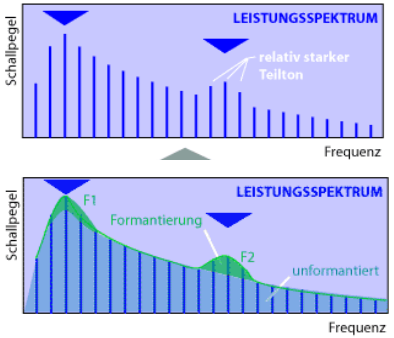
From fMRI images to a 3D model of the human vocal tract, to articulatory speech synthesis

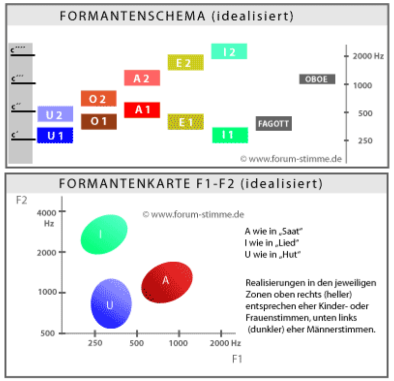


Formant Synthesis

Assumption: Important perceptual information encoded in formants (frequencies with distinct intensity)

First two formants (F1, F2) determine speech perception; sometimes the primary formant is sufficient by itself





www.forum-stimme.de

Formant Synthesis

Rules model relations between tones and acoustic features

Advantages

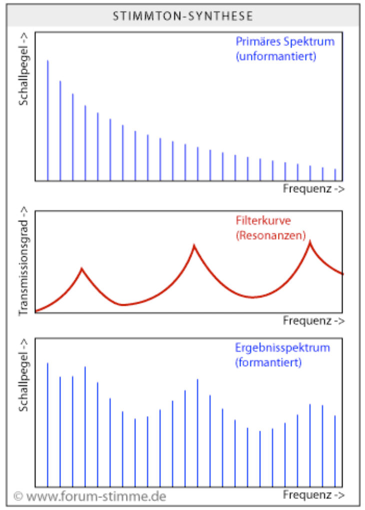
- flexibilty
- not much storage space needed

Disadvantages

- Sounds mechanical
- Complicated rule sets

Common while computers were relatively under-powered

- 1979 MIT MITalk (Allen, Hunnicut, Klatt),
- 1983 DECtalk system, 'Klatt synthesizer'



Data-based synthesis

Almost all current commercial systems use it (1990's-)

Steps:

1. Record basic inventory of sounds (offline)
2. Retrieve sequence of units at run time (run-time)
3. Concatenate and adjust prosody (run-time)

What kind of units?

- Minimize context contamination, but capture co-articulation
- Enable efficient search
- Segmentation and concatenation problems

How to join the units?

- dumb (just stick them together)
- PSOLA (Pitch-Synchronous Overlap and Add), MBROLA (Multi-band overlap and add)

Diphone synthesis

Units = diphones

- Phones are more stable in middle than at the edges

Typically 1500-2000 diphones, need to reduce number

- phonotactic constraints: constraints on the way in which phonemes can be arranged to form syllables
- collapse in cases of no co-articulation

Record one speaker saying each diphone

- "Normalized": monotonous, no emotions, constant volume

Example: MBROLA (Dutoit & Leich, 1993)
http://tcts.fpms.ac.be/synthesis/mbrola.html

Unit selection

One sample of a diphone is often not enough!

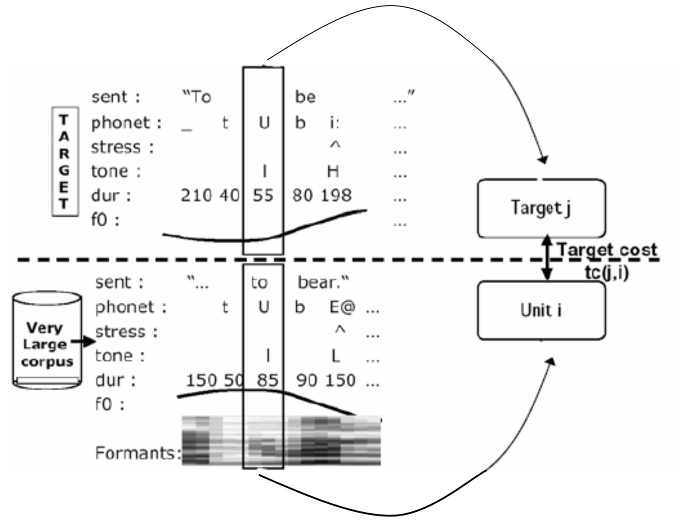
Unit selection:

- Record multiple copies of each unit with different pitches and durations
- How to pick the right units? informed search
- *Example* (Hunt & Black, 1996):
  - Input: three F0 values per phone
  - Database: phones+duration+3 pitch values
  - Cost-based selection algorithm

Non-uniform unit selection

- Units of variable length
- Reduces the need of automatic prosody modeling

Unit selection



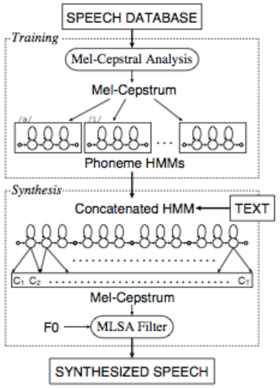
HMM-based synthesis

From a sequence of phonemes and contextual annotation, use HMMs to generate sequences of speech parameters from which a waveform can generated

- aka. Statistical Parametric Synthesis

Parameter forms contain dynamics of

- spectral envelope
- fundamental frequency (F0)
- duration
- aperiodic components (noise)



http://hts.sp.nitech.ac.jp/

- Comparison of state-of-the-art TTS systems http://ttssamples.syntheticspeech.de/deutsch/index.html