Multi-Modal User Interaction

Lecture 2: Talking to Computers (II) and Lip-Reading

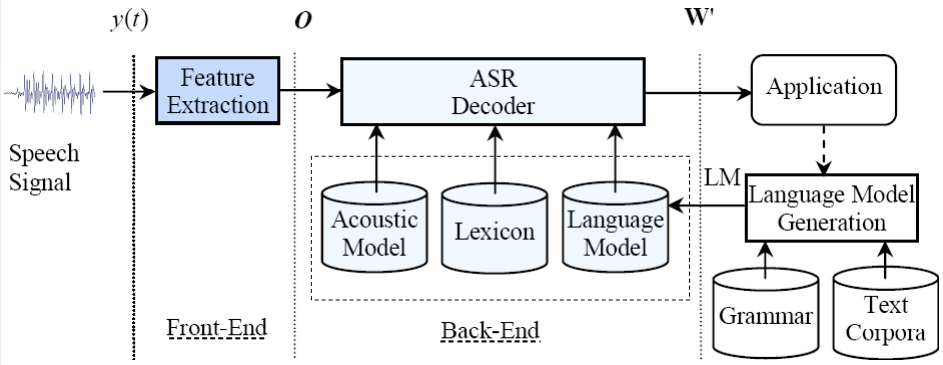
Zheng-Hua Tan

Department of Electronic Systems
Aalborg University, Denmark
zt@es.aau.dk

Outline

- Lexicon and language model
  - Java Speech Grammar Format (JSGF)
- HTK and Sphinx 4
- Mobile speech applications based on Sphinx4
- Lip-reading combined with speech recognition

Speech recognition system



Pronunciation dictionary (lexicon)

SAMPA (Speech Assessment Methods Phonetic Alphabet) is a machine-readable phonetic alphabet.

Danish

- Aalborg Q l b Q:
- café k a f e:
- Paris p A R i: s
- tak t A g

Language modelling for speech recognition

- Speech recognizers seek the word sequence $\hat{W}$ which is most likely to be produced from acoustic evidence A

$$P(\hat{W}|A) = \max_W P(W|A) \propto \max_W P(A|W)P(W)$$

- Speech recognition involves acoustic processing, acoustic modelling, language modelling, and search
- Language models (LMs) assign a probability estimate $P(W)$ to word sequences $W = \{w_1, \ldots, w_n\}$ subject to

$$\sum_W P(W) = 1$$

- Language models help guide and constrain the search among alternative word hypotheses during recognition

(Glass, 2003)

Types of language model

- n-gram
  - based on probabilities of word combinations e.g. bigrams, trigrams
- Finite-state and phrase structure
  - take the form of rules with a left-hand and right-hand side

n-gram language models

- Probability of the sentence S = w_1 w_2 ... w_Q:
  $P(S) = P(w_1\ w_2\ ...\ w_Q) =$
  $P(w_1)P(w_2|w_1)P(w_3|w_1\ w_2)...P(w_Q|w_1\ w_2\ ...w_{Q-1})$
- Conditional word probability:
  $P(w_Q|w_1\ w_2\ ...\ w_{Q-1}) \approx p(w_Q|w_{Q-N+1}\ ...\ w_{Q-1})$
  where N is a constant:
  - Unigram (N=1)
  - Bigram (N=2)
  - Trigram (N=3)
- Google Web 1T 5-gram Corpus! 2006

Finite state grammar (networks)

- Language space defined by word network or graph

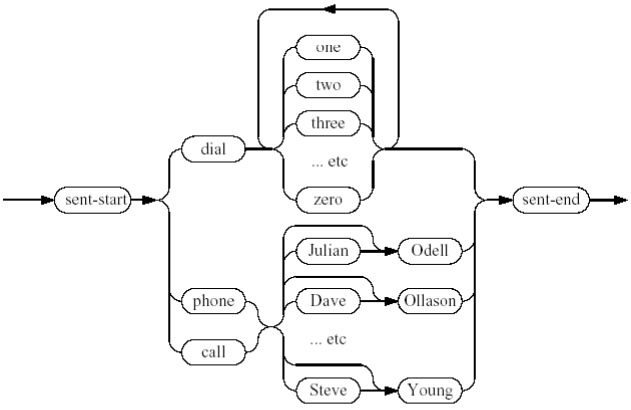


Fig. 3.1 Grammar for Voice Dialling

Rule grammar recogniser

- Grammars determine what the recognizer should listen for and describe the utterances a user may say
- Rule grammar recogniser, i.e. command and control recogniser
- Grammar formats
  - VoiceXML
  - JSGF

Outline

- Lexicon and language model
  - Java Speech Grammar Format (JSGF)
- HTK and Sphinx 4
- Mobile speech applications based on Sphinx4
- Lip-reading combined with speech recognition

Java Speech Grammar Format

Java Speech Grammar Format (JSGF)

- is a platform-independent, vendor-independent textual representation of grammars for use in speech recognition.
- adopts the style and conventions of the Java programming language in addition to use of traditional grammar notations.

Grammar names and declaration

- Each grammar has a unique name that is declared in the grammar header
- Grammar's name must be declared as the first statement of that grammar:
  grammar grammarName
  - simple grammar name e.g.
    - grammar robot;
  - full grammar name (=package name + simple grammar name) e.g.
    - grammar com.acme.politeness;

Rulename

Grammar is composed of a set of rules that define what may be spoken. Rules are combinations of speakable text and references to other rules.

Each rule has a unique rulename:

- Rulename can be written in most of the world's living languages
  - Chinese, Japanese, Korean, European languages…
- Case sensitive
  - <name> and <Name> are different

Comments and grammar header

- /* text */ A traditional comment.
- // text A single-line comment.
- The header format is

#JSGF version char-encoding local;

e.g.

#JSGF V1.0;

Rule definitions

- Grammar body defines rules
  - <ruleName> = ruleExpansion;
  - public <ruleName> = ruleExpansion;
- Weights
  - To indicate the likelihood of each alternative being spoken.
  - <size> = /10/ small | /2/ medium | /1/ large;

Grouping and unary operators

- Grouping
  - <command> = (open | close) (windows | doors);
- Unary operators
  - <polite> = please | kindly | oh mighty computer;
  - <command> = <polite> * don't crash

A rule expansion followed by the asterisk symbol indicates that the expansion may be spoken *zero or more times*. Here a user can say things like "please don't crash", "oh mighty computer please please don't crash", or to ignore politeness with "don't crash".

- <command> = <polite> + don't crash

The plus symbol indicates the expansion may be spoken one of more times.

Tags

- Tags provide a mechanism for grammar writers to attach application-specific information to parts of rule definitions.
- Applications typically use tags to simplify or enhance the processing of recognition results.
- Tag attachments do not affect the recognition of a grammar. Instead, the tags are attached to the result object returned by the recognizer to an application.
- A tag is a unary operator. The tag is a string delimited by curly braces `{}'.
- The tag attaches to the immediate preceding rule expansion. E.g.
  - <rule> = <action> {tag in here}; <command>= please (open {OPEN} | close {CLOSE}) the file;

Hello world application

- Robot control

```
#JSGF V1.0;
/**
 * JSGF Robot Grammar for Hello World example
 */
grammar robot;

public <move> =    (LIFT ARM | STEP FORWARD | SIT
    DOWN | ENTER STAIRS | FETCH THE CUP) * ;
```

Outline

- Lexicon and language model
- HTK and Sphinx 4
- Mobile speech applications based on Sphinx4
- Lip-reading combined with speech recognition

The Hidden Markov Model Toolkit (HTK)

- http://htk.eng.cam.ac.uk/
- A toolkit for Hidden Markov Modeling
- Optimized for speech recognition
- Also used for speech synthesis, character recognition and DNA sequencing general purpose, but...
- Very flexible and complete (always updated)
- Very good documentation

Sphinx

- Speech recognition software.
- Based on HMMs (Hidden Markov Models)
- Four different versions since 1988.
  - Previous versions Sphinx-1, Sphinx-2, Sphinx-3 were programmed in C
  - Latest version (Sphinx-4) was programmed in JAVA

Sphinx-4

- A flexible open source framework for speech recognition
  - Object-oriented design making its integration with other modules simple
  - Support of a wide range of recognition tasks making it dynamically configurable
  - Support of live mode and batch mode recognition
  - Recognizing discrete and continuous speech
  - BSD-style license

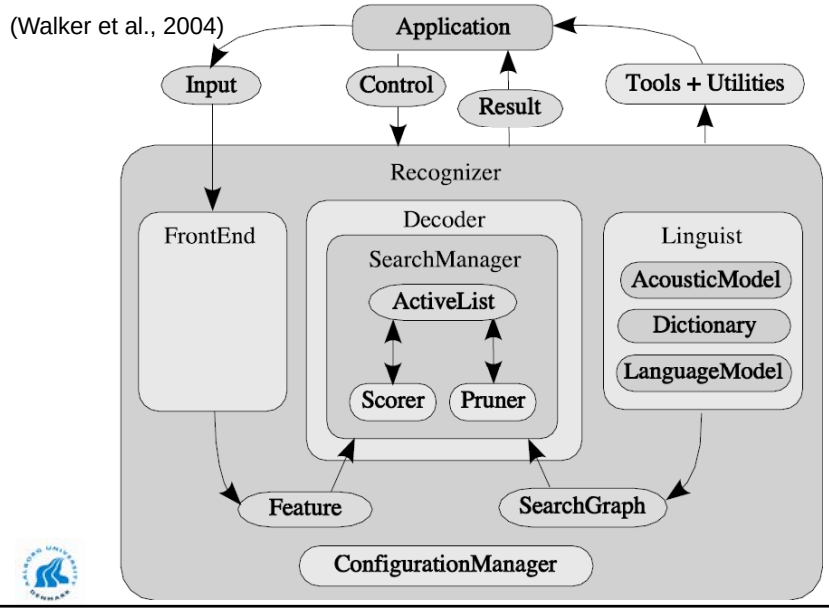
Sphinx-4 performance

(Walker et al., 2004)

Test	*WER*		*RT*		
	Sphinx-3.3	*Sphinx-4*	*Sphinx-3.3*	*Sphinx-4 (1 CPU)*	*Sphinx-4 (2 CPU)*
TI46 (11 words)	1.217	0.168	0.14	0.03	0.02
TIDIGITS (11 words)	0.661	0.549	0.16	0.07	0.05
AN4 (79 words)	1.300	1.192	0.38	0.25	0.20
RM1 (1000 words)	2.746	2.739	0.50	0.50	0.40
WSJ5K (5000 words)	7.323	7.174	1.36	1.22	0.96
HUB-4 (64000 words)	18.845	18.878	3.06	4.40	3.80

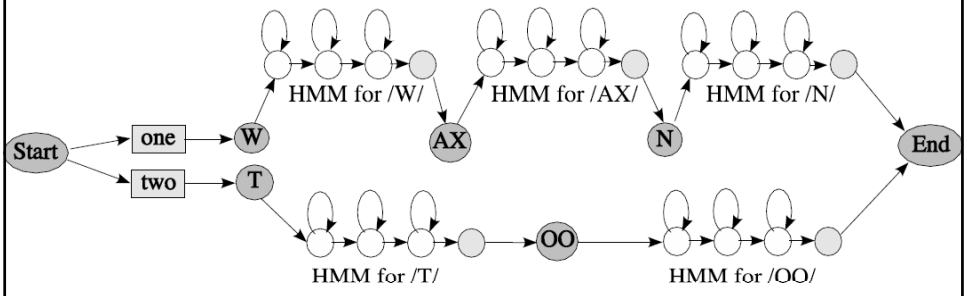
Sphinx-4 decoder framework

(Walker et al., 2004)



Example of search graph

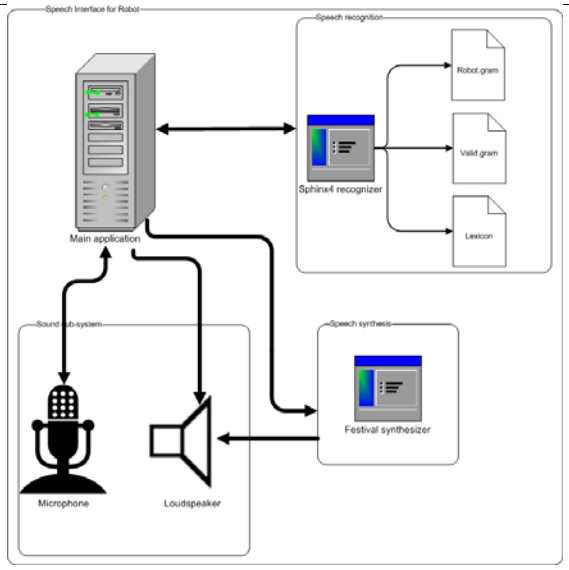
(Walker et al., 2004)



Sphinx 4 @ sourceforge.net

- http://cmusphinx.sourceforge.net/sphinx4/
- To try out the system, download the binary
- for sphinx 4, the setup your environment to support the Java Speech API (JSAPI). Then run the demo:
  - java -mx312m -jar bin/HelloWorld.jar
  - java -jar bin/HelloDigits.jar
  - java -mx312m -jar bin/HelloNGram.jar
  - java -jar bin/ZipCity.jar [-continuous]
  - java -jar bin/WavFile.jar
  - java -jar bin/Transcriber.jar

Speech interface for robot



Speech interface for robot – cont.

- Grammar

```
#JSGF V1.0;
/**
 * JSGF Robot Grammar for Hello World example
 */
grammar robot;

public <move> =    (LIFT ARM | STEP FORWARD | SIT
    DOWN | ENTER STAIRS | FETCH THE CUP) * ;
```

Outline

- Lexicon and language model
- HTK and Sphinx 4
- Mobile speech applications based on Sphinx4
- Lip-reading combined with speech recognition

Mobile technology

The prevalence of mobile devices: being used as digital assistants, for communication or simply for fun.

- Mobile phones: 3.5 billion by 2010
- PDAs, MP3 players, GPS devices, digital cameras

The proliferation of wireless networks: being accessible anywhere, anytime and from any devices.

- 3G, WLAN, Bluetooth, and IP networks
- Free wireless connection for the public

(Tan, 2008)

Mobile technology – cont.

"To the same extent that TV transformed entertainment in the 1960s and the PC transformed work during the 1980s, mobile technology is transforming the way that we will interrelate in the next decade."

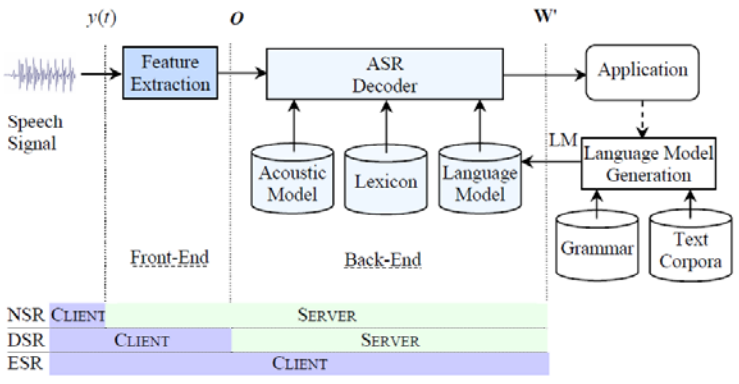
- Michael Gold, SRI Consulting.

When will speech technology transform the way we interact with mobile devices, and what shall be done to make it happen?

Voice Control Demo
Text To Speech (TTS) Demo

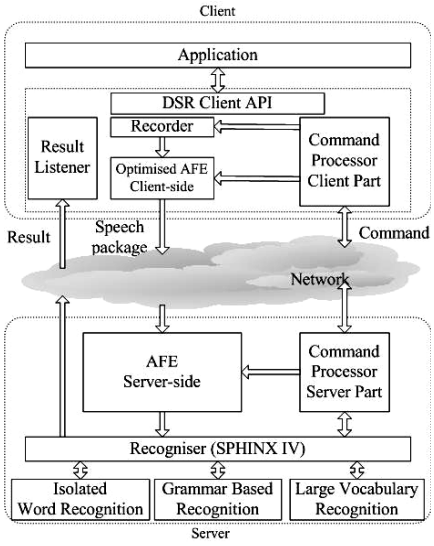
(Tan, 2008)

Architecture of an ASR system



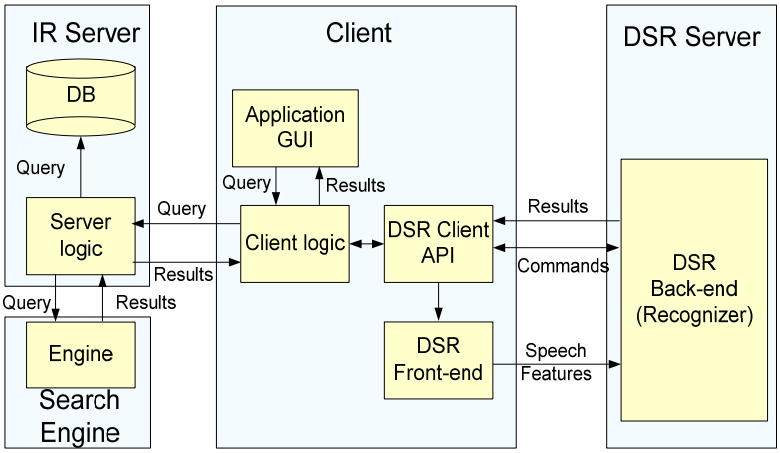
The decision on where to place the ASR components distinguishes three approaches: NSR, DSR and ESR.
It is driven by factors including device and network resources, ASR components complexity and application.

A Configurable Distributed Speech Recognition System



(Xu et al., 2008)

A speech-enabled information retrieval system



A speech-enabled information retrieval system

HCWapp Application for PDA

HCWapp Application for PDA

```
#JSGF V1.0;
grammar jsgf;
<quit> = EXIT | QUIT;
<selection> = SELECT | VIEW | DISPLAY | GET | GO TO;
<tab> = TAB | PAGE | SCREEN;
<application> = PROGRAM | APPLICATION | SOFTWARE;
public <jsgf> =
   [<selection>] ([NEXT] VISITS | [THE] <tab> TWO)   {goto_visits} |
    [<selection>] (PATIENT [DETAILS] | [THE] <tab> THREE)
                                                      {goto_patients} |
    [<selection>] (OPTIONS | [THE] <tab> FOUR)   {goto_options} |
    [<selection>] (HELP | [THE] <tab> FIVE)          {goto_help} |
    (<quit> [[THE] <application>])                  {quit};
```

Speech applications for automotive industry

- Speech in cars: a necessity for safe driving.
  - Eyes off-the-road duration being more than 1.5 sec is considered a distraction.
  - Deployed in 2+ million cars
- Command and control
  - ”Go left. Wait – I mean, go right!” – NO!
  - ”Find me a radio station that plays classical music.” – Yes and soon.
- Telematics (voice activated dialing)
- Navigation (voice destination entry)
- Entertainment systems

Outline

- Lexicon and language model
- HTK and Sphinx 4
- Mobile speech applications based on Sphinx4
- Lip-reading combined with speech recognition

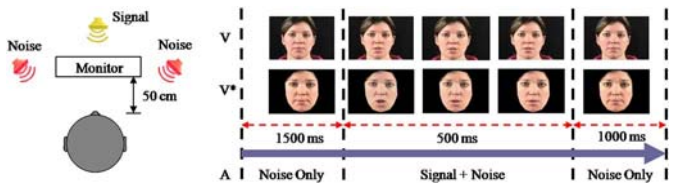
Introduction

Lip-reading aids word recognition

- A widely cited concept: the largest multisensory enhancement is expected when a unisensory stimulus is weakest (the principle of inverse effectiveness).
- A different test: Multisensory word recognition under more natural conditions (without a checklist) using human subjects.

Lip-reading aids word recognition – cont.

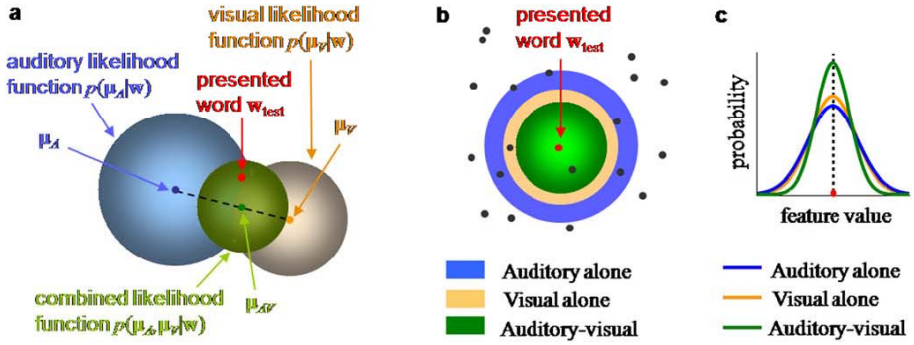
- Multisensory word recognition: experimental set-up and timing of audio-visual stimuli



Stationary acoustic noise,
3 Hz ~ 16 kHz

(Ma et al. 2009)

Bayesian model of AV word recognition

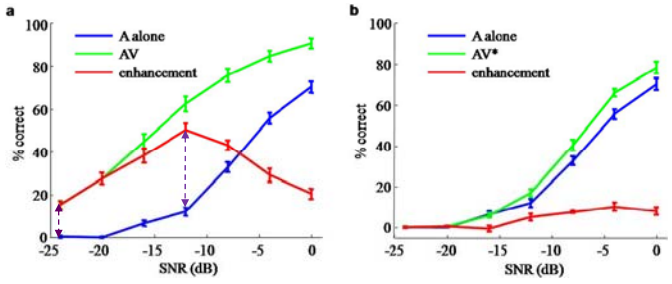


a. Inference process on a single multisensory trial. b. Across many repetitions of the test word, the estimates will form a distribution centered at the test word. Since the distributions "cover" many words, errors will be made. Note the different interpretations of the discs in a and b: single-trial likelihood functions, versus estimate distributions across many trials. c. Side view of the estimate distributions in b. The AV estimate distribution is sharper, leading to fewer errors.

(Ma et al. 2009)

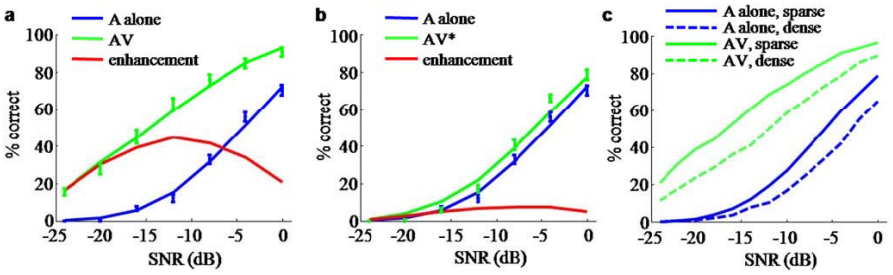
Behavioral performance in open-set word recog.

- Multisensory word recognition under more natural conditions (without a checklist), maximal gain was found not at low, but at intermediate signal-to-ratios.



(Ma et al. 2009)

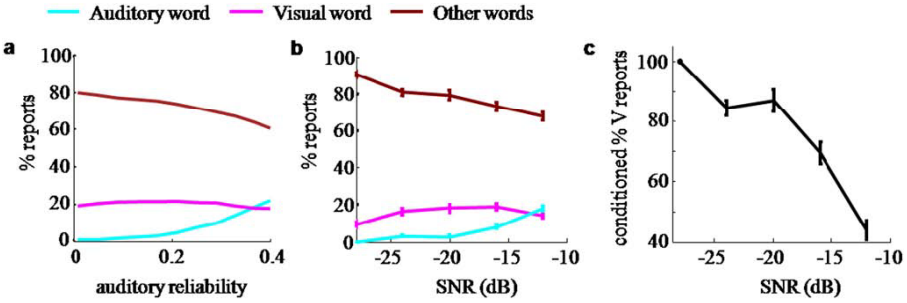
Bayesian model describing human perform.



A Bayesian model of speech perception can describe human identification performance. A vocabulary of size N = 2000 was used. a: Data (symbols) and model fits (lines) for A-alone and AV conditions. The red line is the multisensory enhancement obtained from the model. b: Same for impoverished visual information (AV*). c: Words in high-density regions are harder to recognize.

(Ma et al. 2009)

Incongruent audio-visual stimuli

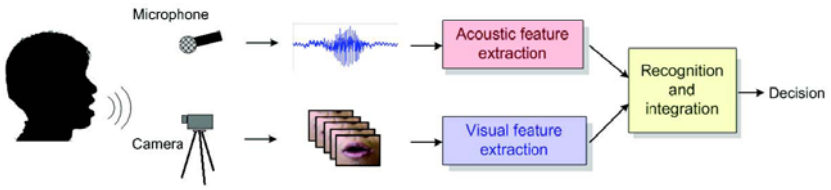


a. Illustration of the Bayesian prediction. As auditory reliability increases, the percentage reports of the visual word reaches a maximum before it eventually decreases. b. Experimental test of the Bayesian prediction. c. Reports of the visual word as a percentage of the total reports of either the auditory or the visual word, computed from the data shown in b.

(Ma et al. 2009)

Audio-visual speech recognition system

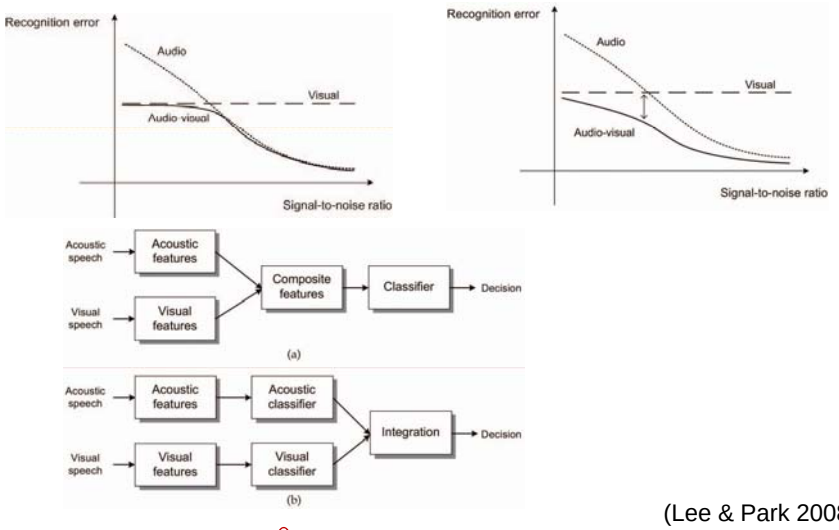
- General procedure



What is the **error rate** for isolated digit recognition by computer lip-reading?

(Lee & Park 2008)

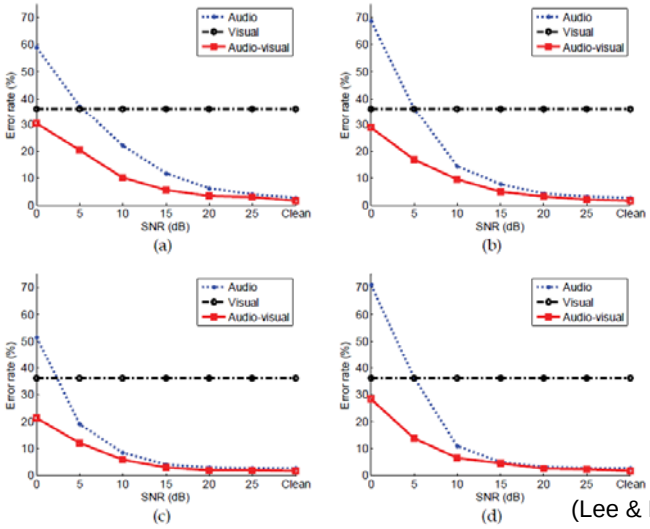
Challenges and fusion approaches



(Lee & Park 2008)

$$C^* = \arg\max_i \left\{ \gamma \log P(O_A \mid \lambda_A^i) + (1-\gamma) \log P(O_V \mid \lambda_V^i) \right\}$$

Recognition performance for digits



(Lee & Park 2008)

Fig. 12. Recognition performance of the unimodal and the bimodal systems in error rates (%) for the DIGIT database. (a) WHT. (b) F16. (c) FAC. (d) OPR.

49

Optical information stream features

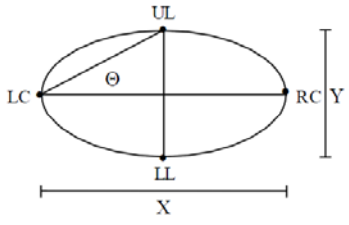


Fig. 1. *The 2-Parameter Lip Contour Model*

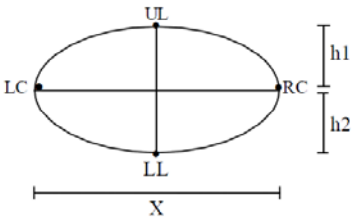


Fig. 2. *The 3-Parameter Lip Contour Model*

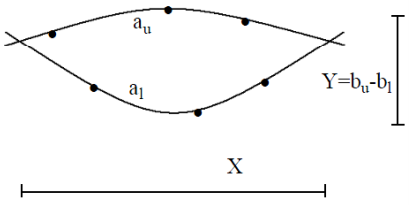


Fig. 3. *The Proposed Parabolic Lip Contour Model*

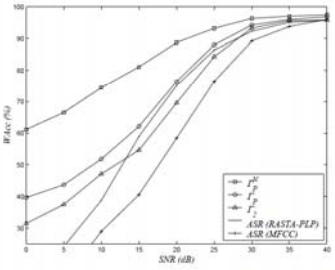


Fig. 4. *Word Recognition Rate of the AV-ASR System Operating With Various High-Level Models.*

50

(Borgstr¨om & Alwan, 2008)

Summary

- Lexicon and language model
- HTK and Sphinx 4
- Mobile speech applications based on Sphinx4
- Lip-reading combined with speech recognition