

On the computational aspects of sign language recognition

Christian Vogler

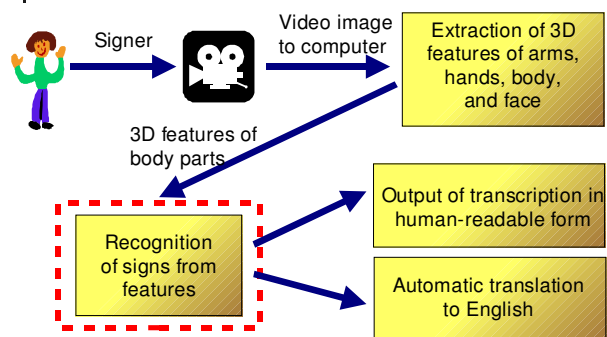
Gallaudet Research Institute

Overview

- **Problem statement**
- Basic probabilistic framework
- Recognition of multiple channels
- Recognition features
- Discussion

2

What is ASL recognition?



3

What makes it hard?

- Sign language recognition is hard:
- Language modeling issues
- Computational issues
 - How does recognition actually work?
 - How to represent data?

4

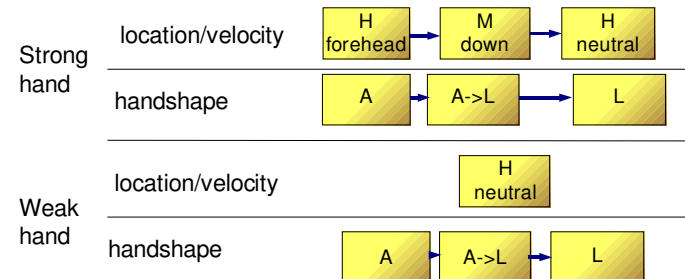
Basic modeling principles

- We break down signs into phonemes
- We model handshape and hand movements in independent channels
- This helps deal with the complexities of sign languages

5

Modeling example

- Example: BROTHER with independent channels



6

Overview

- Problem statement
- **Basic probabilistic framework**
- Recognition of multiple channels
- Recognition features
- Discussion

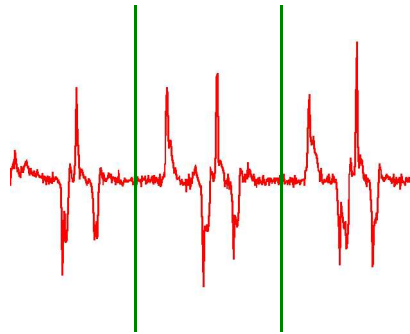
7

Basic Recognition Principles

- We start with the **data signal**
- In our case: Collection of
 - 3D positions of hands
 - 3x3 orientation matrices of hands
 - Joint angles of fingers
 - Abduction (spread) angles of fingers
- The first problem: Variability
- No two instances of a sign look exactly alike

8

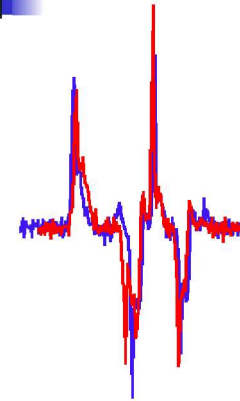
Variability



- 3 repeated instances of sign for "FATHER"
- Structurally similar
- But details are different

9

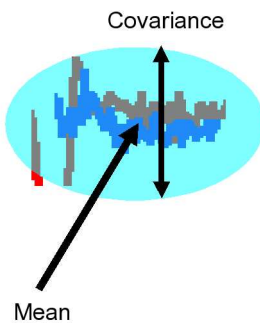
Variations



- Two examples overlaid
- We need to account for variations in both:
 - Space
 - Time

10

Spatial variations



- Model signal with Gaussian probability density
- The closer the signal is to the mean, the higher the probability that the match is correct

11

Temporal variations

- Signers can sign faster or more slowly
- Variations in when exactly each movement occurs
- Speech recognition uses hidden Markov models (HMMs) to deal with them

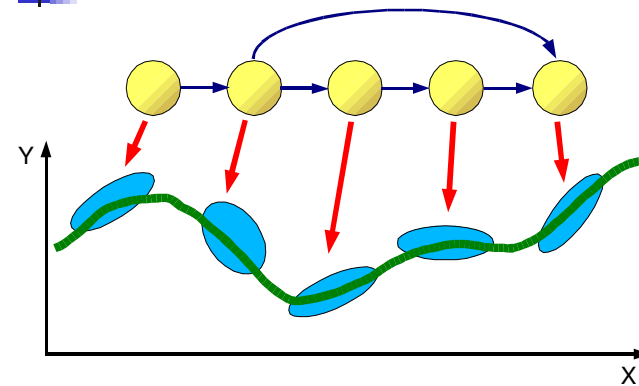
12

Hidden Markov models

- State-based statistical model
 - System is always in some state
 - At discrete time intervals, takes transition to another state
- Probabilistic transitions
 - Accounts for temporal variations
- Each state has output probability distribution
 - Here: Gaussian density mixture
 - Accounts for spatial variations

13

HMM example



14

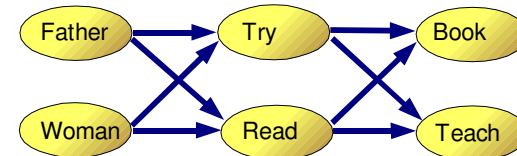
How to use HMMs

- HMMs can generate a signal
- For recognition, consider the converse:
 - What is the probability that the HMM generated signal?
 - Which state sequence generated it?
- Answer to these questions defines continuous recognition problem
- HMM probabilities are trained

15

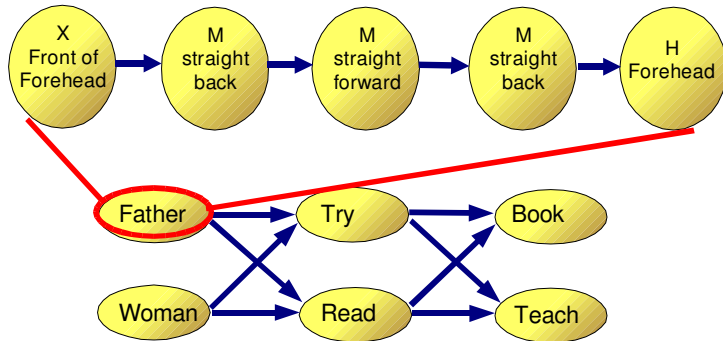
Continuous recognition

- Chain HMMs into network
- Match network to signal
- Find most likely state sequence through network



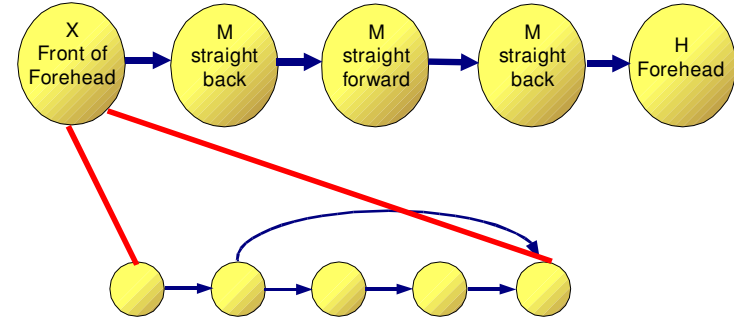
16

Continuous recognition



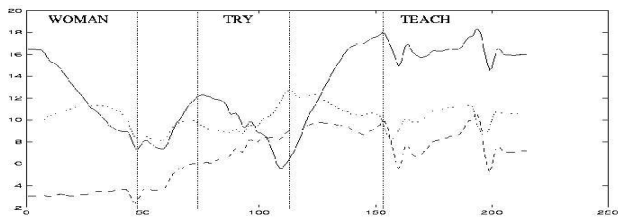
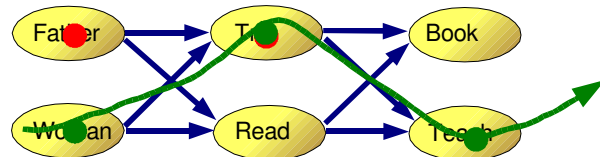
17

Continuous recognition



18

Token passing algorithm



19

Overview

- Problem statement
- Basic probabilistic framework
- **Recognition of multiple channels**
- Recognition features
- Discussion

20

Token passing – Formally

- The token passing algorithm finds

$$\max_{Q_1, \dots, Q_t} \prod_{i=1}^t a_{Q_{i-1}Q_i} b_{Q_i}(O_i)$$

$$= \max_{\mathcal{Q}} P(\mathcal{O}, \mathcal{Q} | \lambda)$$

b_{Q_i} Gaussian density function in state Q_i

O_t Data signal at time t

$a_{Q_{i-1}Q_i}$ Transition probability from state to state

21

Alternative form

- We prefer to write the formula in logarithmic form instead
- Easier to manipulate
- Faster to compute

$$\max_{\mathcal{Q}} P(\mathcal{O}, \mathcal{Q} | \lambda) = \max_{Q_1, \dots, Q_t} \prod_{i=1}^t a_{Q_{i-1}Q_i} b_{Q_i}(O_i) \approx$$
$$\max_{Q_1, \dots, Q_t} \sum_{i=1}^t \log a_{Q_{i-1}Q_i} b_{Q_i}(O_i)$$

22

Important point

- So far, just like speech recognition ...
- But how do we add multiple channels?
- The recognition algorithm maximizes a product of **independent** random variables
 - Each data frame is independent from the others
 - Each state transition is independent from the others
- So: Joint probability of channels is just product of marginal (individual channel) probabilities

23

Parallel HMMs

- This extension formalizes Parallel HMMs
- Instead of computing:

$$\max_{\mathcal{Q}} \log P(\mathcal{O}, \mathcal{Q} | \lambda)$$

Compute over all channels c :

$$\max_{Q^{(1)}, \dots, Q^{(c)}} \sum_{c=1}^C \log P(\mathcal{O}^{(c)}, \mathcal{Q}^{(c)} | \lambda^{(c)})$$

Recall that $\log(ab) = \log a + \log b$

24

Probability combination

- Essentially, searches an HMM network in parallel for each channel
- When to multiply probabilities?
 - Split up sequence into weighted contributions from individual signs

$$\begin{aligned} & \max_{\mathcal{Q}^{(1)}, \dots, \mathcal{Q}^{(c)}} \sum_{c=1}^C \log P(\mathcal{O}^{(c)}, \mathcal{Q}^{(c)} | \lambda^{(c)}) \\ & = \max_{\mathcal{Q}^{(1)}, \dots, \mathcal{Q}^{(c)}} \sum_{w=1}^W \sum_{c=1}^C \omega_w^{(c)} \log P(\mathcal{O}_w^{(c)}, \mathcal{Q}_w^{(c)} | \lambda^{(c)}) \end{aligned}$$

25

Probability combination

- What does this mean?
 - We can combine the partial probabilities after each sign (or each phoneme)
 - Helps constrain the search through the parallel networks
- Another constraint is needed:
 - Paths from the channels must be consistent
 - That is, they must touch the same sequence of signs

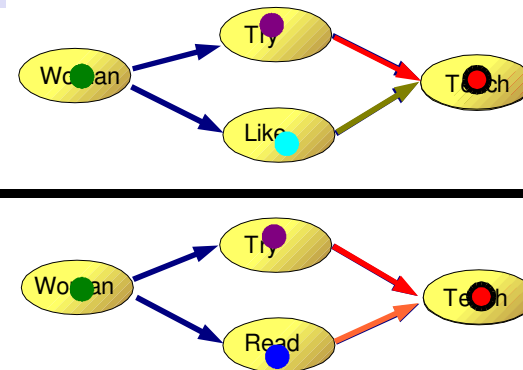
26

Channel combination constraints

- Enforce through path identifiers
 - Assign unique path id to tokens
 - Tokens have same path id iff they touch the same sequence of signs
 - Combine only probabilities of tokens with same path id
- But then maximum joint probability no longer maximizes marginal probabilities
 - Keep multiple hypotheses

27

PaHMM token passing



28



Summary

- Recognition algorithm is rather complicated, compared to basic speech recognition algorithm
 - Extra bookkeeping for path ids, consistency
- Still, reasonably efficient: $O(HC \times N^2 T)$
- Linear in number of hypotheses
- Linear in channels, number of frames
- Quadratic in number of HMM states

29



Overview

- Problem statement
- Basic probabilistic framework
- Recognition of multiple channels
- **Recognition features**
- Discussion

30



Data representation

- Representation of the data is just as important as the recognition algorithm
- Intuition:
 - “Move up 1 inch. Turn left 90 degrees. Move 1 inch. Turn left 90 degrees. Move 1 inch. Turn left 90 degrees. Move 1 inch
 - A square with a side of 1 inch
- Both descriptions refer to the same object
- Clearly, second one easier to grasp

31



Intuition, continued

- The first example is a localized description.
 - No sense of the “big picture”
 - We can infer big picture, because we know the history of steps (move, turn, move, turn)
 - HMMs cannot!
 - Recall that successive data frames are independent
 - So HMMs “lose” large parts of the history
- The second description shows the “big picture”

32



Local features

- The computational equivalent of localized descriptions (“features”):
 - 3D positions
 - 3D velocities
 - Finger joint angles
 - Abduction angles
- We have these data, but none show what happens at a global level

33



Global features

- Some global descriptions
 - Trajectory along line, arc, plane, ...
 - Finger bent, extended
 - Hand open, closed
- Problem: How do we translate these qualitative descriptions into numbers?

34



Trajectories

- For example, this is roughly a line.
- The covariance is large in the x direction
- Small in the y direction

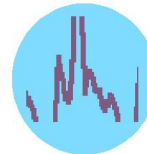


35



Trajectories

- For a plane movement:
- The covariance is large in both directions



36

Global trajectory representation

- Compute 3x3 covariance matrix of 3D positions over a window of 15-30 data frames
- Compute eigenvalues of matrix
- Largest two eigenvalues specify proportions of covariance
- Line: 1st eigenvalue large, 2nd small
- Plane: Both eigenvalues large
- Etc ...

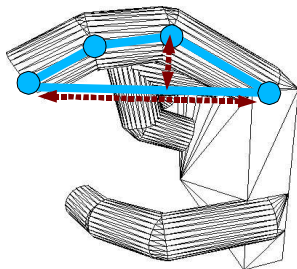
37

Handshape representation

- Take a clue from ASL linguistics here
- Handshape can be characterized by degree of openness of a finger
- Open = finger is extended
- Closed = finger is bent
- How to represent with numbers?

38

Handshape features



- Measure of **openness** of a finger
- Use height and base of quadrilateral
- Open: small height, large base
- Closed: the opposite

39

Features: Summary

- For the final recognition features:
 - Use mix of local & global
 - 3D positions, velocities, eigenvalues
 - Degree of openness, abduction angles
- This is just the beginning
- There is much potential for improvement in the data representation
- Speech recognition is 20 years ahead of us in this respect

40



Overview

- Problem statement
- Basic probabilistic framework
- Recognition of multiple channels
- Recognition features
- **Discussion**

41



Experimental validation

- 500 signed ASL sentences
- 22-sign vocabulary
- Collected with MotionStar motion capture system (both hands), and cyberglove (right hand only)
- Continuous sentences, no pauses between signs

42



Feature comparison

- Do global features help?
- Yes!
- Movement channel:
 - 93% accuracy instead of 90%
- Handshape channel:
 - 95% accuracy instead of 83%

43

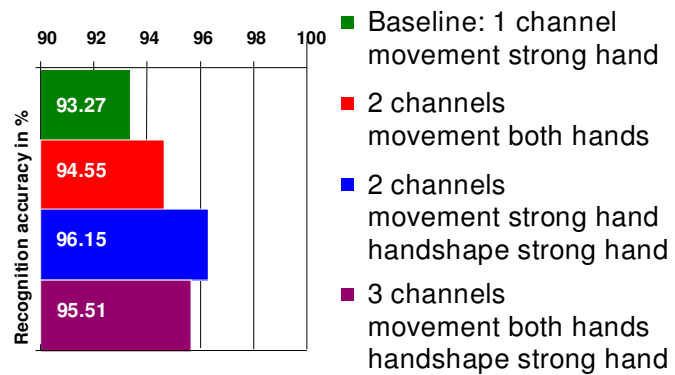


Effect of multiple channels

- PaHMM experiments with 1, 2, and 3 channels
- Basic question: Is modeling the channels independently from one another reasonable?
- Answer is important for large-scale systems

44

Multiple-channel experiments



45

Discussion

- 2 channels definite improvement
- 3 channels did not yield improvement
- Possibly the data set is too small?
 - No signs where the handshape is the only distinguishing feature
- More work with larger data set is needed

46

For more information

- Christian.Vogler@gallaudet.edu
- <http://gri.gallaudet.edu/~cvogler/>

47