Tracking Using Dynamic Programming for Appearance-Based Sign Language Recognition

P. Dreuw, D. Rybach, T. Deselaers, D. Keysers, and H. Ney
Human Language Technology and Pattern Recognition

Introduction

- Automatic Sign Language Recognition (ASLR):
  - ▷ no special hardware, inhomogeneous background, occlusions, ...
  - ▷ important features for ASLR:
    - ∘ hand-shapes
    - ∘ orientation and movement of the hands, arms or body
    - ∘ facial expressions
    - ∘ lip-patterns
  - ▷ tracking of head and hands is hard
  - ▷ problems:
    - ∘ hands are signing in front of the face
    - ∘ hands are moving very fast and abrupt
- We avoid preliminary decisions and propose to use the same techniques that are successfully applied in automatic speech recognition (ASR).

Tracking

- Tracking is necessary to extract sign language features. (hand position relative to the body, facial expressions, ...)
- Possible features
  - ▷ motion, background subtraction
  - ▷ skin, skin-motion
  - ▷ template matching, eigenfaces
- Environment problems: features are not correct and can only be used to produce candidate regions

Dynamic Programming Tracking

- Idea: prevent taking possibly wrong local decisions
- How: tracking is done at the end of a sequence by tracking back the decisions to reconstruct the best path
- the best path is the path with the highest score wrt. a given scoring function
- 2 Steps:
  1. Score calculation: calculate a global score $S(t, x, y)$ and a back-pointer table B for the best tracking until time step t which ends in position (x, y)
  2. Traceback: reconstruct the best path $t \rightarrow (x, y)$ using S and B

1. Step: Score calculation

- ▷ local score: For each pixel (x, y) of a frame X_t at time step $t = 1, ..., T$ a score $s(t, x, y)$ is calculated
- ▷ bookkeeping at time t of the best predecessors at time $t-1$ within a neighborhood of position (x, y)

2. Step: Traceback

- ▷ full traceback: starts from last frame of the sequence at time T using $c_T = \text{argmax}_{x,y} S(T, x, y)$
- ▷ best tracking center c at time $t-1$: $c_{t-1} = B(t, c_t = (x', y'))$

Recursive equation for DPT:

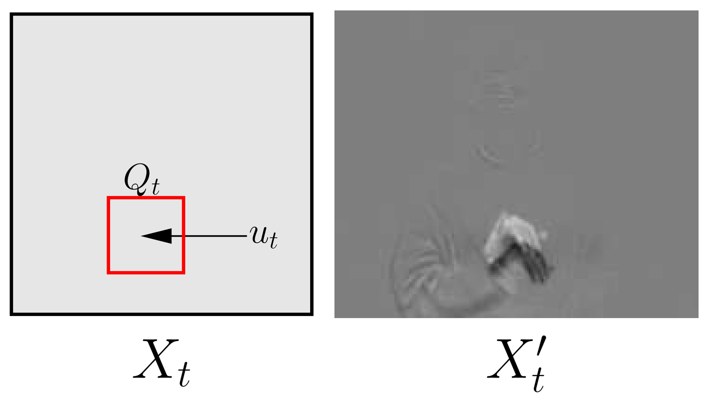
$$S(t,x,y) = \max_{x',y' \in M(x,y)} \{(S(t-1,x',y') - \mathcal{T}(x',y',x,y)\} + s(t,x,y)$$
$$B(t,x,y) = \operatorname*{argmax}_{x',y' \in M(x,y)} \{(S(t-1,x',y') - \mathcal{T}(x',y',x,y)\}$$

$M(x, y)$ is the set of possible predecessors of point (x, y)
$\mathcal{T}(x', y', x, y)$ is the jump-penalty from point (x', y') in the predecessor image to point (x, y) in the current image

Scoring Functions

- Local scoring functions
  - ▷ example: track moving parts, i.e. maximize motion

$$u := (x,y)$$
$$Q := \left\{(x,y) : \begin{array}{l} -w \leq x \leq w, \\ -h \leq y \leq h \end{array}\right\}$$
$$Q_t := \{u_t + u : u \in Q\}$$
$$X'_t[u] := X_t[u] - X_{t-1}[u]$$



$$\operatorname*{argmax}_{u_1^T} \left\{ \sum_{t=1}^{T} \left(\sum_{u \in Q_t} (X'_t[u])^2 + \sum_{u \in Q_{t-1}} (X'_t[u])^2 - \lambda \|u_t - u_{t-1}\|^2 \right) \right\}$$

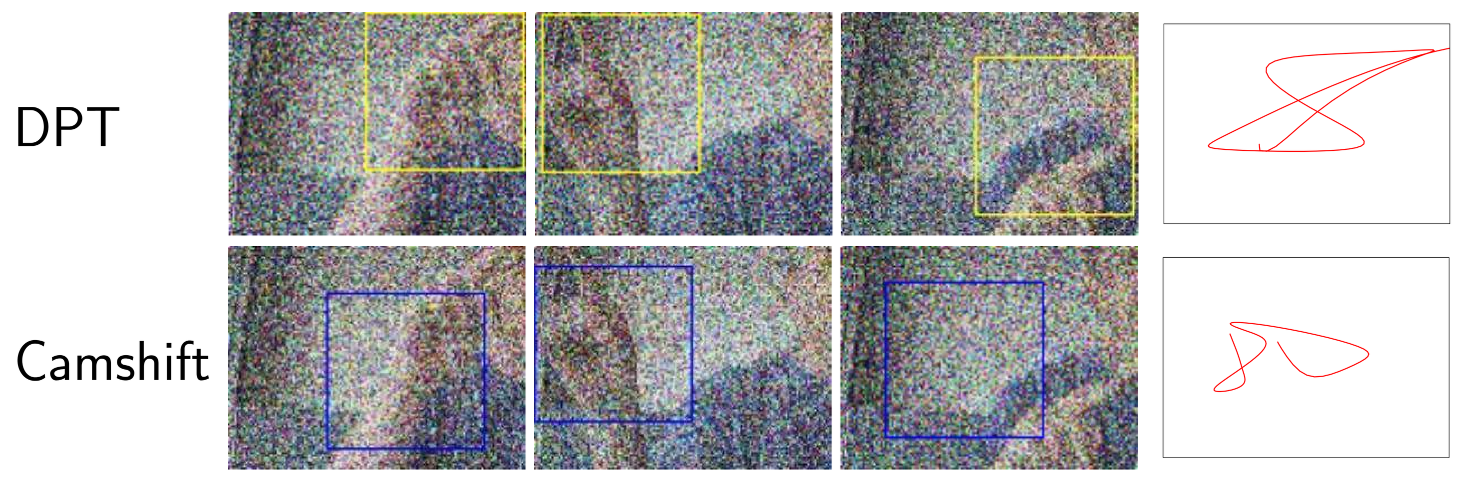
- Additional scoring functions/features:
  - ▷ skin color
  - ▷ Eigenfaces
  - ▷ special distance measures like Tangent distance or IDM
- Example: combination of skin color score s_c and Eigenfaces s_f for head tracking

$$s(t,x,y) = (1-w) \cdot s_c(t,x,y) + w \cdot s_f(t,x,y)$$

- Jump penalty functions
  - ▷ Euclidean distance: $\mathcal{T}(x', y', x, y) = \lambda \cdot \sqrt{(x-x')^2 + (y-y')^2}$
  - ▷ L_1-norm, complex motion models, ...

Experimental Results

- Tracking under noisy circumstances



- Successful tracking of head and dominant hand for ASLR

Conclusion

- if no pruning is used, the optimal path is guaranteed to be found
- the proposed tracking algorithm enables to track a target disregarding information gaps (e.g. due to occlusions)
- advantages under noisy circumstances

Outlook

- integrate further scoring functions which don't require any priori knowledge of the target
- combine tracking and recognition
- extend tracking framework:
  - ▷ multiple objects
  - ▷ automatic traceback
- apply tracking in different areas, e.g. PETS

Lehrstuhl für Informatik 6 – Computer Science Department – RWTH Aachen University – Aachen, Germany
{dreuw,rybach,deselaers,keysers,ney}@cs.rwth-aachen.de
http://www-i6.informatik.rwth-aachen.de