Efficient Approximations to Model-based Joint Tracking and Recognition of Continuous Sign Language

Philippe Dreuw, Jens Forster, Thomas Deselaers, and Hermann Ney
Human Language Technology and Pattern Recognition Group
RWTH Aachen University, Aachen, Germany
<lastname>@cs.rwth-aachen.de

Abstract

We propose several tracking adaptation approaches to recover from early tracking errors in sign language recognition by optimizing the obtained tracking paths w.r.t. to the hypothesized word sequences of an automatic sign language recognition system. Hand or head tracking is usually only optimized according to a tracking criterion. As a consequence, methods which depend on accurate detection and tracking of body parts lead to recognition errors in gesture and sign language processing. We analyze an integrated tracking and recognition approach addressing these problems and propose approximation approaches over multiple hand hypotheses to ease the time complexity of the integrated approach. Most state-of-the-art systems consider tracking as a preprocessing feature extraction part. Experiments on a publicly available benchmark database show that the proposed methods strongly improve the recognition accuracy of the system.

1. Introduction

Hand tracking for sign language recognition is a challenging problem. Frequently, the hands are signing in front of the face, overlap, and may temporarily disappear. In early work on sign language recognition, the problem of hand tracking is facilitated by using special gloves [1, 15]. Other systems require the user to start every gesture from a predefined 'home' position [10].

However, the biggest problem with most work on sign language recognition is that only the recognition of isolated signs is considered [15, 16]. In contrast to these works, here we work on the recognition of *continuous* sign language. For the recognition of sign language the hand is the part of the image that is moving most [3, 18]. Most approaches addressing the recognition of gestures and sign language use a two-step procedure, where in the first step the hand is tracked and in the second step the classification recognition is done [7, 11]. A problem with this approach is that possible tracking errors from the first stage might be impossible to recover in the recognition phase and thus ideally a joint tracking and recognition procedure is used to fuse these steps. Here, we present a method that integrates tracking and recognition into one step which is computationally very complex. Therefore, we present some approximations to this method to reduce the computational demands.

In particular, our proposed rescoring and tracking path adaptation can be applied to any tracking based features, regardless of the chosen tracking method or the features extracted from those tracking regions.

Related Work. Tracking adaptation by learning has been recently addressed e.g. based on a spatial-color mixture appearance model for particle filters [8, 17], or tracking by model-building and detection as presented in [14]. Many of the proposed tracking methods fail if hands are moving abruptly such that the transformations between two frames fall out of the learned or assumed range. Furthermore, the model based approaches are rather detection-based methods, i.e. the resulting path is optimized on a frame-level.

Here, we propose a global and model-based path optimization w.r.t. a word sequence. We present a recognition framework that allows for fully integrated recognition and tracking where the tracking decision is withheld until the recognition phase and explicitly optimized according to recognizing a sentence rather than to optimize some heuristic tracking criterion. Since the computational demands for the proposed procedure are very high, we additionally propose some approximations which greatly ease the computational burden. Second, in sign language recognition, the lack of available data [6, 16] is addressed by using virtual training samples from the existing data by cropping several regions-of-interest for each frame of a video sequence.

2. System Overview & Features

Sign Language Recognition. In a vision-based system, tracking-based features have to be extracted at each time step $t = 1, \ldots, T$ at unknown positions $u_1^T := u_1, \ldots, u_T$

978-1-4244-2154-1/08/$25.00

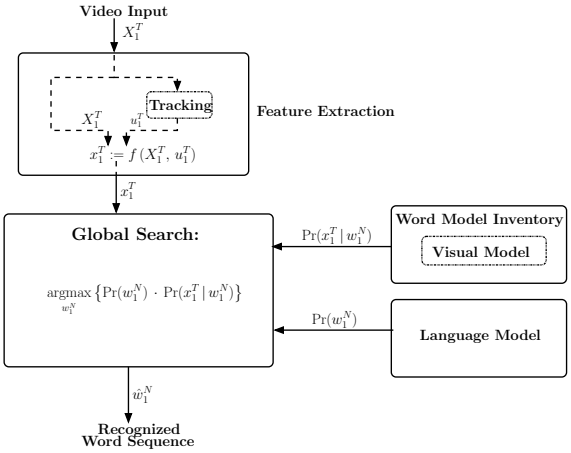


Figure 1: Bayes' decision rule used in ASLR with tracking framework and feature extraction as a pre-processing step.

in a sequence of images $X_1^T := X_1, \ldots, X_T$, with e.g. $x_t = f(X_t, u_t)$ a hand patch feature extracted at position $u_t = (x, y)$ from frame X_t in the image observation sequence X_1^T.

In an automatic sign language recognition (ASLR) system for continuous sign language, we are searching for an unknown word sequence w_1^N, for which the temporal sequence of features $x_1^T := x_1, \ldots, x_T$ best fits to the trained models (see Figure 1). Opposed to the recognition of dynamic (but isolated) gestures, we maximize the posteriori probability $\Pr(w_1^N|x_1^T)$ over all possible word sequences w_1^N with unknown number of words N. This is modeled by Bayes' decision rule:

$$x_1^T \longrightarrow r(x_1^T) = \underset{w_1^N}{\operatorname{argmax}} \left\{\Pr(w_1^N) \cdot \Pr(x_1^T|w_1^N)\right\} \quad (1)$$

where $\Pr(w_1^N)$ is the a-priori probability for the word sequence w_1^N given by the language model (LM). Here, we use a smoothed trigram LM [12]. $\Pr(x_1^T|w_1^N)$ is the probability of observing features x_1^T given the word sequence w_1^N, referred to as visual model (VM).

The probability $\Pr(x_1^T|w_1^N)$ of observing the feature sequence x_1^T given a word sequence w_1^N is defined as the sum over all possible hidden Markov model (HMM) temporal state sequences $s_1^T := s_1, \ldots, s_T$ for this word sequence:

$$\Pr(x_1^T|w_1^N) = \sum_{[s_1^T]} \Pr(x_1^T, s_1^T|w_1^N). \quad (2)$$

Assuming a first order Markov dependency, Eq. 2 can be simplified as:

$$\Pr(x_1^T, s_1^T|w_1^N) = \prod_{t=1}^{T} p(x_t|s_t, w_1^N) \cdot p(s_t|s_{t-1}, w_1^N). \quad (3)$$

The optimal word sequence is found using maximum approximation over all possible state sequences:

$$\hat{w}_1^N = \underset{w_1^N}{\operatorname{argmax}} \left\{ p(w_1^N) \max_{s_1^T} \prod_{t=1}^{T} \left\{ p(f(X_t, u_t)|s_t, w_1^N) \cdot p(s_t|s_{t-1}, w_1^N) \right\} \right\} \quad (4)$$

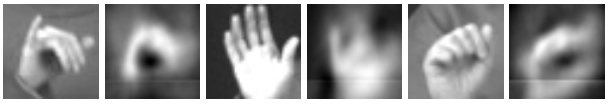


Figure 2: Examples of different hand patches extracted from tracking framework with their corresponding back-projections from PCA space using a using a 1600×30 dimensional PCA matrix

It is well-known that for natural languages the segmentation of a sentence into individual words is a non-trivial task and thus sentences are recognized jointly without segmentation into words [9]. Here, we follow this approach.

Visual Modeling. The ASLR framework and the features used to achieve the experimental results are similar to those presented in [4]. Each phoneme is modeled by a 3-state left-to-right HMM with three separate Gaussian mixtures and a globally pooled covariance matrix as emission models. The baseline system is Viterbi trained and uses a trigram LM. We use appearance-based image and hand features, i.e. thumbnails of video sequence frames, which can be reduced by linear feature reduction methods like PCA or LDA (see Figure 2). These features give a global description of all (manual and non-manual) features that have been shown to be linguistically important.

To analyze the impact of the proposed rescoring and adaptation methods within the emission probabilities $p(x_t|s_t, w_1^N)$ in Eq. 3, we focus in the following sections only on these low-level frame and hand based features instead of possibly high-level features presented e.g. in [13]. Nevertheless, the achieved results in Section 5 even outperform other approaches on the same benchmark set.

2.1. Hand Tracking

To extract manual features, the dominant hand (i.e. the hand that is mostly used for one-handed signs such as finger spelling) is tracked in each image sequence. Therefore, a robust tracking algorithm for hand tracking is required. Instead of tracking by detection, it is also possible to optimize the tracking decision considering the full sequence using dynamic programming (DP) [3]. This has the advantage to reduce tracking errors and the structure of the algorithm allows for fully integrating this into the recognition process (c.f. next section).

The DP tracking consists of two steps. In the first step the recursion of the dynamic programming tracking is executed

to obtain scores D and backpointers B:

$$D(t,x,y) = \max_{x',y' \in M(x,y)} \{(D(t-1,x',y') - \mathcal{J}(x',y',x,y)\} + d(x',y',x,y,X_{t-1}^t) \quad (5)$$

$$B(t,x,y) = \operatorname*{argmax}_{x',y' \in M(x,y)} \{(D(t-1,x',y') - \mathcal{J}(x',y',x,y)\}$$

where $M(x,y)$ is the set of possible predecessors of point (x,y) and $\mathcal{J}(x',y',x,y)$ is a jump-penalty from point (x',y') in the predecessor image to point (x,y) in the current image (e.g., the Euclidean distance). The local score function $d(x',y',x,y,X_{t-1}^t)$ measures the movement of the object to be tracked in the current frame.

In the second step, the traceback process reconstructs the best path $t \rightarrow u_t = (x,y)$ using the score table D and the backpointer table B starting from time step T.

$$u_{t-1} = B(t,u_t) \text{ with } u_T = \operatorname*{argmax}_{(x,y)} \{D(T,x,y)\} \quad (6)$$

with $D(t,x,y)$ the total score for the best path of hand positions until time step t which ends in position (x,y). Using this full traceback, the decision for a single frame automatically depends on all preceding and succeeding decisions.

Using early tracebacks over Δ frames (e.g. $\Delta = 25$), the decisions for each frame only depend on the frames which are considered in the same partial optimization. Opposed to the work of [3], here we propose to use multiple distorted tracebacks (c.f. Section 4) which are optimized later w.r.t. a hypothesized word sequence.

2.2. Integrated Tracking and Recognition

As described above, conventionally, first the tracking is performed leading to a sequence of hand positions and then features extracted from these positions are used to do the recognition. Ideally, the tracking path is chosen according to hypothesized word sequences in the recognition phase which would postpone the tracking decisions to the end of the recognition phase and lead to tracking decisions optimal w.r.t. the hypothesized word sequences.

To integrate the tracking into the recognition process (i.e., the simultaneous optimization of a tracking path u_1^T w.r.t. a tracking criterion *and* a hypothesized word sequence w_1^N), image locations u_1^T and states s_1^T can be modelled as hidden variables, leading to the following formulation for the emission probabilities:

$$\begin{aligned}\Pr(x_1^T|w_1^N) &= \sum_{[s_1^T]} \sum_{[u_1^T]} \Pr(X_1^T, s_1^T, u_1^T|w_1^N) \\ &\propto \max_{[s_1^T]} \max_{[u_1^T]} \prod_{t=1}^{T} \Big[\underbrace{\Pr(X_t|s_t,u_t,w_1^N)}_{\textbf{emission prob}} \\ &\cdot \underbrace{\Pr(s_t|s_{t-1})}_{\textbf{state transition prob}} \cdot \underbrace{\Pr(u_t|u_{t-1},X_{t-1}^t)}_{\textbf{location transition prob}} \Big] \quad (7)\end{aligned}$$

A problem with this integrated approach are resulting time complexities. Let L be the number of active locations in an image during the tracking, W the size of the vocabulary, T the length of the sequence in frames, and S the number of active states in the recognition HMM. Then, the complexity for a normal tracking is $O(TL^2)$ because for each time frame each position and each position in the predecessor frame has to be hypothesized (i.e., each transition). The time complexity for the normal search (using unigram LM) is $O(TWS)$, since for each time frame each word is hypothesized in each state. Using bigram or trigram LM, the complexity for the search is $O(T[WS+W])$ and $O(TW[WS+W])$, respectively. Thus, the conventional two-step tracking/recognition procedure has a complexity of $O(TWS + TL^2)$ which is feasible.

In the combined approach, the complexity becomes $O(TWSL^2)$ for the unigram and $O(TL^2[WS+W])$ and $O(TL^2W[WS+W])$ for the bi- and trigram search respectively, which is unfeasible for reasonably sized images.

Furthermore, obviously not each pixel in an image (let e.g. L be 320·240=76,800) is a good candidate for a tracking center. In preliminary experiments we observed that, even with a strongly pruned search space, either the runtime is too high or the image search space has to be reduced too strongly for accurate hand position tracking.

To ease the computational problems but stick to the proposed integrated tracking and recognition approach we present several approximations to this procedure in the following sections.

3. Virtual Training Samples

Due to the lack of data in video benchmark databases for sign language recognition, some visual models contain only a few observations per density. Even "one-shot" training is necessary for singletons (c.f. Section 5). This results in too sharp means which do not generalize well on unseen data.

However, for other pattern recognition problems it has been reported that the usage of additional virtual training samples (VTS) can significantly improve the system performance [2]. Here, as only a region-of-interest (ROI) is cropped from the original video frames, the amount of training data can be increased by VTS, i.e. ROIs extracted at slightly shifted positions from the original ROI position. The ROI cropping center (x,y), is shifted by δ pixels in x- and y-direction. For $\delta = \pm 1$, the training corpus is already enlarged by a factor of nine.

The proposed virtual training samples generation can be interpreted as distortion and adaptation on the signal level. Each additional virtual training sample may lead to a slightly different tracking path and thus effectively different tracking paths are considered in training and testing.

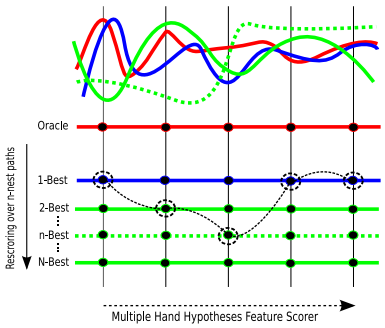


Figure 3: Rescoring by n-best list rescoring or multiple hand hypothesis are supposed to recover from tracking errors.

4. Rescoring and Path Adaptation

In Eq. 3, the emission probability $p(x_t|s_t, w_1^N) = p(f(X_t, u_t)|s_t, w_1^N)$ depends on the quality of the hand tracking position u_t and the extracted feature x_t. The system was trained with the path optimal w.r.t. the tracking criterion in order to learn word dependent hand appearance models. However in the recognition, we propose to rescore over multiple hand hypotheses in order to adapt the given path to a path being optimal w.r.t. the hypothesized word sequence and their corresponding hand models.

Figure 3 shows a tracking path scheme over time and space. Typically the path optimal w.r.t. the tracking scoring criterion (blue lines) and the resulting n-best paths (green lines) usually differ from the ground truth oracle path (red line). It can happen that correct locations occur in globally non-optimal tracking paths, whereas the globally optimal tracking path might have even worse positions for these time stamps.

***n*-Best Tracking List Rescoring.** An n-best tracking list can be generated by tracing back multiple times over the sorted score table D and the backpointer table B. Eq. 6 changes for $i = 1, .., n$ as follows:

$$u_{t-1,i} = B(t, u_{t,i}) \text{ with } u_{T,i} = \underset{\substack{(x,y)\notin \\ \{u_{T,1},\dots,u_{T,i-1}\}}}{\operatorname{argmax}} D(T, x, y)$$

The tracking list which best describes the hypothesized word sequence will be chosen. In speech recognition, this process is known as "*acoustic rescoring*" [9]. The visual model probability in Eq. 3 changes as follows: $\Pr(x_1^T, s_1^T|w_1^N) =$

$$\prod_{t=1}^{T} \left\{ \max_{\substack{i:u_1^T:= \\ (u_{1,i},\dots,u_{T,i})}} \left\{ p(f(X_t, u_t)|s_t, w_1^N) \right\} \cdot p(s_t|s_{t-1}, w_1^N) \right\}$$

Tracking list rescoring is schematically shown in Figure 3 (from top-to-bottom). Figure 4 shows an example where we visualized an n-best tracking list with 450 hand hypotheses. By incorporating different tracking hypotheses in the recognition, we allow to recover from tracking errors even for objects where tracking failed miserably and thus e.g. confusions of the hands can be resolved. This technique allows the recognition to choose among a set of tracking path candidates.

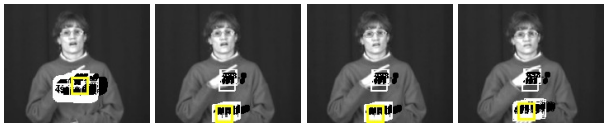


Figure 4: n-best tracking list with 450 hand hypotheses in each frame at different time stamps of a video sequence: the target object, i.e. the right and dominant-hand, is always among the active hypotheses tracking set. However, due to an abrupt movement of the dominant hand, the best path w.r.t. the tracking criterion (yellow rectangles) would track the entering non-dominant hand.

Multiple Hand Hypotheses Rescoring. Instead of rescoring with a set of n complete tracking paths, it is also possible to select multiple hand hypotheses among these: during recognition at each time step t, a set of n possible hand locations $\{u_{t,1}, .., u_{t,n}\}$ is considered and selected depending on the hypothesized word sequence. The visual model probability in Eq. 3 changes as follows: $\Pr(x_1^T, s_1^T|w_1^N) =$

$$\prod_{t=1}^{T} \left\{ \max_{i=1,..,n} \left\{ p(f(X_t, u_{t,i})|s_t, w_1^N) \right\} \cdot p(s_t|s_{t-1}, w_1^N) \right\}$$

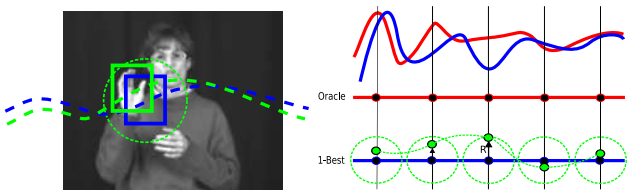
Multiple hand hypotheses (MHH) rescoring is schematically shown in Figure 3 (from left-to-right). Opposed to n-best tracking list rescoring, at each time step t, not a full path but only a tracking position is selected. Compared to the previous method, this effectively weakens the tracking constraints and allows for a higher flexibility in choosing alternative tracking position candidates in the recognition phase at the expense of a possibly loss of path smoothness.

Path Distortion Model. Another possibility to obtain a tracking path being adapted to the hypothesized word sequence is to locally distort within a range R a given tracking path.

Figure 5 (a) shows an example where the hand tracking failed: a small local tracking distortion (see Figure 5 (b)) can recover from tracking errors which results in better hand hypotheses matching to the hypothesized visual models (i.e., better emission scores).

Furthermore, it is possible to penalize locations far away from the original tracking path. Each distortion depends on the currently hypothesized word (i.e. the trained hand models), which changes the visual model probability in Eq. 3 as follows: $\Pr(x_1^T, s_1^T|w_1^N) =$

$$\prod_{t=1}^{T} \left\{ \max_{\substack{\delta\in\{(x,y):\\ -R\le x,y\le R\}}} \left\{ p(\delta) \cdot p(f(X_t, u_t+\delta)|s_t, w_1^N) \right\} \cdot p(s_t|s_{t-1}, w_1^N) \right\} \text{ with } p(\delta) = \frac{\exp(-\delta^2)}{\exp(\sum_{\delta'} -\delta'^2)}.$$



(a) (b)

Figure 5: Rescoring by distortion: The distorted hand hypotheses can be weighted by the distance to the optimal path (a). The path optimal w.r.t. the tracking scoring functions (blue line) can be distorted locally in the feature scorer (b).

Another possibility is to penalize w.r.t. trained hand positions (μ_x, μ_y) which can also be used for the proposed MHH rescoring method.

The path distortion model prunes the search space starting from a path being optimal to a tracking criterion in order to obtain a distorted path according to the hypothesized word sequence. Compared to the previous two methods, here not several tracking hypotheses are considered but we assume that the tracking may be inaccurate up to δ pixels and allow for compensating tracking errors up to this range in the recognition phase.

5. Experimental Results

For our experiments, we use a publicly available database of 201 American Sign Language sentences performed by 3 different signers, 161 are used for training and 40 for testing [4]. On the average, these sentences consist of 5 words out of a vocabulary of 104 unique words. 26% of the vocabulary words seen in training are singletons (i.e. words which occur only once in the training corpus).

We use only unseen data from the test sentences for evaluation. As we are dealing with continuous sign language sentences (instead of isolated gestures only), the recognition experiments are evaluated using the word error rate (WER) in the same way as it is done in speech recognition. The WER represents the minimum number of deletion (DEL), insertion (INS), and substitution (SUB) errors divided by the total number of signs in the recognized sentence.

In order to analyze the proposed tracking rescoring and adaptation methods, here we focus only on the usage of appearance-based hand features in contrast to full appearance-based frame features, more complex tracking features, and their combinations as proposed by the authors of [4]. The baseline results for our proposed system is shown in Table 1.

Rescoring results for n-best path rescoring over 350 paths (i.e. from best to 350^{th}) are presented in Table 2. For short tracking delays, which is good for near real-time tracking, n-best tracking list rescoring consistently improves the WER over the reference system, which uses only

Table 1: Baseline results using appearance-based features

Features	DEL	INS	SUB	errors	WER %
Frame (32x32)	43	6	16	65	35.62
PCA-Frame (200)	40	9	18	27	30.34
Hand (32x32)	31	7	43	81	45.51
PCA-Hand (70)	40	10	21	49	44.94

Table 2: Rescoring results for n-best tracking lists and multiple hand hypotheses (MHH) with different traceback delays Δ and predecessor search ranges M.

Delay Δ	WER[%]					
	$M = \pm 1$			$M = \pm 10$		
	1-best	n-best	MHH	1-best	n-best	MHH
Full	80.34	76.97	76.40	**45.51**	**45.51**	**45.51**
100	79.78	75.28	73.03	45.51	45.51	45.51
25	70.79	64.61	66.29	56.18	50.56	53.37
10	69.10	67.98	65.17	63.48	60.11	58.99
1	91.01	83.71	65.17	91.01	83.71	65.17

the best path. However, a full traceback with sufficiently large search regions M outperforms short delays. No further improvements are achieved for $\Delta > 100$ which corresponds to the average length of the sentences. The best result is obtained for a predecessor search range of $M = \pm 10$ pixels in Eq. 5. This is the best setting for all delays Δ.

Rescoring results for multiple hand hypotheses (MHH) on an n-best tracking list for $n = 150$ are presented in Table 2. We observed that many paths recombine to one path for a long or full traceback delay. Therefore short tracking delays can be used to generate a larger path diversity. It can be seen that multiple hand hypotheses outperform the standard approach and the n-best tracking list rescoring results from Table 2 up to a short tracking delay of $\Delta = 10$ frames (Section 2.1). However, the chosen tracking scoring functions and delays led on the one hand to a large path diversity but on the other hand to many wrong hand hypotheses (e.g. the moving elbows), so that the optimal path w.r.t. the tracking criterion is also the best w.r.t. the WER.

Table 3 shows some rescoring results obtained with a tracking path distortion model and different distortion ranges and penalties. We used the squared Euclidean point distance as distortion penalty. It can be seen that too large distortions increase the WER, and that an additional distortion penalty reduces the WER again for larger distortion. A distortion range of $R = 10$ pixel with additional δ-penalty is sufficient, larger values led to no further improvements.

The rescoring results using the path distortion model in combination with virtual training samples are shown in Table 4. The usage of additional training data by virtual training samples (VTS) leads to improvements in all experiments. The WER of 11.29% is the best result reported for this data in the literature so far (17.98% in [4]).

Table 3: Rescoring results for a tracking path distortion model with different distortion ranges R and δ-penalties.

Feature	WER[%]			
	$R=0$	$R=3$	$R=5$	$R=10$
Hand (32×32)	45.51	41.51	34.27	41.03
+ δ-penalty	—	38.02	34.27	**35.96**
PCA-Hand (70)	44.94	32.58	34.83	56.74
+ δ-penalty	—	33.25	30.90	**32.58**

Table 4: Rescoring results using the path distortion model and virtual training samples.

Features / Rescoring	WER[%]			
	pixel values		PCA transformed	
	Baseline	VTS	Baseline	VTS
Frame 32×32	35.62	**27.53**	30.34	**19.10**
Hand (32×32)	45.51	20.79	44.94	15.73
+ distortion ($R=10$)	41.03	16.29	56.74	12.92
+ δ-penalty	35.96	**15.73**	32.58	**11.24**

6. Conclusions

We presented several tracking rescoring and adaptation methods to obtain an adapted hand tracking path with optimized tracking positions w.r.t. recognition instead of a tracking criterion.

Different tracking rescoring methods showed large improvements. On the one hand, the proposed n-best path rescoring and multiple hand hypotheses require short tracking delays in order to obtain a large diversity of the possible tracking paths, and led so far to no improvements for the used tracking method. On the other hand, the proposed path distortion model yields large improvements.

More robust models were trained using virtual training samples (VTS) easing the lack of data problem in vision based sign language recognition. The VTS data improved the system performance in all cases, and the proposed method can be applied to any vision based system. In combination with a path distortion model rescoring, the baseline WER of 44.94% on the benchmark database was improved to 11.29% WER, which is the currently best known WER on the used database.

In particular, the proposed path distortion model can be applied to any tracking based features (e.g. body part models), regardless of the chosen tracking method or the features extracted from those tracking regions (e.g. color or contour based). Interesting will be an iterative recognition and re-training of the system using the model adapted tracking path, and an analysis and extension of the proposed methods for e.g. body part features [14].

References

[1] R. Bowden, D. Windridge, T. Kadir, A. Zisserman, and M. Brady. A linguistic feature vector for the visual interpretation of sign language. In *ECCV*, vol 1, pp. 390–401, 2004.

[2] C. J. C. Burges and B. Schölkopf. Improving the accuracy and speed of support vector machines. In *NIPS*, vol 9, pp. 375–385, Vancouver, Canada, dec 1997.

[3] P. Dreuw, T. Deselaers, D. Rybach, D. Keysers, and H. Ney. Tracking using dynamic programming for appearance-based sign language recognition. In *FG*, pp. 293–298, Southampton, Apr. 2006.

[4] P. Dreuw, D. Rybach, T. Deselaers, M. Zahedi, and H. Ney. Speech recognition techniques for a sign language recognition system. In *Interspeech 2007*, pp. 2513–2516, Antwerp, Belgium, Aug. 2007.

[5] A. Elgammal. Learning to track: Conceptual manifold map for closed-form tracking. In *CVPR*, San Diego, CA, USA, June 2005.

[6] A. Farhadi, D. Forsyth, and R. White. Transfer learning in sign language. In *CVPR*, 2007.

[7] H. Guan, R. S. Feris, and M. Turk. The isometric self-organizing map for 3d hand pose estimation. In *FG*, Southampton, UK, Apr. 2006.

[8] M. Isard and A. Blake. CONDENSATION – conditional density propagation for visual tracking. *IJCV*, 29(1):5–28, August 1998.

[9] F. Jelinek. *Statistical Methods for Speech Recognition*. MIT Press, Cambridge, Massachusetts, January 1998.

[10] A. S. Junwei Han, George M. Awad and H. Wu. Automatic skin segmentation for gesture recognition combining region and support vector machine active learning. In *FG*, Southampton, UK, Apr. 2006.

[11] A. Just, Y. Rodriguez, and S. Marcel. Hand posture classification and recognition using the modified census transform. In *FG*, Southampton, UK, Apr. 2006.

[12] R. Kneser and H. Ney. Improved backing-off for m-gram language modeling. In *ICASSP*, vol 1, pp. 49–52, Detroit, MI, 1995.

[13] S. Ong and S. Ranganath. Automatic sign language analysis: A survey and the future beyond lexical meaning. *PAMI*, 27(6):873–891, June 2005.

[14] D. Ramanan, D. A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *PAMI*, 29(1):65–81, Sept. 2007.

[15] C. Vogler and D. Metaxas. A framework for recognizing the simultaneous aspects of american sign language. *CVIU*, 81(3):358–384, Mar. 2001.

[16] C. Wang, X. Chen, and W. Gao. Re-sampling for chinese sign language recognition. In *Gesture in Human-Computer Interaction and Simulation*, vol 3881 of *LNCS*, pp. 57–67, Feb. 2006.

[17] H. Wang, D. Suter, K. Schindler, and C. Shen. Adaptive object tracking based on an effective appearance filter. *PAMI*, 29(9):1661–1667, Sept. 2007.

[18] R. Yang, S. Sarkar, and B. Loeding. Enhanced level building algorithm for the movement epenthesis problem in sign language recognition. In *CVPR*, 2007.