

NON-INTRUSIVE SIGN LANGUAGE RECOGNITION FOR HUMAN-COMPUTER INTERACTION

Jörg Zieren ^{*,1} Karl-Friedrich Kraiss ^{*}

** Chair of Technical Computer Science,
RWTH Aachen University, Germany*

Abstract: We present a system for non-intrusive sign language recognition from a monocular frontal view. Geared towards background independence, sophisticated localization and tracking methods, such as a combined EM/CAMSHIFT overlap resolution procedure and the parallel pursuit of multiple hypotheses regarding hand position and movement, are applied. High-level knowledge is incorporated through a biomechanical skeleton model and dynamic Kalman filter predictions. Using an HMM classifier, a person dependent recognition rate of 97.6% is achieved on a vocabulary of 152 signs from German sign language.

Keywords: Sign Language Recognition, EM, CAMSHIFT, Multiple Hypotheses

1. INTRODUCTION

The use of gestures as a means to convey information is an important part of human communication. The automatic recognition of gestures enriches human-computer interaction by offering a natural and intuitive method of data input (Pavlovic *et al.*, 1997; Hienz *et al.*, 1999; Akyol *et al.*, 2000b).

Sign language recognition constitutes a special field in this research area (Akyol *et al.*, 2000a; Akyol and Canzler, 2002). In sign language, information is communicated primarily through hands and face. The application of gesture recognition methods in this context is challenging, since – in contrast to common gesture control systems – the vocabulary cannot be shaped in a way that optimizes the system’s performance by avoiding common computer vision problems such as occlusion, overlap, or minimal pairs.

This work focuses on the manual parameters of sign language and proves them to be suitable for the recognition of 152 signs from German sign language. Our system employs a single video camera positioned in front of the signer. It facilitates non-intrusive interaction, i. e. it does not require the use of markers and poses very low restrictions on the user’s clothing.

This task requires methods to accurately localize and track the signer’s hands even if they overlap or move in front of the face, which happens frequently in sign language. We use a combined EM/CAMSHIFT procedure to compute manual features in this case. In order to cope with the inherently ambiguous results of a skin color based threshold segmentation, multiple hypotheses are pursued in parallel. The evaluation of these hypotheses exploits high-level knowledge such as a biomechanical human skeleton model and predictions computed by Kalman filters. A winner hypothesis is chosen only after all available information (i. e. all frames) has been processed.

¹ This work has been performed within the European Commission funded project WISDOM (Wireless Information Services for Deaf people On the Move), IST-2000-27512

The extracted features are classified using Hidden Markov Models (HMMs). On our data set the system achieves a person dependent recognition rate of 97.6%.

2. EXPERIMENTAL SETUP

The chosen experimental setup contains the following assumptions regarding interaction, video recording, and user (see Figure 1):

- The view is chosen so that the signer’s upper body is captured.
- The face is always in the image, although it may be completely occluded by the hands.
- The signer is standing at the center of the image, facing the camera. No other objects are visible.
- Signing starts and ends with the arms hanging down in a relaxed position. In this idle position, the hands do not need to be visible in the image.
- A tripod mounted camera and strong diffuse lighting provide stable recording conditions.
- The user wears long sleeved non-skin colored clothing.

Complying with these requirements 152 signs have been performed by one person ten times each. Recorded at a resolution of 384×288 pixel and 25 frames per second, this constitutes a data base of about 70.000 individual images. A single sign takes about two seconds.

3. TRACKING

The object tracking approach can be split hierarchically:

- (1) A low level processing stage detects areas of interest using skin color as an image cue.
- (2) A high level processing stage identifies these areas of interest as targets or distractors by evaluating multiple hypotheses per frame and over time.

Section 3.1 discusses the low level stage, focussing on overlap detection and resolution. Section 3.2 explains the multiple hypotheses method used in the high level stage to interpret the previous step’s output and solve the actual tracking task. The computation of features is described in section 3.3.

3.1 Low Level Processing Stage

Initially, skin colored areas are detected using generic skin and non-skin color histograms (Jones and Rehg, 1998). A threshold segmentation yields several blobs that, in the presence of distractors,



Fig. 1. Example sign “computer”.

form a superset of the target objects (face and hands).

Rather than processing a blob’s boundary, an elliptical approximation called “blob ellipse” is used. A blob ellipse is characterized by five scalar values:

- Center coordinates (x, y)
- Standard deviations of the underlying skin color distribution in direction of the principal and secondary axis (σ_1, σ_2)
- Orientation of the principal axis (α) .

The principal drawback of this segmentation algorithm is that overlapping skin colored objects form a single blob (Zieren *et al.*, 2002). To extract meaningful features, however, a separation of the overlapping objects is required. This is described in the following section.

3.1.1. Overlap Detection

First, a distinction is introduced between the set of “raw” blob ellipses extracted by the threshold segmentation in frame t , called $\mathcal{B}_{\text{raw},t}$, and a corresponding set of “overlap resolved” blob ellipses \mathcal{B}_t that is actually forwarded to the high level stage. The overlap detection and resolution method is based on the computation of an assignment $\mathcal{B}_{\text{raw},t} \mapsto \mathcal{B}_{t-1}$. \mathcal{B}_t results from this assignment. Four cases can be identified:

- (1) A blob ellipse $\mathbf{b}_{\text{raw},t} \in \mathcal{B}_{\text{raw},t}$ is assigned exactly one blob ellipse $\mathbf{b}_{t-1} \in \mathcal{B}_{t-1}$. This 1:1 assignment does not require further processing.
- (2) A blob ellipse $\mathbf{b}_{\text{raw},t} \in \mathcal{B}_{\text{raw},t}$ is assigned two or more blob ellipses $\mathbf{b}_{t-1,1}, \mathbf{b}_{t-1,2}, \dots, \mathbf{b}_{t-1,n} \in \mathcal{B}_{t-1}$. This 1:n assignment, usually caused by an overlap, requires further processing as described below.
- (3) A blob ellipse $\mathbf{b}_{\text{raw}} \in \mathcal{B}_{\text{raw}}$ is not assigned to any blob ellipse $\mathbf{b} \in \mathcal{B}_{t-1}$. This case occurs if a new object enters the image.
- (4) A blob ellipse $\mathbf{b} \in \mathcal{B}_{t-1}$ is not assigned to any blob ellipse $\mathbf{b}_{\text{raw}} \in \mathcal{B}_{\text{raw}}$, meaning that an object has either been completely occluded or left the image.

Three different assignment methods are now applied in order of decreasing accuracy:

Assignment of static blob ellipses. Using a Bayesian Belief Network (BBN) to evaluate position and shape differences between $\mathbf{b}_{\text{raw},t}$ and \mathbf{b}_{t-1} , static objects can be identified reliably.

Assignment by degree of overlap. Another BBN is used to evaluate the overlapping area for $\mathbf{b}_{\text{raw},t}$ and \mathbf{b}_{t-1} .

Assignment by prediction. Shape and position of each blob ellipse are estimated from past observations using a Kalman filter. The filter’s prediction is compared to $\mathbf{b}_{\text{raw},t}$ and evaluated by a BBN. This step allows to track even discontinuous hand movements. It is described in more detail in Section 3.1.2.

These steps solve the assignment problem for the 1:1 and 1:n case. The remaining cases, as well as the treatment of overlap, is described below.

Identification of new blob ellipses. All blob ellipses $\mathbf{b}_{\text{raw},t}$ left unassigned until now are inserted into the set \mathbf{B}_t .

Identification of invalid blob ellipses. Blob ellipses $\mathbf{b}_{t-1} \in \mathbf{B}_{t-1}$ not assigned to any blob ellipse $\mathbf{b}_{\text{raw},t}$ are removed, provided that they were at or approaching the border of the image.

Overlap resolution using the Expectation Maximization (EM) algorithm. Each blob ellipse $\mathbf{b}_{\text{raw},t}$ assigned $n > 1$ blob ellipses $\mathbf{b}_{t-1} \in \mathbf{B}_{t-1}$ indicates (the beginning of) an overlap. The application of the EM algorithm (see Section 3.1.3) resolves this overlap, yielding n overlapping blob ellipses.

Overlap resolution using the CAMSHIFT algorithm. In case of a complete overlap (e.g. one blob ellipse placed completely within another) the EM algorithm can no longer provide accurate results. The system then falls back on the CAMSHIFT algorithm (see Section 3.1.4).

The algorithms introduced above are now explained in more detail.

3.1.2. Assignment by Prediction

In every frame, an array of Kalman filters compute predictions for all five components of each blob ellipse. Different motion models are employed, as shown in Table 1.

Table 1. Kalman filter motion models for blob ellipse parameter prediction.

Parameter	Motion Model
center x coordinate	constant acceleration
center y coordinate	constant acceleration
σ_1	constant velocity
σ_2	constant velocity
α	constant value

Each candidate’s conformance with the predicted values is quantified by a BBN based on a Mahalanobis distance computed from the Kalman predictions for σ_1 , σ_2 , and α . From the set of all possible assignments, the one with the highest probability is selected.

3.1.3. Overlap Resolution Using the EM Algorithm

The Expectation Maximization (EM) algorithm (Bilmes, 1998) is suitable for resolution of partial overlap (Akyol, 2003) as shown in Figure 2a. Beforehand, a morphological distance transformation (Jain, 1989) is applied to the threshold segmented binary skin color mask to obtain a pseudo-multivariate distribution. Figure 2 visualizes this procedure.

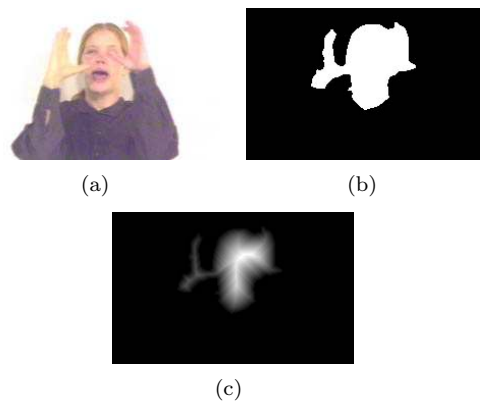


Fig. 2. Preparation of the skin color mask for the EM algorithm. (a) Original image, (b) Binary skin color mask, (c) Distance transformed skin color mask.

The approximation status at different iterations is shown in Figure 3. As a modification to the original algorithm, constraints are placed upon certain shape parameters to allow adaptation only in a limited range or not at all.

3.1.4. Overlap Resolution Using the CAMSHIFT Algorithm

In order to localize objects that completely overlap with other objects (e.g. a hand right in front of the face), the CAMSHIFT algorithm (Bradski, 1998) is applied to an image $I_c(x, y)$ which is obtained from a Motion History Image $I_m(x, y)$ (Davis and Bobick, 1997) and the skin color probability distribution $p_{\text{skin}}(x, y)$ according to the following equations.

$$I_s(x, y) = p_{\text{skin}}(x, y) \quad (1)$$

$$I_{s,m}(x, y) = k_1 I_s(x, y) + k_2 I_m(x, y) \quad (2)$$

$$I_c(x, y) = \min(I_s(x, y), I_{s,m}(x, y)) \quad (3)$$

$I_s(x, y)$ and $I_{s,m}(x, y)$ denote the skin color and combined skin color/motion image. The weights

k_1 and k_2 have been empirically optimized to 0.3 and 0.7, respectively. The minimum operation in equation 3 prevents the CAMSHIFT search window from being distracted by moving but non-skin colored objects.

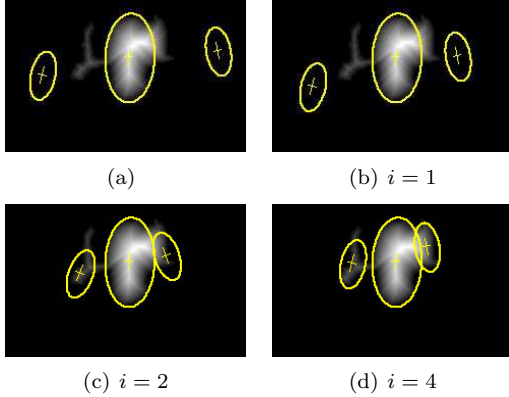


Fig. 3. Application of the EM algorithm. (a) Initialization, (b – d) after i iterations.

3.2 High Level Processing Stage

In every frame, the output of the low level stage allows a number of hypotheses as to which of the extracted blob ellipses represent the signer’s face, left hand, or right hand. Some of these are more likely while others are unlikely or even impossible. Also, considering two subsequent frames, it is obvious that a hypothesis’ probability is never independent of its predecessor because the underlying motion that occurred between the two frames is subject to physical laws. Solving the tracking problem means to find the correct hypothesis in as many frames as possible.

These considerations suggest to view the input image sequence $I_{t_0}, I_{t_0+1}, \dots, I_{t_1}$ as a state space with N_t states at time t , resulting in $\prod_{t=t_0}^{t_1} N_t$ different paths (i.e. tracking results). The high level stage now searches this state space for the most likely path by computing static probabilities p_{stat} for states and dynamic probabilities p_{dyn} for transitions. An example state space is shown in Figure 4, using the following notation:

- $p_{\text{stat},t}(i)$ is the static probability for state i at time t .
- $p_{\text{dyn},t}(i, j)$ is the dynamic probability for the transition from state i at time t to state j at time $t + 1$.

Since a complete search of the state space is computationally not feasible, the Viterbi search algorithm (Rabiner and Juang, 1986) is applied to reduce the number of considered paths to N_t at time t . The following sections sketch the evaluation criteria used for computation of the above probabilities.

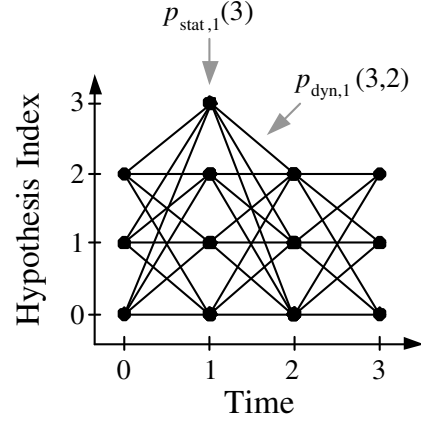


Fig. 4. Hypothesis space with static and dynamic probabilities.

3.2.1. Static Evaluation Criteria

Body Posture. Based on the face width and the hands’ positions, a body model is computed for both visualization and hypothesis evaluation. Configurations that are physiologically unlikely or do not occur in sign language receive a low probability.

Hand Position. From a manual segmentation of the test/training data (cf. Section 4.1), a position histogram has been computed for each hand. This allows to evaluate a given position (localization result) with respect to its likeliness in the test/training data set.

Motion Information. Using the Motion History Image described in Section 3.1.4, the amount of motion within the boundary of each blob ellipse can be computed. Since sign language uses motion to convey information, blob ellipses with a high amount of motion are considered more likely than others.

Idle Position at Start and End. In the available input video clips, each sign starts and ends with the hands hanging approximately beside the hips. This idle position is computed from the body model introduced above. The first and last configurations’ probabilities decrease with increasing y coordinate difference between the hypothesized and the idle position.

Preferred Hand. This criterion considers the signer’s handedness and exploits it for tracking single handed signs. Knowledge of the handedness is also a prerequisite for a correct interpretation of the extracted features.

3.2.2. Dynamic Evaluation Criteria

Kalman Filter Prediction. Based on the previous states of the current path, Kalman filters are used to compute predictions for position, shape, and orientation of the left and right hand’s blob ellipses. This process is similar to the assignment by prediction performed by the

low level stage as described in Section 3.1.2. The hypothesized configuration’s deviation from the predicted values is input into a BBN which computes the corresponding dynamic probability.

3.2.3. Conclusion of Tracking

After the input clip has been completely processed, the Viterbi path search (see Section 3.2) yields a winner path through the state space that constitutes the tracking result. The system then composes a sequence of feature vectors as described in the following section.

3.3 Computation of Features

The chosen features describe the two-dimensional projection of each hand in the image plane:

- Center coordinates
- Area
- Orientation
- Ratio of principal to secondary axis
- Compactness (Sonka *et al.*, 1999)
- Eccentricity (Sonka *et al.*, 1999)

The first four items refer to the ellipse approximation of the hand shape, while the last two describe the shape of the actual object border resulting from the threshold segmentation. Position and area are normalized to body measures for independence from resolution and from the signer’s position within the image. For all of these features, temporal derivatives are computed, yielding 14 elements per hand and a total of 28 elements in the feature vector. After composing this feature vector for every frame, idle frames/vectors are cropped at the start and end of the sign.

4. EVALUATION

Two aspects of the system’s performance were measured on the available data set: The accuracy of the tracking result, and the actual recognition rate achieved with the HMM classifier. Both will be discussed below.

4.1 Tracking Accuracy

In order to quantitatively measure the tracker’s hit rate, all input clips (70.000 images) were manually segmented. Based on this reference a hit can be defined as (see Figure 5):

- (1) The estimated position lies within the border of the target object (“hit on object”).
- (2) The estimated position lies within a limited region around the target center, but not necessarily on the object (“hit near center”).



Fig. 5. Definition of tracker hit and miss.

Table 2 shows the experimental results. The difficulty of the tracking task depends on the degree and type of overlap and the number of hands used. The vocabulary has been grouped into five categories to demonstrate this effect. In the right column, idle positions at the start and end of a sign have been excluded from the evaluation.

Table 2. Quantitative evaluation of tracking accuracy (H=hand, F=face).
(a) Idle frames included, (b) Idle frames excluded.

Sign Category	Hit Rate (a)	Hit Rate (b)
One Handed	98.4%	96.9%
Two Handed	95.4%	93.9%
No Overlap	99.0%	98.6%
H-H Overlap	96.8%	94.3%
F-H-H Overlap	97.1%	95.1%

4.2 Recognition Performance

Recognition performance can be characterized by the percentage of correct classifications on the test/training data set (recognition rate) and the processing time required per recognition. Table 3 lists recognition rates on the vocabulary of 152 signs in German sign language, obtained in a leaving-one-out test, and average processing times per sign on a 1 GHz PC. Reducing resolution is shown to cause only minor loss in recognition rate, while the processing time can be reduced by about 50%. The exact processing time depends on the amount of overlap and the number of distractors present.

Table 3. Recognition rate and average processing time as functions of the input image resolution.

Resolution	Recognition Rate	Avg. Proc. Time
384 × 288	97,6%	14.2 sec.
192 × 144	97,2%	7.4 sec.
128 × 96	97,1%	7.2 sec.

Compared with existing intrusive and non-intrusive systems such as (Gobel, 1999), (Hienz, 2000), (Starner *et al.*, 1998), or (Yang *et al.*, 2002), an increase in performance has been achieved. Higher recognition rates have only been published for significantly smaller vocabularies (around 40 signs). Furthermore, many approaches do not explicitly consider distractors, or they use a complex setup with multiple cameras and/or markers. In contrast, the concept presented here is suitable for application in “real life” or mobile scenarios.

5. ACKNOWLEDGEMENTS

This work was carried out at the Chair of Technical Computer Science, RWTH Aachen University, based on a dissertation by Suat Akyol (Akyol, 2003) and a diploma thesis by Nils Unger. Further contributions include dissertations by Britta Bauer (Bauer, 2003), Kirsti Grobel (Grobel, 1999), and Hermann Hienz (Hienz, 2000). Numerous other researchers and students have contributed code to the project (*LTI-Lib: A C++ library for image processing and computer vision*, 2003).

REFERENCES

- Akyol, S. (2003). *Nicht-intrusive Erkennung isolierter Gesten und Gebärden (Non-Intrusive Recognition of Isolated Gestures and Signs)*. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.
- Akyol, S. and U. Canzler (2002). An information terminal using vision based sign language recognition. In: *ITEA Workshop on Virtual Home Environments, C-LAB Publication* (U. Büker, H.-J. Eikerling and W. Müller, Eds.). Vol. 12. Paderborn, Germany. pp. 61–68.
- Akyol, S., U. Canzler and B. Bauer (2000a). Gesture and mimic interpretation for sign language recognition. In: *Proceedings of the 4th International Student Conference on Electrical Engineering (POSTER 2000)*. Prague, Czech Republic. p. IC1.
- Akyol, S., U. Canzler, K. Bengler and W. Hahn (2000b). Gesture control for use in automobiles. In: *Proceedings of the IAPR MVA 2000 Workshop on Machine Vision Applications*. Tokyo, Japan. pp. 349–352.
- Bauer, B. (2003). *Erkennung kontinuierlicher Gebärdensprache mit Untereinheiten-Modellen (Recognition of Continuous Sign Language Using Subunit Models)*. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.
- Bilmes, J. A. (1998). A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report TR-97-021. International Computer Science Institute. U.C. Berkeley.
- Bradski, G. R. (1998). Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*.
- Davis, J.-W. and A.-F. Bobick (1997). The Representation and Recognition of Action Using Temporal Templates. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*. San Juan, Puerto Rico. pp. 928–934.
- Grobel, K. (1999). *Videobasierte Gebärdenspracherkennung mit Hidden-Markov-Modellen (Video-Based Sign Language Recognition Using Hidden Markov Models)*. Fortschritts-Berichte VDI 10/592. VDI Verlag. Düsseldorf. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.
- Hienz, H. (2000). *Erkennung kontinuierlicher Gebärdensprache mit Ganzwortmodellen (Recognition of Continuous Sign Language Using Whole Word Models)*. Shaker Verlag. Aachen. Dissertation, Chair of Technical Computer Science, RWTH Aachen University.
- Hienz, H., J. Marrenbach, R. Steffan and S. Akyol (1999). Multimodal human-computer communication in technical applications. In: *Proceedings of the International Conference on Human Computer Interaction (HCI)* (H. J. Bullinger and J. Ziegler, Eds.). Munich, Germany. pp. I/755–I/759.
- Jain, A. K. (1989). *Fundamentals of Digital Image Processing*. Prentice-Hall Inc.. Englewood Cliffs, NJ.
- Jones, M. J. and J. M. Rehg (1998). Statistical Color Models with Application to Skin Detection. Technical Report CRL 98/11. Compaq Cambridge Research Lab.
- LTI-Lib: A C++ library for image processing and computer vision* (2003). <http://ltilib.sf.net>.
- Pavlovic, V. I., R. Sharma and T. Huang (1997). Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**(7), 677–695.
- Rabiner, L. and B.-H. Juang (1986). An Introduction to Hidden Markov Models. *IEEE ASSP Magazine* **3**(1), 4–16.
- Sonka, M., V. Hlavac and R. Boyle (1999). *Image Processing, Analysis, and Machine Vision*. Brooks/Cole Publishing Company.
- Starner, T., J. Weaver and A. Pentland (1998). Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**(12), 1371–1375.
- Yang, M.-H., N. Ahuja and M. Tabb (2002). Extraction of 2D Motion Trajectories and its Application to Hand Gesture Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(8), 1061–1074.
- Zieren, J., N. Unger and S. Akyol (2002). Hands Tracking from Frontal View for Vision-Based Gesture Recognition. In: *Lecture Notes in Computer Science LNCS 2449* (L. van Gool, J. Hartmanis and J. van Leeuwen, Eds.). Zürich, Switzerland.