COMPARISON OF APPROACHES TO CONTINUOUS HAND GESTURE RECOGNITION FOR A VISUAL DIALOG SYSTEM

Peter Morguet and Manfred Lang

Institute for Human-Machine-Communication
Munich University of Technology
Arcisstr. 21, D-80290 Munich, Germany
{mor, lg}@mmk.e-technik.tu-muenchen.de

ABSTRACT

Continuous hand gesture recognition requires the detection of gestures in a video stream and their classification. In this paper two continuous recognition solutions using Hidden-Markov-Models (HMMs) are compared. The first approach uses a motion detection algorithm to isolate gesture candidates followed by a HMM recognition step. The second approach is a single-stage, HMM-based spotting method improved by a new implicit duration modeling. Both strategies have been tested on continuous video data containing 41 different types of gestures embedded in random motion. The data has been derived from usability experiments with an application providing a realistic visual dialog scenario. The results show that the improved spotting method in contrast to the motion detection approach can successfully suppress random motion providing excellent recognition results.

1. INTRODUCTION

Gestures are an efficient communication modality in many everyday situations. To study and optimize the possible use of gestures in human-machine interaction a visual dialog system, which is exclusively controled by gestural commands, has been developed (see fig. 1 and sec. 2). The central task of the system is the temporal segmentation and classification of image sequences.

Since the HMM-based recognition of isolated image sequences works satisfactorily (e. g. see in [1, 4, 5, 8, 9]), it suggests itself to add an independent motion detection to achieve the temporal segmentation. The advantage of this first approach is the low additional computational cost of the implemented detection which is based on the image features (see sec. 3.2). Since, of course, any kind of motion is detected, the efficiency of this approach can only be measured in a realistic dialog scenario (see secs. 2 and 4).

The second approach is an integrated, HMM-based spotting method which was introduced in [6]. Tests using synthetically connected gesture material out of a small size catalog proved the ability to ignore non-meaningful and to indicate meaningful movements [7]. For the first time, the newly improved spotting algorithm (see sec. 3.3) has been applied to true continuous video data and tested against the above two-stage solution (see sec. 6).

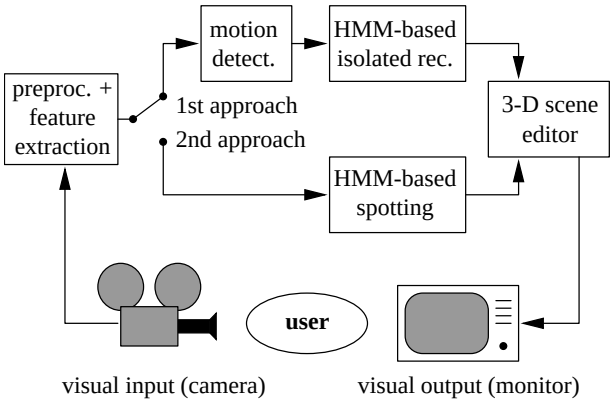

Figure 1: System overview

2. THE VISUAL DIALOG APPLICATION

The used visual dialog application is a gesture controled 3-D scene editor. The editor allows to create three-dimensional objects, to change their position and orientation, and to destroy them. Certain object attributes, like color and size, can be varied. Finally, the orientation and the observation distance of the whole scene can be changed.

The objects in the scene editor are manipulated *indirectly*: the editor contains a graphically represented agent which is supposed to be the communication partner. This agent receives gestural directives from the user, interprets them, and carries out the desired actions to a certain extent independently. These actions and the internal state of the application are represented by graphical animations and appearance changes. Complex or abstract actions, like gen-

erating and destroying objects or changing their attributes, are assigned to graphical objects in the scene. As a consequence, those actions can be consistently selected and grabbed like any other scene object.

3. DESCRIPTION OF ALGORITHMS

3.1. Spatial segmentation and feature extraction

Previous tests showed that the hand *shape* provides enough information to distinguish image sequences [6, 7]. The hand shape is obtained using a segmentation method trained on skin color. The segmented binary images are transformed into Hu moment invariants (HMIs) [2]. The HMIs $h_{i,t}$ up to order H, the differences of the HMIs of successive images $\Delta h_{i,t}$, the difference of the shape areas ΔA_t and of the centers of mass $(\Delta\bar{x_t}, \Delta\bar{y_t})$ form a feature vector $\mathbf{v}_t$ at time t:

$$\begin{aligned}\mathbf{v}_t &= [v_{1,t}, v_{2,t}, \ldots, v_{2N_H+3,t}]^{\mathrm{T}} \\ &= [\Delta A_t, \Delta\bar{x_t}, \Delta\bar{y_t}, \\ &\quad h_{1,t}, \ldots, h_{N_H,t}, \Delta h_{1,t}, \ldots, \Delta h_{N_H,t}]^{\mathrm{T}}. \end{aligned} \quad (1)$$

The mean value μ_{v_i} and standard deviation σ_{v_i} of the feature vector elements $v_{i,t}$ are used to calculate an unbiased and normalized feature vector $\mathbf{v}'_t$ with comparable elements:

$$v'_{i,t} = \frac{v_{i,t} - \mu_{v_i}}{\sigma_{v_i}}. \quad (2)$$

The feature vector $\mathbf{v}'_t$ is the basis of the motion detection, the isolated recognition and the continuous spotting process.

3.2. Approach 1: motion detection and HMM-based isolated recognition

The *differential* feature vector elements $v'_{i,t}$ reflect motion or shape changes in the image sequence. The absolute value of a feature vector built from these differential elements is defined as the *motion value* of the image sequence at time t:

$$m_t = \sqrt{\Delta A'^2_t + \Delta\bar{x}'^2_t + \Delta\bar{y}'^2_t + \Delta h'^2_{1,t} + \ldots + \Delta h'^2_{N_H,t}}. \quad (3)$$

To indicate the begin t_b of a coherent segment of motion, this motion value has to be above a certain motion threshold m_thres for the next τ_bmin subsequent images of the sequence. The end of a motion segment t_e is reached if the motion value stays below the threshold m_thres for at least the following τ_emin images. This is necessary since many gestures contain short motion pauses at turning points. A minimum total length of the motion segment τ_lmin and a minimum distance to the last detected motion τ_dmin help to find only motion segments that are possible gesture candidates.

Semi-continuous, left-to-right structured HMMs were used to classify the motion segments. The specific probability density functions $F_{s_i,t} = \log f_{s_i}(\mathbf{v}'_t)$ in the states s_i, $i = 1, \ldots, N$ of the HMMs are defined by a codebook of L mixture density functions (or *prototypes*) calculated for the whole training data. The transition probabilities $A_{s_j,s_i} = \log a_{s_j,s_i}$ describe the sequence of states.

Training and recognition are based on the Viterbi algorithm [3]. It recursively accumulates and maximizes the so called local *score* $D_{s_i,t}$ for every HMM state:

$$D_{s_i,t} = \max_j [D_{s_j,t-1} + A_{s_j,s_i}] + F_{s_i,t}. \quad (4)$$

To classify a motion segment, the score accumulation starts at time t_b and results in the final score $D^{(\lambda_i)}_{s_N,t_\mathrm{e}}$ in the last state of the respective models λ_i at the end of a detected motion. A final maximum likelihood decision provides the best matching model λ_i. As a result, a gestural meaning is assigned to *every* detected motion segment.

3.3. Approach 2: HMM-based spotting

To spot gestures in a continuous video stream, the features are fed into the HMMs at every time step. To prevent the output score $D_{s_N,t}$ from increasing or decreasing permanently, the local scores have to be normalized to its respective Viterbi path lengths [6, 7]. For that reason, the local path lengths $L_{s_i,t}$ are stored along with the local scores, and a *normalized* Viterbi algorithm is formulated:

$$\begin{aligned} D_{s_i,t} &= \max_j \left[\frac{D_{s_j,t-1} \cdot L_{s_j,t-1} + A_{s_j,s_i} + F_{s_i,t}}{L_{s_j,t-1} + 1} \right], \\ L_{s_i,t} &= L_{s_k,t-1} + 1 \quad \text{with } k = \text{index of best } s_j. \end{aligned} \quad (5)$$

Several methods to trigger new paths in the first state s_1 have been examined [7]. Optimal results are obtained if a new path with the length $L_{s_1,t} = 1$ starts permanently. The output score of the respective model will start to increase if an appropriate gesture appears; it will decrease after the gesture has ended. Consequently, peaks in the output score indicate the possible end of a gesture.

A smoothing process and several peak detection rules are necessary for a reliable peak detection requiring the following variables [7]: τ_sb and τ_se define the smoothing interval, τ_pb and τ_pe the peak detection interval. A relative rejection threshold S_rel, based on the model dependent absolute maximum and minimum output scores, and a minimum temporal peak distance t_dist help to suppress irrelevant peaks.

The local path length can be manipulated allowing a simple duration modeling. Given the estimated average duration [3] of the whole HMM as[1]:

$$\bar{d} = \sum_{i=1}^{N} \bar{d}_{s_i} \approx \frac{N}{N-1} \sum_{i=1}^{N-1} \frac{1}{1-a_{ii}}, \quad (6)$$

[1]The approximation in eq. (6) is necessary since in a left-to-right model the transition probability of the last state is always 1.

functional category	number of variations
displace	23
rotate	4
tilt	2
change size	4
stop	2
release	3
point	1
trigger action	2
	41

Table 1: Gesture catalog

a new normalization length *function* can be defined as:

$$L'_{s_i,t} = \begin{cases} L_{s_i,t} & \text{for} \quad L_{s_i,t} < \bar{d} \\ v \cdot (L_{s_i,t} - \bar{d}) + \bar{d} & \text{for} \quad L_{s_i,t} \geq \bar{d} \end{cases} . \qquad (7)$$

As a result, Viterbi paths are artificially "extended" if they are longer than the average model duration and if the normalization function parameter v is greater than 1. This indirectly helps shorter Viterbi paths to grow since the score of longer paths is reduced. The normalization length function can diminish the error rate significantly (see sec. 6).

4. "WIZARD OF OZ" EXPERIMENTS AND TEST DATA DESCRIPTION

The 3-D scene editor (see sec. 2) was tested in "Wizard of Oz" experiments employing a human "wizard" to recognize the gestures and to control the scene editor remotely. During the experiments the test persons invented over 60 different gestures to operate the editor. The 41 most frequently used gestures have been selected forming a catalog which is the basis of all the following tests (see table 1).

All the training and test material is contained in a continuous video sequence 46 minutes in length containing gestures of a single person. The motion JPEG compressed version of the video takes about 2 Gigabyte of disk space and contains noninterlaced images at a rate of 50 fields per second. The image size after the color segmentation process (see sec. 3.1) is 360×288 pixels. The camera was mounted above the table observing the desk area between the user and the monitor. The area was large enough to perform gestures using either one or two hands.

The video contains each of the 41 gestures at least 41 and at most 64 times. Beginning and end of the gestures were labeled by hand allowing to train the models and to evaluate the recognition process. 26 versions of each gesture were cut out and used to train the models with isolated gestures. The recognition was made on the whole continuous data consequently containing about 50 % gestures known from training and 50 % unknown gestures.

5. EVALUATION CRITERIA AND PARAMETER SETTINGS

A motion segment that reaches from time t_b to t_e is considered as correctly *detected* if it lies within a detection interval of ± 50 images around a manually labeled gesture. This interval is chosen very large to be sure to obtain any gesture candidate for the subsequent classification. The *detection rate* r_d is the "ratio of correctly detected motion segments to the total number of valid gestures". The *false detected rate* f_d represents the relative number of wrongly detected motion segments. The *multiple detection rate* $r_{d,mult}$ is the relative number of motion segments that are repeatedly assigned to one gesture.

Similar to the above definition, a gesture that ends at time t_g is defined as correctly *recognized* if the system indicates it at a time t_p[2] that lies within an interval of ± 50 images around t_g. The temporal recognition delay is $t_d = t_p - t_g$. The *recognition rate* r is the "ratio of correctly recognized gestures to the total number of valid gestures". The *multiple recognition rate* r_{mult} measures correctly but repeatedly recognized gestures. $\bar{t}_d$ is the *average recognition delay* of correctly recognized gestures. The *false accept rate* f is measured in fa/kg/h = "number of wrongly accepted gestures/number of key gestures/hour"[3].

The HMM and spotting parameters have been extensively varied to empirically find the optimal recognition results. These optimal parameters are the same in all the result tables 2–4: maximum order of HMIs $H = 2$, smoothing interval parameters $\tau_{sb} = 30$ and $\tau_{se} = 0$, peak detection interval parameters $\tau_{pb} = 10$ and $\tau_{pe} = 1$, minimum temporal peak distance $t_{dist} = 10$, number of HMM states for isolated recognition $N = 5$ and for spotting $N = 15$.

6. EXPERIMENTAL RESULTS

6.1. Motion detection and isolated recognition

The motion detection parameters (see sec. 3.2) were determined by a numerical optimization process. Three different optimization strategies M1–M3 were used: M1 is emphasizing a maximum r_d, M2 a maximum r_d combined with a minimum $r_{d,mult}$, and M3 a minimum $r_{d,mult}$ (see table 2). The appropriate detection results demonstrate that more than 90 % of the possible gestures can only be detected if a high multiple detection rate is tolerated (see table 2, M1).

The classification results of the motion segments prove that a high multiple detection rate results in a high false recognition rate: obviously many motion segments are nonmeaningful and *cannot* be correctly recognized (see table 3).

[2] $t_p = t_e$ for motion detected gestures.

[3] The number of different key gestures is 41 (see table 1).

	m_{thres}	τ_{bmin}/ τ_{emin}	τ_{lmin}/ τ_{dmin}	r_{d}	f_{d}	$r_{\text{d,mult}}$
M1	0.670	3/3	6/4	91.64	4.98	36.24
M2	0.700	3/5	6/3	88.21	4.72	25.75
M3	0.391	6/2	14/5	74.51	3.01	7.06

Table 2: Optimal setting of motion detection parameters (m_{thres}, τ_{bmin}, τ_{emin}, τ_{lmin}, and τ_{dmin}) and detection results (r_{d}, f_{d}, and $r_{\text{d,mult}}$) using optimization strategies M1–M3

		r	f	r_{mult}	t_{d}
M1		82.29	24.70	9.87	-1.15
M2		80.17	18.20	7.32	-2.02
M3		56.70	15.60	1.82	-1.82
SP	$S_{\text{rel}} = \infty$	95.48	24.73	0.26	18.18
	$S_{\text{rel}} = 0.0266$	90.03	11.38	0.21	18.33

Table 3: Recognition results (r, f, r_{mult}, and $\bar{t_{\text{d}}}$) based on motion detection (parameter settings M1–M3: $L = 256$, $N = 5$) and based on spotting (SP: $L = 256$, $v = 1.0$, $N = 15$)

6.2. Spotting

If spotting is used instead of motion detection and isolated recognition, the recognition rate is more than 13 % higher at the same false accept rate (see table 3). The remaining multiple detection rate of 0.26 % demonstrates that the spotting method is hardly disturbed by random motion that is close to the actual gestures. Applying a score threshold, the false accept rate can be even halved at an acceptable recognition rate of 90 %. The recognition delay of about 18 images corresponds to a system reaction time of 0.36 seconds which is insignificant for a visual dialog application.

The spotting results can even be improved if the normalization function parameter v is raised (see eq. (7) and table 4). Increasing v without using a rejection threshold, diminishes the false accept rate significantly (more than 40 %) while the recognition rate is only reduced slightly (less than 4 %). At a constant recognition rate of 90 % the false accept rate can still be improved by 13 %.

7. CONCLUSION

A one-stage and a two-stage approach to continuous hand gesture recognition were compared. A realistic visual dialog scenario provided video data containing a typical mixture of meaningful und non-meaningful motion and a catalog of 41 different gestures. The results proved that the one-stage HMM-based spotting method is far superior to the straight-forward, two-stage algorihm that combines an independent motion detection with an isolated HMM recognition. The spotting could be further improved introducing a new implicit duration modeling.

		v			
		1.0	1.5	2.0	3.0
$S_{\text{rel}} = \infty$	r	96.73	96.11	95.59	92.94
	f	24.41	20.48	17.44	14.01
	r_{mult}	0.36	0.42	0.42	0.36
	t_{d}	18.19	18.34	18.44	18.67
$r = 90$	f	9.89	9.32	8.59	9.86
	r_{mult}	0.26	0.26	0.26	0.21
	t_{d}	18.38	18.52	18.61	18.78
	S_{rel}	0.0229	0.0230	0.0225	0.0277

Table 4: Recognition results based on spotting (r, f, r_{mult}, and $\bar{t_{\text{d}}}$) for different v and S_{rel} ($N = 15$, $L = 512$)

8. REFERENCES

[1] G. I. Chiou, J.-N. Hwang: *Lipreading from Color Motion Video*. ICASSP 1996, Atlanta, Vol. 4, pp. 2156–2159, 1996.

[2] M.-K. Hu: *Visual Pattern Recognition by Moment Invariants*. IRE Trans. Inform. Theory, vol. IT-8, pp. 179-187, Feb. 1962.

[3] X. D. Huang, Y. Ariki, M. A. Jack: *Hidden Markov Models for Speech Recognition*. Edinburgh University Press, 1990.

[4] P. Morguet, M. Lang: *Feature Extraction Methods for Consistent Spatio-Temporal Image Sequence Classification Using Hidden Markov Models*. ICASSP 1997, Munich, Vol. 4, pp. 2893–2896, 1997.

[5] P. Morguet, M. Lang: *A Universal HMM-Based Approach to Image Sequence Classification*. ICIP 1997, Santa Barbara, USA, Vol. 3, pp. 146–149, 1997.

[6] P. Morguet, M. Lang: *An Integral Stochastic Approach to Image Sequence Segmentation and Classification*. ICASSP 1998, Seattle, USA, Vol. 5, pp. 2705–2708, 1998.

[7] P. Morguet, M. Lang: *Spotting Dynamic Hand Gestures in Video Image Sequences Using Hidden Markov Models*. To appear in proceedings of the ICIP 1998, Chicago, USA, 1998.

[8] T. Starner, A. Pentland: *Visual Recognition of American Sign Language Using Hidden Markov Models*. International Workshop on Automatic Face- and Gesture-Recognition 1995, Zürich, pp. 189–194, 1995.

[9] J. Yamato, J. Ohya, K. Ishii: *Rocognizing Human Action in Time-Sequential Images using Hidden Markov Model*. IEEE Comp. Vision and Pattern Recog. 1992, pp. 379–385, 1992.