The Gesture Interpretation Module

Rui Ping Shi, Johann Adelhardt, Anton Batliner, Carmen Frank, Elmar Nöth, Viktor Zeißler, Heinrich Niemann

Friedrich-Alexander Universität Erlangen-Nürnberg, Germany
{shi,adelhardt,batliner,frank,noeth,zeissler,
niemann}@informatik.uni-erlangen.de

Summary. Humans make often conscious and unconscious gestures, which reflect their mind, thoughts and the way these are formulated. These inherently complex processes can in general not be substituted by a corresponding verbal utterance that has the same semantics (McNeill, 1992). Gesture, which is a kind of body language, contains important information on the intention and the state of the gesture producer. Therefore, it is an important communication channels in *human computer interaction*.

In the following we describe first the state of the art in gesture recognition. The next section describes the *gesture interpretation module*. After that we present the experiments and results for recognition of user states. We summarize our results in the last section.

1 State of the Art

1.1 Applications of Gesture

Gesture can be used in a wide range of applications: gesture in conventional human computer interaction (HCI), interaction through linguistic gesture and manipulation through physical contact. We cover each of these in the following.

Gesture in Conventional HCI

Under the window, icon, menu and pointing device (WIMP) paradigm, the use of mouse and pen of a graphic tablet such as that of the Wacom[1] Company are typical example applications of gesture. This kind of gestures with the help of pointing devices is intensively employed in *computer aided design* (CAD) (Sachs, 1990) and online handwriting recognition (Buxton et al., 1985). In the literature this category of gestures is called pen-based gesture. Rubine introduced the GRANDMA system (Rubine, 1991), in which the user is allowed to define arbitrary gestures interactively. These user-defined gestures can be input either through a mouse or with the help of a pen. The system is able to learn the static and dynamic properties of the gestures on the basis of some training data and subsequently analyzes them in real time.

[1] http://www.wacom.com

Interaction Through Linguistic Gesture

American Sign Language (ASL) and audio–video–speech recognition represent applications of linguistic gestures. In the ASL there exist, as a general rule, strict syntax, semantics and their mapping in the gesture configuration similar to their "acoustic" counterpart — speech. The understanding of ASL takes place in the space in which gesture and its grammatical structure are expressed through the hand movement and posture. Humans also use facial expression as well as head and body posture to support their expressions. In Attina et al. (2003) a system is described in which speech recognition is supported by the conventionalized gestures, similar to ASL. This kind of application often deals with hearing-impaired patients. Lip-reading is the only reliable way for these patients to communicate with other people in daily life, assumed that they have no hearing device and have not yet learned some strict sign language like ASL. The speech accompanied by such gestures is referred to as Cued Speech (Cornett, 1967). Moreover, linguistic gestures can be used in HCI as artificial commands, which a computer can interpret and execute. The Morpha system in Lütticke (2000) can be controlled through dynamic gestures, which are learnde by the system offline. This kind of gesture is more intuitive and flexible in comparison with the gestures in ASL.

Manipulation Through Gesture

Through the commitment of data glove and touch screen, the user can physimechanically interact with (virtually) presented objects in 2D/3D space. Virtual reality is by all means the direct application of such gestures, in which the user gesticulates with the virtual environment with relatively free gestures, as if he or she were also a part of that world. The use of the data glove can even improve the impression of authenticity by providing the user with feedback in response to the gesture input such as through pressure and temperature.

1.2 Approaches in Gesture Recognition

There are different methods in the field of gesture analysis. These methods are shown in the following with respect to sensor and recipient.

From the View of the Sensor

The very first attempts were data gloves, which were equipped with light sensors made of fibreglass (Zimmerman and Lanier, 1987; Marcus and Churchill, 1988; Eglowstein, 1990). The light sensors installed on the fingers convert each finger movement, like bending, rotation and outstretching, into analog signals, which are in turn used to calculate the angles of the joints of the fingers, their respective positions and the orientation of the hand. Different configurations and postures of the hand can be interpreted as commands for the computer. However, the data glove has a big disadvantage due to its unwieldiness: The user has to carry a lot of hardware with him- or herself which consequently makes this kind of gesture interaction unnatural.

Table 1. An overview of gesture analysis systems

Author (see references)	Hardware	Interaction	System
Oviatt (1999)	Pen	Gesture, speech	Service transaction
Rubine (1991)	Mouse	Mouse-based	GRANDMA
Buxton et al. (1985)	Mouse	Mouse-based	Editor for music note
Waibel and Yang	Graphic tablet, pen	OCR	INTERACT
Raab et al. (1979)	Magnetic field	–	Person tracking
Bolt (1980)	ROPAMS	gesture, speech	“Put-that-there”
Azuma (1993)	Ultrasonic	–	Person tracking
Fels and Hinton (1993)	VPL DataGlove	Speech synthesis	Glove-Talk
Kurtenbach and Baudel (1992)	VPL DataGlove	Presentation	HyperCard
Wu and Huang	Camera	Hand posture	Paper–Rock–Scissors
Quek (1995)	Camera	Hand gesture	Finger mouse
Kettebekov and Sharma	Front camera	Gesture, speech	*i*Map
Akyol et al. (2001)	Infrared camera	Gesture	Car infotainment

Electromagnetic fields (Raab et al., 1979) for gesture localization and recognition are also popular. However, the high acquisition costs, the sensitivty to noise and the short working range are the negative factors, which must be accounted for. In contrast, the video-based gesture analysis with the help of a CCD camera and the corresponding image processing technique seems more promising on account of its uncumbersome hardware. By using a camera, a set of modalities in addition to hand gestures can be integrated into the HCI, e.g., lip, gaze direction, head movement and interpretation of facial expressions. Several typical systems of gesture analysis are listed in Table 1, some of which operate simultaneously with speech. There are also other methods, which use special hardware to achieve high throughput and efficiency, e.g., the SiVit (Siemens Virtual Touchscreen) unit introduced in Maggioni (1995). SiVit is also integrated in SmartKom.

From the View of the Recipient

Video-based gesture analysis is advantageous according to the comparison above, thanks to the ever-improving efficiency and capacity of the computer hardware nowadays. The current major problem lies in the increasing demand for algorithms, which should be fast, robust, traceable, efficient, reliable, modularized and fault tolerant. There are mainly three different methods in the video-based gesture analysis: marker-based, hand model-based and view-based. Due to the nonconvex volume of the hand, many researchers attach markers to the hands, which are placed at certain positions of the hand. Normally, they have a special color or geometric form, with which the detection of hand and fingers becomes easier such as in case of occlusion of some part of the hand without markers. This is an indirect method and thus makes the gesture interaction unnatural. The hand model provides a full-fledged modeling of the respective finger joints and postures. Therefore, this method is able, theoretically, to analyze any gesture, given enough training data. In practice, however, the computing complexity and the lack of efficiency hinder its spread, although it can shed

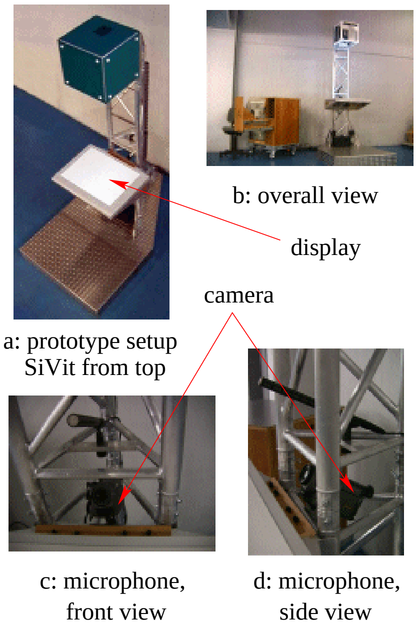

Fig. 1. SMARTKOM demonstration system. **a** prototype with integrated SiVit; **b** overall view, with camera for facial expression analysis and microphones for speech analysis (**c,d**)

light on the solution of many practical problems. The view-based method utilizes the pixel values as its starting point, which can be either directly used as features or be converted to a suitable form through some transformations. It has a relatively small computing intensity and is therefore preferred in practice.

2 Module description

2.1 Gesture in SmartKom

Figure 1 shows the set up of an intended SMARTKOM system with an integrated SiVit unit at the top of the machine. A similar version of this system was used to collect the gesture data in the Wizard-of-Oz experiments. The SiVit unit consists of a video projector, an infrared camera and a virtual touch screen, which is not sensitive to vandalism. The system works in the following manner: The video projector projects all the graphical user interface (GUI) information onto the display, where the user can use his hand to select or search for objects. The infrared camera captures the trajectory of the users hand for the gesture analysis. Gestures are captured together with the recording of the face via video camera, and speech through a microphone array. The positions of these components are pointed out in Fig. 1.

In SMARTKOM the hand gesture is used in two categories: object manipulation and contribution to user state recognition. An introduction to the latter subject is given separately in Streit et al. (2006), while experiments referring to user state recognition as well as object manipulation are shown in the following section.

2.2 Work Course of the Gesture Module

Apart from selecting virtual objects by gesture, the user state, which is expressed through gesture and describes the mood of the user, influences to some degree the way of gesturing: If the user gets annoyed, his gesture tends to be quick and iterating, while it becomes short and determined if the user is satisfied with the service and the information provided by the system. Both gesture and speech indicate the user state and both complement each other. Thus, we will base our experiment on a joint sample set of speech, gesture and facial expression. Since we deal with constantly changing user states, it is clear that the central point of this issue is concentrated on the dynamics of the gesture and its interpretation, instead of focusing on its segmentation from background. In SMARTKOM, all exchanged information packets are coded in an XML format.

2.3 Data flow in the gesture module

Figure 2 shows the data flow in the gesture module, which consists of two main input streams and two output streams, all coded in the XML format. Based on the two assignments of the gesture model, it reads from the data pool *generated.presentation* the geometric coordinates of the virtual objects on the GUI surface, and from *recognized.gesture* the trajectory of the gesture. After aligning the time stamps of these two packets and parsing these two XML packets, the module decides which object the user has chosen or manipulated with regard to the pointing gesture position and virtually depicted GUI objects. In this scenario, the hand gesture takes over the role of a mouse. Afterwards an object hypothesis will be generated in XML format and sent to the *gesture.analysis* pool, whose content can be evaluated further by other modules to respond to the user gesture input by calling the corresponding service such as cinema information or TV programs.

In the case of user state recognition a similar process happens, where the raw gesture data go through XML parsing, feature extraction and classification by Hidden Markov Models (HMMs). As a result, the recognized user state hypotheses will also be sent to the user state pool.

2.4 Hidden Markov Models and Gesture Analysis

HMMs are a suitable model to incorporate temporal continuity. Temporal continuity here means that a pixel of the gesture trajectory belongs to a certain category (state) for a period of time. If a pixel moves at a high speed at a given time, it is likely that this pixel will still keep moving fast at the next time step. HMMs are able to learn the observation distributions for different categories (hidden states) from the trajectory of the gesture. The training data are recorded in a system similar to the one depicted

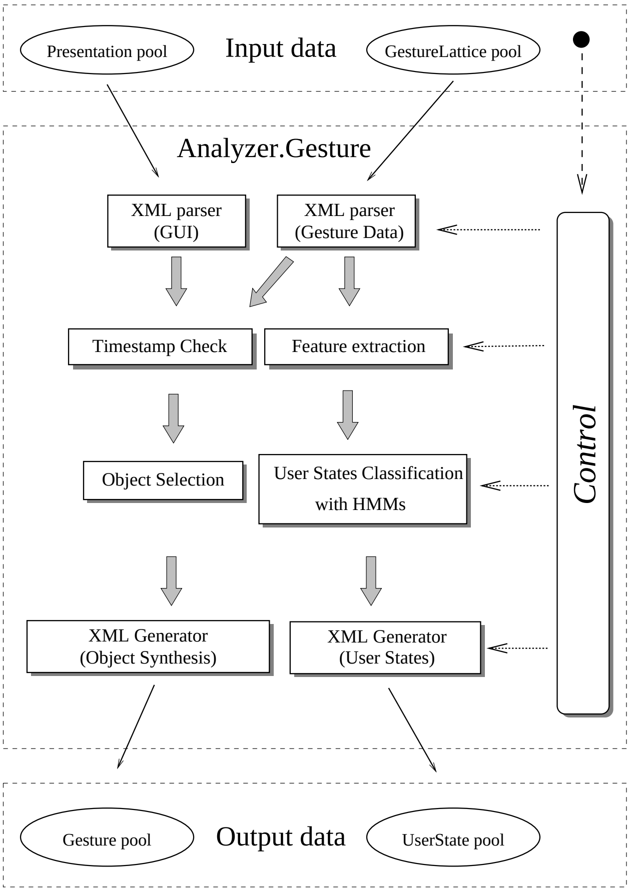

Fig. 2. Architecture of the gesture module

in Fig. 1. In this paper, each feature vector will be assigned to one of three or four hidden states of the HMM (see Sect. 2.7).

We use the standard *Baum–Welch* reestimation algorithm for the training, which is based on the expectation maximization (EM) algorithm (Rabiner and Juang, 1986), and the standard *Forward Algorithm* to solve the classification problem. A detailed description of these algorithms can be found in Rabiner and Juang (1986, 1993); an example of how to apply these algorithms can be found in Rabiner (1989). Here we use discrete HMMs because of their simplicity.

2.5 Feature Extraction

In order to incorporate the temporal continuity, we choose four features: trajectory variance, instantaneous speed, instantaneous acceleration and kinetic energy as the feature vector, which best represents the motion and the dynamics of the gesture. The

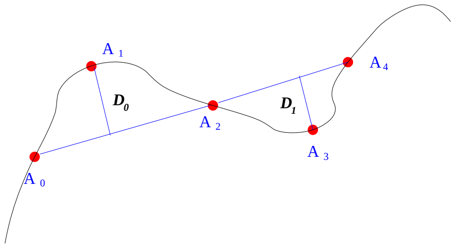

Fig. 3. Typical trajectory of a pointing gesture with sampling points $A_I, I = 0...N$ points and geometric variances $D_I, I = 0..N-1$, which show the intensity of fluctuation of the gesture

continuous two-dimensional coordinates (trajectories) and the time stamp, which are recorded by the SiVit unit, are the most important information on the dynamics of the gesture. The reason for computing the instantaneous velocity v over time is for the system to learn from the behavior of the user's gesture. That is, with simple data analysis, it would be possible to determine trends and anticipate future moves of the user. The next set of data points is the acceleration a of the gestures, which is easily computed by approximating the second derivative of the position coordinate. Kinetic energy K, which is just the square of the velocity while the mass is neglected, is also a significant factor.

In our feature set, the trajectory variance is also included. This is the geometric variation or oscillation of the gestures with respect to their moving direction. A large value of this variance can indicate that the user gesticulates hesitantly and moves his or her hand around on the display, while a determined gesture leads to a small variance. Figure 3 shows how the trajectory variance D is computed. So we have a feature vector

$$f = (v, a, K, D). \tag{1}$$

The vector D can be computed every N points along the gesture trajectory. Other possible features are, e.g., the number of pauses of a gesture, the transient time before and after a pause, the transient time of each pause relative to the beginning of the gesture, the average speed, and the average acceleration or change of moving direction. However, in this study we just consider the feature vector shown in Eq. (1).

2.6 Modification of User State Classes

As mentioned above, the goal of SMARTKOM is the combination of all three input modalities. Gesture, as one of the input channels, must define its own output to contribute to the fusion of the analysis of the three modalities. In contrast to facial and prosodic analysis, where four user states are defined, *neutral*, *angry*, *joyful* and *hesitant*, we define in gesture analysis only three user states: *determined*, *angry* and *hesitant*. The reason for making this mapping is the intuition that normally people cannot tell if the user is neutral or joyful by only observing his or her gesture. This

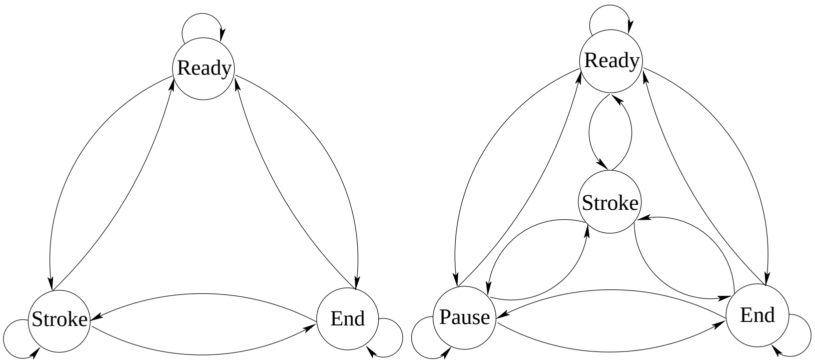

Fig. 4. Ergodic HMMs with different numbers of hidden states for gesture analysis

was confirmed by the preliminary experiments, where *neutral* and *joyful* had a high confusion. We decided thus in favor of the three states topology. The user state *determined* is given if the user knows what he wants from SMARTKOM, e.g., if he decides to zoom in a part of a city map on the GUI by pointing to it. If the user gets confused by SMARTKOM and does not know what to choose, his gesture will probably ponder around or zigzag among different objects presented on the SMARTKOM GUI. Finally, if he feels badly served by SMARTKOM or if the information presented is not correct, he can use gestures in such a way as to show a strong negative expression like a windshield wiper, which in our context corresponds to the user state *angry* in facial expression.

2.7 Choice of Different Topologies

For the HMMs, we evaluated different topologies; an HMM with three or four states gave the best results. We suppose that a gesture consists of some basic states such as *ready*, *stroke*, *end* and/or *pause*. This can be observed in the production of the gesture: The user moves her hand to a start position, then makes a gesture consisting of several strokes, probably with pauses in between, and finally ends her gesture. An alternative is to merge *pause* and *ready*. We also tried different connection schemata; the simplest one is an ergodic HMM, while a partially connected HMM better corresponds to the correct physical order of each state (Fig. 4). The conventional left–right HMM model is also an alternative that has been successfully used in speech recognition.

3 Experiments

Tables 2, 3 and 4 show the results of the gesture analysis. We can see that the user state *hesitant* is sometimes mismatched with *angry*. The reason is that some users, whose gestures are used in the training set, made similar gestures like those in *angry*

Table 2. Confusion matrix of user state recognition with gesture data using ergodic HMM (see Fig. 4, CL: classwise averaged recognition rate)

Reference user state	3 HMM states (%)			4 HMM states (%)		
	Determined	Hesitant	Angry	Determined	Hesitant	Angry
Determined	**61**	5	34	**80**	15	5
Hesitant	5	**72**	23	15	**77**	8
Angry	10	6	**84**	10	18	**72**
CL	72			76		

Table 3. Confusion matrix of user state recognition with gesture data using ergodic HMM (LOO, see Fig. 4, CL: classwise averaged recognition rate)

Reference user state	3 HMM states (%)			4 HMM states (%)		
	Determined	Hesitant	Angry	Determined	Hesitant	Angry
Determined	**62**	5	33	**75**	7	18
Hesitant	5	**74**	21	13	**74**	13
Angry	8	8	**84**	30	8	**62**
CL	73			70		

Table 4. Confusion matrix of user state recognition with gesture data using nonergodic HMM (CL: classwise averaged recognition rate)

Reference user state	3 HMM states (%)			4 HMM states (%)		
	Determined	Hesitant	Angry	Determined	Hesitant	Angry
Determined	**72**	16	12	**40**	49	11
Hesitant	32	**45**	23	2	**70**	28
Angry	60	12	**28**	2	24	**74**
CL	48			61		

states, in that the windshield wiper movement has the same zigzag only with different dynamics and speed. Probably, some persons gesticulate slowly while indicating anger, thus their corresponding gestures may have similar properties like those of a *hesitant* state.

Table 5. Confusion matrix of User state recognition using left–right HMM (CL: classwise averaged recognition rate)

Reference user state	3 HMM states (%)			4 HMM states (%)		
	Determined	Hesitant	Angry	Determined	Hesitant	Angry
Determined	**63**	4	33	**66**	6	28
Hesitant	6	**47**	47	13	**51**	36
Angry	20	4	**76**	30	4	**66**
CL	62			61		

Another reason for a wrong classification is that the training data for the user state *determined* consists of those from *joyful* and *neutral*; the latter makes the HMM for *determined* biased towards *hesitant* in Table 4 with four internal states. In general, the classification has a classwise (CL) averaged recognition rate of 72% for three internal states and 76.3% for four internal states, while the leave-one-out (LOO) test achieves 73% for three internal states and 67% for four internal states. Table 5 shows the recognition result when using a conventional left–right HMM model.

4 Conclusion

Gesture is an important communication channel in HCI, whose usage ranges from direct manipulation of object to indication of user states as shown above. These two models have been successfully integrated into the SMARTKOM demonstrator, which runs as an autonomous service agent between the user and different information sources through gesture, speech and facial expression. The user is, therefore, free to communicate with the system, similar to talking with another human. Furthermore, the gesture, speech and facial expression complement each other in a redundant way so that the demand of precise expression in each modality can be relaxed and thus extend the applicability with respect to prospective users.

References

S. Akyol, L. Libuda, and K.F. Kraiss. Multimodale Benutzung adaptiver Kfz-Bordsysteme. In: T. Jürgensohn and K.P. Timpe (eds.), *Kraftfahrzeugführung*, pp. 137–154, Berlin Heidelberg New York, 2001. Springer.

V. Attina, D. Beautemps, M.A. Cathiard, and M. Odisio. Toward an Audiovisual Synthesizer for Cued Speech: Rules for CV French Syllables. In: J.L. Schwartz, F. Berthommier, M.A. Cathiard, and D. Sodoyer (eds.), *Proc. AVSP 2003 Auditory-Visual Speech Processing*, pp. 227–232, St. Jorioz, France, September 2003. ISCA Tutorial and Research Workshop.

R. Azuma. Tracking Requirements for Augmented Reality. In: *ACM*, vol. 36, pp. 50–51, July 1993.

R. Bolt. “Put-That-There”: Voice and Gesture. In: *Computer Graphics*, pp. 262–270, 1980.

W. Buxton, R. Sniderman, W. Reeves, S. Patel, and R. Baecker. An Introduction to the SSSP Digital Synthesizer. In: C. Roads and J. Strawn (eds.), *Foundations of Computer Music*, pp. 387–392, Cambridge, MA, 1985. MIT Press.

R.O. Cornett. Cued Speech. *American Annals of the Deaf*, 112:3–13, 1967.

H. Eglowstein. Reach Out and Touch Your Data. *Byte*, 7:283–290, 1990.

S. Fels and G.E. Hinton. Glove-Talk: A Neural Network Interface Between a Data-Glove and a Speech Synthesizer. In: *IEEE Transactions on Neural Networks*, vol. 4, pp. 2–8, 1993.

S. Kettebekov and R. Sharma. Multimodal Interfaces. http://www.cse.psu.edu/~rsharma/imap1.html. Cited 15 December 2003.

G. Kurtenbach and T. Baudel. Hypermarks: Issuing Commands by Drawing Marks in Hypercard. In: *ACM SIGCHI*, p. 64, Vancouver, Canada, 1992.

T. Lütticke. Gestenerkennung zur Anweisung eines mobilen Roboters. Master's thesis, Universität Karlsruhe (TH), 2000.

C. Maggioni. Gesture Computer — New Ways of Operating a Computer. In: *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, pp. 166–171, 1995.

A. Marcus and J. Churchill. Sensing Human Hand Motions for Controlling Dexterous Robots. In: *The 2nd Annual Space Operations Automation and Robotics Workshop*, Dayton, OH, July 1988.

D. McNeill. *Hand and Mind: What Gestures Reveal About Thought*. University of Chicago Press, Chicago, IL, 1992.

S. Oviatt. Ten Myths of Multimodal Interaction. *Communications of the ACM*, 42 (11):74–81, 1999.

F. Quek. FingerMouse: A Freehand Pointing Interface. In: *Int. Workshop on Automatic Face- and Gesture-Recognition*, pp. 372–377, Zurich, Switzerland, June 1995.

F.H. Raab, E.B. Blood, T.O. Steiner, and H.R. Jones. Magnetic Position and Orientation Tracking System. In: *IEEE Transaction on Aerospace and Electronic Systems*, vol. 15, pp. 709–718, 1979.

L.R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In: *Proc. IEEE*, vol. 77, pp. 257–286, 1989.

L.R. Rabiner and B.H. Juang. An Introduction to Hidden Markov Models. *Acoustics, Speech and Signal Processin*, 3(1):4–16, 1986.

L.R. Rabiner and B.H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, Englewood Cliffs, NJ, 1993.

D. Rubine. Specifying Gestures by Example. In: *SIGGRAPH '91 Proceedings*, vol. 25, pp. 329–337, New York, 1991.

E. Sachs. Coming Soon to a CAD Lab Near You. *Byte*, 7:238–239, 1990.

M. Streit, A. Batliner, and T. Portele. Emotion Analysis and Emotion Handling Subdialogs, 2006. In this volume.

A. Waibel and J. Yang. INTERACT. http://www.is.cs.cmu.edu/js/gesture.html. Cited 15 December 2003.

Y. Wu and T.S. Huang. "Paper–Rock–Scissors". http://www.ece.northwestern.edu/~yingwu/research/HCI/hci_game_prs.html. Cited 15 December 2003.

T.G. Zimmerman and J. Lanier. A Hand Gesture Interface Device. In: *ACM SIGCHI/GI*, pp. 189–192, New York, 1987.