A 3D Gesture Recognition System for Multimodal Dialog Systems

Robert Neßelrath and Jan Alexandersson

DFKI GmbH
Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany
{rnessel,janal}@dfki.de

Abstract

We present a framework for integrating dynamic gestures as a new input modality into arbitrary applications. The framework allows training new gestures and recognizing them as user input with the help of machine learning algorithms. The precision of the gesture recognition is evaluated with special attention to the elderly. We show how this functionality is implemented into our dialogue system and present an example application which allows the system to learn and recognize gestures in a speech based dialogue.

Introduction

Within the last years the number of new technologies and appliances for home entertainment and household has vastly increased, confronting users with a high amount of new electrical devices and heterogeneous interaction concepts. In addition to the often difficult interaction with modern devices like computers and consumer electronics, the interaction with more traditional devices such as television, telephone, and in-car-appliances, gets more and more complicated with the growing number of functionalities. This surplus creates the problem of accessing all of these functionalities in an easy way. Most systems try to solve this problem with a more or less well structured menu concept. In practice, however, a lot of users still have difficulties to navigate through different menus and find the operations they are looking for.

While users with technical expertise and a great interest in modern devices cope with the interaction after some time, especially the elderly or persons with cognitive disabilities never get a chance to use all electrical devices in their home. Within the i2home project [1] new concepts for ambient assisted living are developed based on existing and evolving industry standards, for example the Universal Remote Console (URC) Standard [2]. One approach [3] is based on the multimodal dialogue platform ODP which has been used successfully in several research and industrial projects [4]. Our dialogue system supports the user in the problem-solving process by taking into account the discourse context and providing several communication modalities. Input commands of the user are given via speech, click gestures, or a combination of these. System answers are given either visually or acoustically through speech and sounds. The i2home dialog system allows the user to control its environment by speech interaction and a well designed and easy to use graphical user interface. This allows user to control their kitchen, a reminder and television.

In the future we will develop systems that address the needs of single users as well as user groups not only with cognitive but also with physical disabilities. That confronts us with the individual problems of users that are too limited to interact with a system even by click gestures or speech. In order to provide the opportunity for those persons to communicate with the system, new input devices and modalities must be included. These could be gyroscopes and accelerators but also eye-trackers. All of these devices have the advantage that they not depend on humans using their fingers to manipulate their environment, as they do traditionally. This gives us the opportunity to enhance our input modalities by taking into account other aspects like movements of arms, eyes or head.

In this paper we introduce the integration of dynamic gesture input into the multimodal dialogue system. To record the gestures we use an accelerometer which is integrated in the common input device for the Nintendo Wii, the Wii Remote. The measured values are used to describe the movement of the arm in a three dimensional space and are trained with machine learning systems, in order to recognize the executed dynamic gestures.

The next chapter introduces gestures as part of human communication and demonstrates their use in man-machine communication. The following one deals with TaKG, a toolkit for classifying gestures, and its integration into our dialogue system. For usability tests we evaluated the gesture recognition system with elderly persons in a retirement home.

Gestures

Human communication is a combination of speech and gestures. Gestures are part of the nonverbal conversation and are used consciously as well as subconsciously. Gestures are a basic concept of communication and were used by humans even before speech developed, they have the

potential to be a huge enrichment to an intuitive man-machine communication.

One distinguishes between dynamic and static gestures. Static gestures are used for finger spelling among other things. In this case only the position of hand and the alignment of the fingers provide the information for the communicative act. Dynamic gestures additionally contain a movement and in most cases have either a pantomimic meaning, i.e. imitating an action or a symbolic meaning, for example waving to someone.

Gestures in Related Projects

Generally, there are two ways for recording gestures. Non-instrumental projects recognize hand and finger postures with cameras and image processing algorithms [5]. Other projects use instruments for recording, for example sensor gloves or hand devices with integrated sensors like accelerometers or gyroscopes [6] [7]. This is also the concept of the Wii game console and it is their device that is used for this project. In Wii games, often easy properties are used for interpreting gestures, for example the strength and the direction of a movement. The tool LiveMotion[1] from Ai-Live is a framework for Wii game developers focused on learning and recognizing more complex gestures. The creation of motion recognizers is mastered by showing gesture examples without coding or scripting. Recognition should be very fast and without using buttons but is only usable by game developers who have a contract with Nintendo.

A worthwhile goal to use gestures as input is to integrate sensors into devices of everyday life and to recognize device-related gestures. For example the gestures used during operating a mobile device can be taken to recognize scenarios, for example picking up a ringing mobile phone from the table and hold it to the ear as a scenario for accepting a call [8].

Wii Remote Acceleration Sensors

In this work the movement of a dynamic gesture is detected by an ADXL330 accelerometer which is integrated in the Wii Remote controller. The ADXL330 measures acceleration values with 3 axis sensing in the interval -/+ 3g. The acceleration is described in a right-handed Cartesian coordinate system. A Wii Remote, which lies bottom side down on a table, measures the value of 1g in the direction of the z-axis. This is the force the hand needs to exert against gravity and thus an unmoved Wii Remote always measures the absolute acceleration value of 1g. In free fall the absolute value is zero.
The complete movement of the hand within the three-dimensional space can be described by observing acceleration in a series respective to the time. Figure 1 shows the x-axis measurement of a hand movement to the left, that is the axis of interest for this movement. First the curve contains positive amplitude for the acceleration. During the time of the movement with constant speed the acceleration value first goes down to zero and then deceleration causes it to become negative. Figure 2 shows the measurement of a movement to the right and the same curve is observed only with amplitudes in opposite directions. Since hand gestures take place in 3D-space, the sensor values of three accelerometers are part of one measurement. Figure 3 shows the measurement of a circle movement.

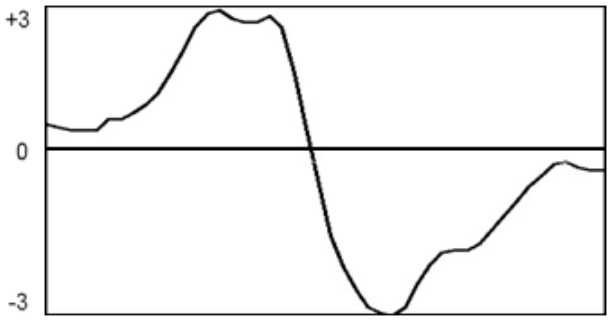

Figure 1: Acceleration data of a movement to the left

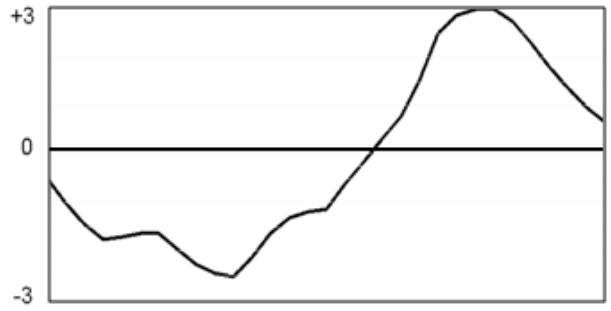

Figure 2: Acceleration data of a movement to the right

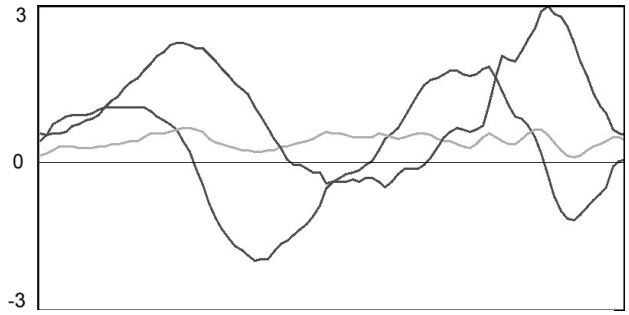

Figure 3: 3D- acceleration data of a circle gesture

WiiMote Recorded Gestures

The character of the WiiMote as an input device leads to some technical constraints which influence the gesture type we can recognize. This includes only dynamic gestures, i.e. the movement of the hand, which is described and modified by the movement path relative to the start position, as well as by the alignment of the WiiMote holding hand, and by the movement speed.

In difference to mouse gesture recognition frameworks [9] or gesture controlled internet browsers [10], the recogni-

[1] http://www.ailive.net/

tion of WiiMote recorded gestures is not limited to two dimensions. Movement in space includes the third dimension and leads to different problems than the recognition of mouse gestures in 2D-space. The 3D-movements are described by their acceleration values, mouse gestures by an array of positions on a plane.

Gesture Recognition

The path of a gesture recorded by the WiiMote involves all three dimensions, giving multidimensional time series in which not only exceptional measurements are included but also their temporal relations to other dimensions. For gesture recognition this means that it is not sufficient to examine the dimensions separately. There must also be synchronization between them.
Another problem is that the measurements for only one gesture differ in time, movement-path and speed with every execution. Comparison of two measurements of the same gesture thus has to handle warps in a non-linear way by shrinking or expanding along the time axis and also the single space axes.

An algorithm which is often used to measure similarities between two signals is the Dynamic Time Warping (DTW). This algorithm calculates the distance between each possible pair of points out of two signals and finds the least expensive path through the resulting distance matrix using dynamic programming. The resulting path expresses the ideal warp between the two signals and synchronizes the signal in order to minimize the distance between the synchronized points. With some adaption it can also be used for multi-dimensional gestures [11].

Another approach is to use machine learning algorithms for classifying data. The WEKA framework [12] is a collection of machine learning algorithms implemented in Java and provides interfaces for the easy usage of the most common algorithms. Besides the DTW we also test algorithms that are included in WEKA to learn and recognize gestures: Support Vector Machines (SVM) und Neuronal Networks (NN). Because these algorithms need a fix number of attributes for an instance the acceleration data is preprocessed for input. This includes normalization of the values and interpolation of the measurement on a fix sized set of sampling points.

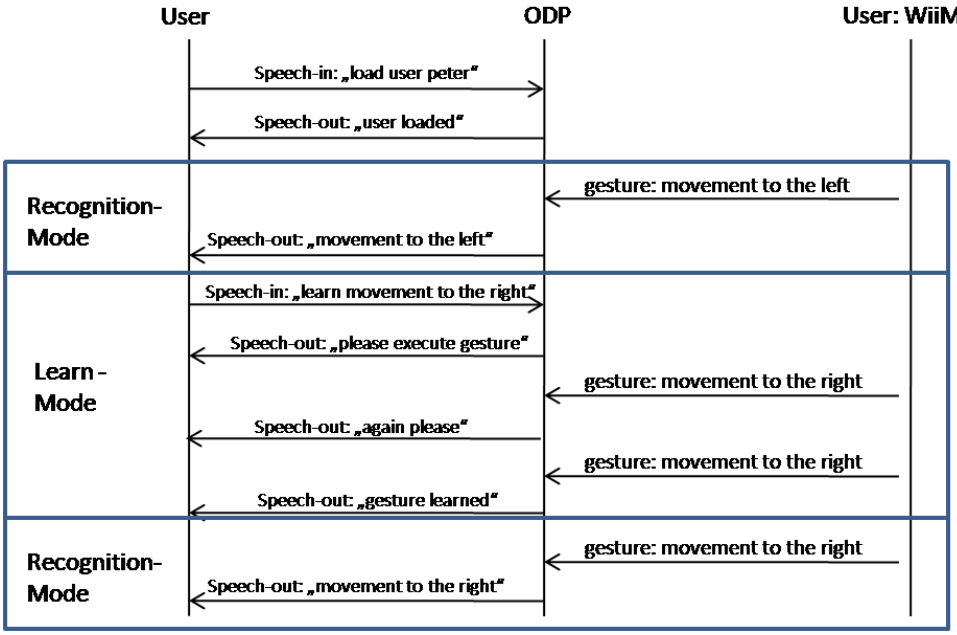

Figure 4: Typical cycle of the gesture training and recognition dialog

The TaKG Framework

The framework TaKG [13] is a toolkit for gesture recognition and serves to simplify the integration of gesture controlled interaction into applications. It implements needed functionalities for signal feature extraction and the recognition algorithms mentioned in the chapter before: SVM, NN and DTW.
Furthermore, TaKG is responsible for learning new gestures and organizing them into user specific training sets. Within a set, the information for every trained gesture is listed including the measured signal data and a *gesture tag* denoting the gesture. The main API contains the following functionalities: Load data for a special user, learn and delete gestures and classify new recorded gestures.
A gesture classifying request returns the gesture tag of the gesture in the training set with the highest similarity to the gesture which has been provided together with the request. Another option is to ask for a ranked list of all trained gestures. SVM and NN provide just a ranking, the DTW algorithm describes similarity based on Euclidian distance.

Gesture controlled calculator

One example application which was also used for evaluating the gesture recognition precision is a simplified calculator. The calculator contains buttons for the digits from 0-9, the operators plus (+), minus (-) and equals (=) and a clear button. Every button can be pressed by painting the appropriate figure into the air (or an arbitrary gesture the user associates with the button). The movement depends on the gestures the user performed for training the gesture classifier.

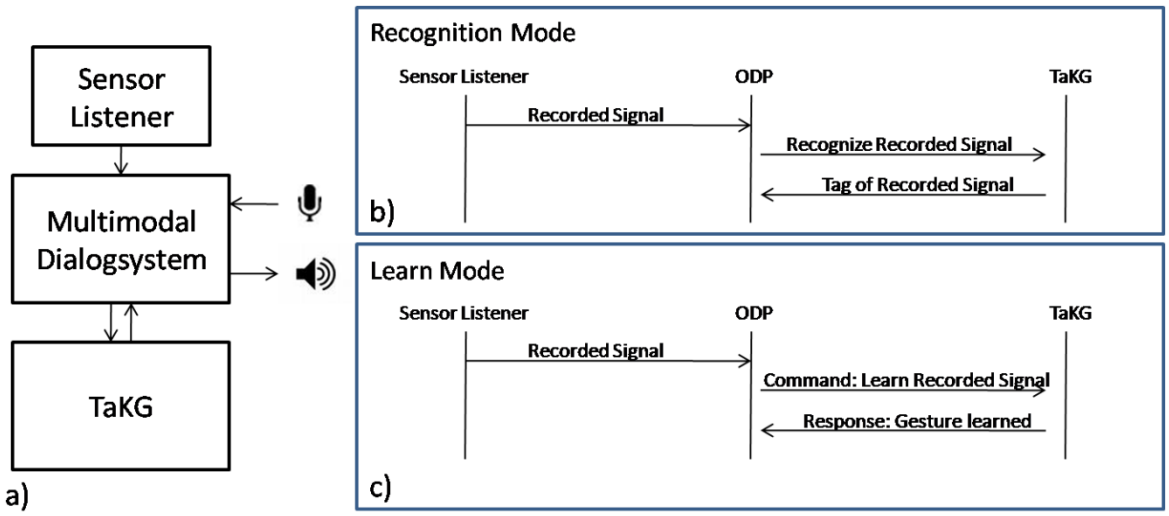

Figure 5: a) Module overview of the gesture training and recognition dialogue. b) Message Flow in Recognition Mode c) Message Flow in Learn Mode

Integration into a dialog system

Gesture Training and Recognition Dialog

The main intention in building a gesture recognition system was to integrate it into our multimodal dialogue system. Our example dialogue application uses speech and gestures to train the classifier with new gestures. During this process we distinguish between two signal processing modes, one mode is the learning mode, the other one the recognizing mode. The standard mode is 'recognize', this means that every incoming signal is classified and the detected tag is synthesized for speech output. By voice command the user can set the system into learn mode which adds a new received signal to the training data set. Figure 3 shows a typical dialogue between a user and the system. Additional speech commands allow the user to manage the training data set, i.e. to load data sets for other users or to delete already learned gestures or even complete gesture information about a user.

For implementation of gesture recognition the dialogue system is expanded with two new modules (fig. 4a). One module, the sensor listener, is informed about new recorded gestures. The second module addresses the classifier and works as an adapter to the TaKG.
The dialogue system in the middle is responsible for context-based reaction on new sensor inputs and the communication with the user. In recognition mode new recorded signals are sent to TaKG with a recognition request (fig. 4b). The received tag for the gestures is send to speech-synthesis and the user gets a speech reaction, announcing the user the recognized gesture. In learn mode, ODP sends a learn request to TaKG and controls whether the gesture has been learnt successfully. The result is committed to the user via speech.

Evaluation

The gesture system was evaluated with a heterogeneous group of participants, differing in age and gender. Thus we avoided that only younger persons with experiences in using modern input devices attended the test. Since we were especially interested in how elderly people would respond

Table 1: Evaluation Results. The numbers present the percentage of the correct recognized gestures.

Proband-Id	SVM	NN	DTW
Age 20-50			
M1	76 %	81 %	81 %
W1	**93** %	**93** %	**93** %
W2	76 %	64 %	55 %
M3	88 %	90 %	62 %
M4	79 %	69 %	69 %
M5	74 %	62 %	40 %
W3	83 %	83 %	79 %
Mean	**81** %	77 %	68 %
Age 50-90			
W4	83 %	81 %	**86** %
W5	60 %	57 %	40 %
W6	48 %	55 %	57 %
W9	55 %	55 %	50 %
Mean	**62** %	**62** %	58 %
Age 90+			
W7	**71** %	67 %	69 %
W8	45 %	38 %	43 %
W10	57 %	43 %	48 %
Mean	**58** %	49 %	53 %
General Average	**66** %	63 %	60 %

to the system, we conducted some tests in a retirement home. The following participants took part in the testing:

- 7 persons aged between 20 and 50 years
- 4 persons from 50-90 years
- 3 persons older than 90 years

A fourth person in the 90+ group decided not to participate.
The evaluation helped to answer several questions. Our first interest was to analyze the recognition quality of the gesture recognition algorithm. For this we evaluated the recorded gesture information with all the implemented gesture recognition algorithms.
Furthermore, we observed how users from different age groups dealt with gesture control. Certainly the number of participants is too low to assure statistical significance but we get a first insight how even elderly handle gesture controlled applications.

Test scenario

For the test scenario we used the previously mentioned gesture controlled calculator. Every participant first trained the system with his own gestures for the different digits and operators. For this every gesture was recorded three times. Most of the participants used figures that were similar to that of drawing the number/operator on the black board.
After the system was trained the participants had to solve three different arithmetic problems which were read out loud by the test leader. The users were not informed whether or not a gesture was correctly recognized, in order to avoid that this would have an effect on how they realized the specific gestures. All gestures were recorded and later evaluated with the different recognition methods.

Recognition results

Table 1 shows an overview over the evaluation results. Support Vector Machines (SVM), Neuronal Networks (NN) and the Dynamic Time Warp algorithms (DTW) were used for the precision tests. We observe that the modern machine learning algorithms have an advantage over the dynamic time warp. Results were especially striking for younger people who reach an average precision of 81 % while people who are ninety years and above still achieve an accuracy of more than half of the gestures being recognized correctly. When examining this result we should take into account that the participants only had a relatively short training phase to get used to the gesture interaction. We suspect that after a longer learning phase, the performance of the gestures and thus the precision would improve for this group as well.

A closer look to the confusion matrix in figure 5 reveals that the most of the mistakes were made with gestures which are very similar in their movement paths. For example, mistakes often occurred between zero and six (0--6) or

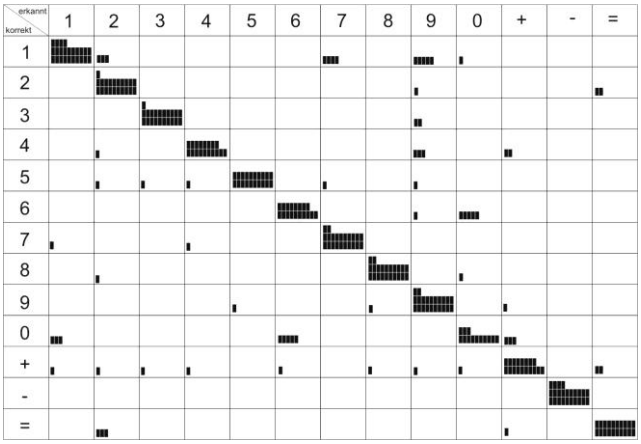

Figure 6: Confusion matrix. Lines contain the executed gestures, columns the recognized. Every rectangle is one gesture input.

one and seven (1—7). For practical use this can be avoided by defining sufficiently different gestures. One example for this is the pre-given alphabet of handwriting recognition on mobile devices with touchscreen.

Conclusion & Outlook

We introduced TaKG, a toolkit for automatic classification of gestures, and showed how it is integrated as a new input modality into a multimodal dialogue system implemented using the ODP framework. The classifier can be trained with tagged 3D-gestures which are recorded by a WiiMote. New gestures are classified by their gesture tag, which serves as input for the dialogue system and can be processed with other input modalities. An example application shows how to use gesture recognition in combination with speech synthesis. A small spoken dialogue model is used to guide the user for training the classifier and then switch to recognition mode.

Evaluation showed that even elderly persons are able to effectively use the WiiMote as an input modality and that the recognition results are very precise, although the participants did not have a very long training phase.

In the future, we will integrate the gesture modality into system aiming at scenarios in everyday life. Relevant applications include consumer electronic equipment, e.g. TV or media player. Here, a quick move to the right could switch to the next song or channel, a move to the left to the previous one. A direct channel access could be realized by writing the number of the channel into the air like in the calculator example. Turning the hand influences the volume and a fast slash could mute the sound. Since the system allow the user to train the system with his/her own gestures, new gestures can be introduced for specific music genres etc. Performing a gesture would create a play list which only contains songs of the genre the user related to his gesture. Furthermore we want to combine deictic and symbolic gestures. For this we take advantage of the Wii-

Mote IR sensors. This allows us perform gestures relative to an object presented on a monitor or objects in a room.

A further research interest is to move away from the Wii-Mote and to use the signal classifier for other input devices with different sensors. A follow up project deals with a highly personalized dialogue system, especially for disabled persons. Here it is important to support various different devices and sensors which are adapted to the abilities of a single person. The gesture classification algorithms introduced in this paper are independent from a special device and can be used to indentify signals from diverse sensors, giving the dialogue system flexibility in its input modalities.

Literature

[1] *Intuitive Interaction for Everyone with Home Appliances based on Industry Standards* (i2home). http://www.i2home.org/

[2] J. Alexandersson, G. Zimmermann, J. Bund. *User Interfaces for AAL: How Can I Satisfy All Users?*. In Proceedings of Ambient Assisted Living - AAL. 2. Deutscher Kongress mit Ausstellung/Technologien - Anwendungen - Management, pp. 5-9, Berlin, Germany, 2009.

[3] A. Pfalzgraf, N. Pfleger, J. Schehl, J. Steigner. *ODP - Ontology-based Dialogue Platform*. Technical Report, 2008, SemVox GmbH.
http://www.semvox.de/ whitepapers/odp_whitepaper.pdf

[4] J. Schehl, A. Pfalzgraf, N. Pfleger, J. Steigner. *The BabbleTunes System - Talk to your iPod!* To appear in: Proc. of the 10th International Conference on Multimodal Interfaces (ICMI 2008), pp 77-80, Crete, Greece.

[5] I. Laptev, T. Lindeberg. *Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features*. Technical report, 2001, Department of Numerical Analysis and Computer Science, KTH (Royal Institute of Technology), Stockholm, Sweden.

[6] D. Wilsona, A.Wilson. *Gesture Recognition Using the Xwand*. Technical report, Assistive Intelligent Environments Group Robotics Institute Carnegie Mellon University and Microsoft Research.
http://www.cs.cmu.edu/~dwilson/papers.xwand.pdf

[7] A. Y. Bensabat, J.A. Paradiso, *An Inertial Measurement Framework for Gesture Recognition and Applications*, 2001, In Gesture Workshop, pp 9-20.

[8] V. M. Mäntylä, J. Mäntyjärvi, T. Seppänen. *Hand Gesture Recognition of a Mobile Device User*. In Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on, pp 281-284.

[9] B. Signer, U. Kurmann, M. C. Norrie. *iGesture: A General Gesture Recognition Framework.* In Proceedings of ICDAR 2007, 9th International Conference on Document Analysis and Recognition, 954-958, Curitiba, Sept.2007.

[10] Mouse Gesture Plugin*, Optimoz Team*
http://optimoz.mozdev.org/gestures/index.html

[11] G.A. ten Holt, M.J.T. Reinders, E.A. Hendriks. *Multi-Dimensional Dynamic Time Warping for Gesture Recognition*. Thirteenth annual conference of the Advanced School for Computing and Imaging, 2007.

[12] G. Holmes, A. Donkin, I. H. Witten. *Weka: A Machine Learning Workbench*, Technical report, Department of Computer Science, University of Waikato, 1994.

[13] R. Neßelrath. *TaKG: A toolkit for automatic classification of gestures*, Masterthesis, Saarland University, 2008