A Brief Overview of Hand Gestures used in Wearable Human Computer Interfaces[1]

Technical report: CVMT 03-02, ISSN: 1601-3646

Thomas B. Moeslund and Lau Nørgaard
Laboratory of Computer Vision and Media Technology
Aalborg University, Denmark
E-mail: tbm@cvmt.dk

Abstract

This technical report provides a brief overview of how human hand gestures can be used in wearable Human Computer Interfaces (HCI).
This report is divided into two parts, one dealing with the different technologies that can be used to detect/recognise hand gestures in wearable HCI systems and another part focusing on one particular technology, namely the computer vision-based technology.
The first part describes a number of wearable HCI systems that use different technologies in order to include gestures into their interface. Furthermore, it also compares the different technologies with respect to certain general HCI aspects.
The second part describes a three-class taxonomy for the computer vision-based technology and gives examples within each of the classes.

[1]This research is in part funded by the ARTHUR project under the European Commissions IST program (IST-2000-28559). This support is gratefully acknowledged.

Chapter 1

Introduction

The purpose of this technical report is to provide a brief overview of how human hand gestures can be used in wearable Human Computer Interfaces (HCI).

The context of the report is an EC-project that the laboratory of Computer Vision and Media Technology (CVMT) is a part of. The project is know as the ”ARTHUR” project which is short for: ”Augmented Round Table for Architecture and Urban Planning”. The purpose of this project is to bridge the gap between real and virtual worlds by enhancing the users’ current working environment with virtual 3D objects.

In the ARTHUR project CVMT is responsible for all computer vision-based algorithms. These include: tracking and recognizing placeholder objects, head tracking, and gesture recognition. All algorithms are based on input from head mounted cameras.

During the fall of 2002 and the spring of 2003 a master student, Lau Nørgaard, was working on gesture recognition in the context of the ARTHUR project, i.e., via head mounted cameras. His work is documented in the thesis: ”Probabilistic Hand Tracking for Wearable Gesture Interfaces” [24]. The main text for this technical report is taken from his thesis.

This report is divided into two parts, one dealing with the different technologies that can be used to detect/recognise hand gestures in wearable HCI systems and another part focusing on one particular technology, namely the computer vision based technology.

Chapter 2

Different Technologies used for Including Gestures into Wearable HCI

In this chapter we describe a number of wearable HCI systems that use different technologies in order to include gestures into their interface. After this description we compare the different technologies with respect to certain general HCI aspects.

2.1 Summaries of Different Systems

The "Tinmith-Hand" [29] is a glove based gesture interface system for augmented and virtual reality. The system provides two types of interaction. The first is a menu system where each menu item is assigned to a finger, the item is selected by touching conducting pads on the matching fingertip with a pad on the thumb. The second mode of interaction is for manipulation of virtual 3D objects. This is done by computer vision based tracking of markers on both hands in six degrees of freedom. The main application for the Tinmith-hand is outdoor augmented reality for 3D modeling of architecture.

Several existing virtual reality data gloves are evaluated in [36] and criticized for being too expensive and bulky for widespread mobile use. A lightweight input device is described using only one bend sensor on the index finger, an acceleration sensor on the hand and a micro switch for activation. User test indicates that the developed interface, with use of simple gestures for controlling information appliances, is very intuitive and easy to learn. However several test persons complained about the necessary cables and other physical properties of the input device.

A wireless finger tracker is presented in [7]. An ultrasonic emitter is worn on index finger and the receiver, capable of tracking the position of the emitter in 3D, is mounted on the HMD. This provides excellent head relative tracking of the finger with an expected resolution of 0.5mm at a distance of 400mm from the HMD.

To avoid placing sensors on the hand and fingers the “GestureWrist” uses capacitive sensors on a wristband to determine the configuration of the fingers [31]. This is done by measuring the cross sectional shape of the wrist and use the bulges and cavities made by the sinews moving under the skin. The GestureWrist is sensitive to the positioning of the sensors on the wrist and has only been used to differentiate between two gestures (fist and point). On the other hand all necessary sensors, including an acceleration sensor, can be mounted in a normal wristwatch and are therefore highly unobtrusive. Furthermore the article proposes the GestureWrist combined with wireless on-body networking to avoid any cables going to the wristband.

Active infrared imaging can be used to simplify the task of separating hands and handheld objects from the background, and is therefore used in several wearable gesture interface systems [33][38]. A source of infrared illumination is mounted near to a camera fitted with an infrared-pass filter. If the light from this source is significantly stronger than the infrared part of the ambient illumination, the image captured by the camera will mainly be produced by light from the head mounted illuminant. As the intensity of the infrared light decrease with distance from the source, it is relatively simple to separate objects close to the head from objects farther away.

In [38] a HMD mounted active infrared setup is used for fingertip drawing and recognition of objects held in the hand. Object recognition is made possible by adding a color camera to the infrared camera on the HMD and using a beam splitter to give the two cameras the exact same image. Hereby the object can be extracted from the color image using the depth information from the infrared image.

The “Gesture Pendant” demonstrated in [33] is an active infrared camera in a necklace. It is a gesture interface primarily designed for home automation and as an aid for disabled and elderly people. Combining the active infrared principle with a fisheye lens an excellent image of the wearer’s hands is obtained. The Gesture recognition is performed with Hidden Markov Models based on the work done in mobile interpretation of sign language [35].

Multimodal interfaces are proposed combining the Gesture Pendant with voice recognition or tracking of the persons position within the home. Apart from control of the home and information appliances, the Gesture Pendant monitors the health of its wearer by observing pathological tremors in the gestures. The performance of the prototype system proved excellent in preliminary experiments. However as with many other vision based systems, processing power and battery life issues have to be solved before mobile use is possible.

Wearable computer vision based gesture recognition and hand tracking not using active infrared and not requiring the hand to be marked are presented in [14], [10] and [43].

[14] combine shape and skin color based tracking to create a mouse like interface with distinct point- and click gestures. Three applications are presented using the implemented mouse: A universal remote control for electronic appliances as TVs and stereos. Secure password input for augmented reality. A real world OCR translator capable of selecting signs and other text by pointing and then translating it from Japanese to English. To provide the necessary computing power, the system distributes the processing of the hand tracking algorithm between a wearable computer and a remote host through a wireless LAN. By performing some of the processing locally the wearable can provide faster response times and robustness to instabilities in the LAN due to roaming or interference.

[43] presents a finger menu similar to the "Tinmith-Hand" [29] though the interface is vision based and needs a clear view of the hand and outstretched fingers. A menu item is selected by simply bending the corresponding finger. Detection of the hand and fingers is accomplished by pixel level color segmentation using an adaptive color model to cope with variations in skin color and changes in lighting.

Not specifically an interface modality in itself, contextual awareness is a major area of research within the field of wearable computers [34]. The idea is to make the computer observe the user and his surroundings. This knowledge will enable intelligent interfaces that adapt to the user's immediate needs. For example the computer will know not to interrupt with phone calls or email in the middle of a meeting unless there is an emergency. On the other hand while driving a car alone the system can notify the user of incoming emails by a spoken summary. A long term goal is to have the computer model the user and his surroundings. Predictions based on this model will enable the system to retrieve information before it is needed and have it ready, when the user asks for it.

In [34] computer vision is used to obtain contextual awareness. Two cameras are mounted on the users head, one pointing forward at the surroundings and one looking down at the hands and body of the wearer. Estimates of the users location and current actions are made using Hidden Markov Models.

Another example of contextual awareness is DyPERS (Dynamic Personal Enhanced Reality System) [12], where audio and video messages can be associated with images of real world objects. Whenever the head mounted camera sees a known object the computer plays back the associated media. Many possible applications are listed ranging from note taking at a visual arts gallery to an aid for persons with poor vision.

2.2 Comparing the Different Technologies

As can be seen from the previous section many different interface technologies and modalities have been tried for wearable computers. The importance of different characteristics vary greatly with the application, and what is seen as an advantage to some is a disadvantage to others. The general advantages and disadvantages of the most common interface technologies are summarized in table 2.1.

The interface technologies can be divided into two groups: Devices for data entry and devices for selection and free form input. In the desktop environment these are exemplified by the keyboard and mouse respectively.

The data input technologies in table 2.1 are all in a mature state of development, and do not need research at the technical level. The vision based solutions, on the other hand, still need much work, but could ultimately become very intuitive interface modalities.

The free form input technologies in table 2.1 all have significant drawbacks. Systems based on active infrared pose the smallest problems with regards to image processing. But as they do not work in direct sunlight, they are not applicable for wearable computers for both in- an outdoor use. Glove based solutions are considered unfit for general use in wearable computers, as they prevent the user from unrestricted use of the hands. Consequently vision based gesture interfaces not relying on active infrared are the most generally applicable solutions. However these also pose the largest technical challenges.

Interface modality	Hands free	On body attachments	Text input	Free form input	Ease of use	Power consumption	Processing requirements	Technological Maturity
Voice recognition	YES	-	Possible	NO	Verry good	Low	High	Production
Chording keyboard	NO	-	Good	NO[3]	Hard to learn	Very low	None	Production
Half QWERTY	NO	Forearm	Good	NO	Good	Very low	None	Production
Glove based gesture[1]	NO	Sensors and cables	Poor	YES	Good	Low	Low	Usable
Vision based gesture	NO[2]	-	Poor	YES	Good	High	Very high	Experimental
Active IR gesture	NO[2]	-	Poor	YES	Good	Very high	Very high	Experimental

Table 2.1: Strengths and weaknesses of common interface technologies.

Chapter 3

Computer Vision-based Gesture Recognition

In this chapter we briefly describe how computer vision-based gesture recognition can be applied in gesture recognition. We structure this description via a three class taxonomy. For more insight into this research field please refer to the surveys [16, 42, 13, 27, 40].

Gesture recognition systems in general can be divided into three main components: Image preprocessing, tracking, and gesture recognition. In individual systems some of these components may be merged or missing, but their basic functionality will normally be present:

1. **Image preprocessing:** The task of preparing the video frames for further analysis by suppressing noise, extracting important clues about the position of the hands and bringing these on symbolic form. This step is often referred to as feature extraction.

2. **Tracking:** On the basis of the preprocessing, the position and possibly other attributes of the hands must be tracked from frame to frame. This is done to distinguish a moving hand from the background and other moving objects, and to extract motion information for recognition of dynamic gestures.

3. **Gesture recognition:** Based on the collected position, motion and pose clues, it must be decided if the user is performing a meaningful gesture.

The knowledge about the hands used for the tracking and recognition can exist on different levels of abstraction. Two main approaches exist in this regard differentiated by whether the system is based on an abstract model of the hand or on knowledge of the appearance of the hand in the image:

1. **Model based approach:** A model of the hand is created. This model is matched to the results of the preprocessing to determine the state of the tracked hand. The model can be more or less elaborate, from the 3D model with 27 degrees of freedom (DOF) used in the DigitEyes system [30] over the cardboard model used in [18] to a contour model of the hand seen straight on [21]. In addition to the model of the hand a model, of how features in the image corresponding to the real hand are produced, is required. This measurement model is needed in order to determine the state of the hand model from the appearance of the hand in the image.

   Continuously fitting the model to the hand in the video frames, is a process of tracking the complete state of the hand not just its position. This process is consequently called state based tracking. If the model contains a sufficient number of internal degrees of freedom, recognition of static gestures can be reduced to inspection of the state.

2. **Appearance based approach:** The tracking is based on a representation learned from a large number of training images. As no explicit model of the hand exists all the internal degrees of freedom will not have to be specifically modeled.

   When only the appearance of the hand in the video frames is known, differentiating between gestures is not as straight forward as with the model based approach. The gesture recognition will therefore typically involve some sort of statistical classifier based on a set of features that represent the hand, see e.g., [1].

3.1 Gesture based interfaces

A large part of the literature on gesture recognition deals with recognizing sets of dynamic gestures either as individual commands to a computer or with the ultimate goal of understanding sign language. An example of the latter is [35] which proposes to recognise sign language for both desktop and wearable computers. The obtained results were actually better with a head mounted downward looking camera than with a static desk based camera, as the head mounted camera is insensitive to body posture. The recognition is based on skin color segmentation to extract the position, shape, motion and orientation of the hands. The hands are modeled as ellipses, and the system is able to obtain good performance without modeling individual fingers. Using Hidden Markov Models (HMM) continuous recognition of full sentences of sign language is accomplished, although the vocabulary is limited to forty words.

Appearance based recognition of static gestures is presented in [1], where letters from the hand alphabet are recognized by principal component analysis (PCA) and a Bayessian classifier. The appearance of the individual signs are learned from a large number of training images. The PCA is used to create a low dimensional feature space in which hands located in the video frames can be compared with classes representing the defined gestures. The classes and the corresponding classifier are created in an off line learning process. This is the principle of ”eigen-hands” inspired by ”eigen-faces”, which are used in face recognition (see e.g., [37]). The main problem with these appearance based models is that they are view-dependent and therefore require multiple views in the training, see e.g., [6].

In addition to the work on how to detect and recognize gestures, research is being done on designing intuitive and natural gesture sets [23], and on how gestures and body language are used as a part of inter person communication [3].

3.1.1 Hand or finger tracking

Below we briefly exemplify our three class taxonomy in more detail.

Image Preprocessing

Pixel level segmentation:

Regions of pixels corresponding to the hand are extracted by color segmentation or background subtraction. Then the detected regions are analyzed to determine the position and orientation of the hand. The color of human skin varies greatly between individuals and under changing illumination. Advanced segmentation algorithms, that can handle this, have been proposed [43][5], however these are computationally demanding and still sensitive to quickly changing or mixed lighting conditions. In addition color segmentation can be confused by objects in the background with a color similar to skin color. Background subtraction only works on a known or at least a static background, and consequently are not usable for mobile or wearable use. Alternatives are to use markers on the fingers [39] or use infrared lighting to enhance the skin objects in the image, see e.g., [26].

Motion segmentation:
Moving objects in the video stream can be detected by calculation of inter frame differences and optical flow. In [41] a system capable of tracking moving objects on a moving background with a hand held camera is presented. However, such a system can not detect a stationary hand or determine which of several moving objects is the hand.

Contour detection:
Much information can be obtained by just extracting the contours of objects in the image [11]. The contour represent the shape of the hand and is therefore not directly dependent on skin color and lighting conditions. Extracting contours by edge detection will result in a large number of edges both from the tracked hand and from the background. Therefore some form of intelligent post processing is needed to make a reliable system.

Correlation:
A hand or fingertip can be sought in a frame by comparing areas of the frame with a template image of the hand or fingertip [4] [25]. To determine where the target is, the template must be translated over some region of interest and correlated with the neighborhood of every pixel. The pixel resulting in the highest correlation is selected as the position of the target object. Apart from being very computationally expensive template matching can not cope with neither scaling nor rotation of the target object. This problem can be addressed by continuously updating the template [4], with the risk of ending up tracking something other than the hand.

Tracking

On top of most of the low level processing methods a tracking layer is needed to identify hands and follow these from frame to frame. Depending on the nature of the low level feature extraction, this can be done by directly tracking one prominent feature or by inferring the motion and position of the hand from the entire feature set.

Tracking with the Kalman filter:
One way of solving the problem of tracking the movement of an object from frame to frame is by use of a Kalman filter. The Kalman filter models the dynamic properties of the tracked object as well as the uncertainties of both the dynamic model and the low level measurements. Consequently the output of the filter is

a probability distribution representing both the knowledge and uncertainty about the state of the object. The estimate of the uncertainty can be used to select the size of the search area in which to look for the object in the next frame.

The Kalman filter is an elegant solution and easily computable in real time. However, the probability distribution of the state of the object is assumed Gaussian. As this is generally not the case, especially not in the presence of background clutter, the Kalman filter in its basic form can not robustly handle real world tracking tasks on an unknown background [11] [19]. However on a controlled background good results can be obtained [30].

CONDENSATION:

An attempt to avoid the limiting assumption of normal distribution inherent in the Kalman filter was introduced in [11] and denoted the CONDENSATION algorithm. The approach is to model the probability distribution with a set of random particles and perform all the involved calculations on this particle set. The group of methods, to which the CONDENSATION algorithm belongs, are generally refereed to as: Random sampling methods, sequential Monte Carlo methods or particle filters.

Very promising results have been obtained using random sampling in a variety of applications on complex backgrounds. [8] and [9] propose a combination of appearance based eigen tracking [2] and CONDENSATION for gesture recognition. Sequential Monte Carlo methods and adaptive color models are used in [28] providing robust tracking of objects undergoing dramatic changes in shape. To the task of face and hand tracking, motion clues are combined with the color information, to eliminate stationary skin colored objects like wooden doors and desks. [22] use skin color segmentation, region growing and CONDENSATION for simultaneous tracking of both hands. Solutions to handle occlusions are proposed resulting in reliable operation even when the blobs corresponding to the hands merge for extended periods.

In [15] a hand model consisting of blobs and ridges of different scales representing the palm, fingers and fingertips is used with particle filtering to track the position of the hand and the configuration of the fingers. Real time performance is obtained but the model and state space is limited to 2D translation, planar rotation, scaling and the number of outstretched fingers.

[21] propose a method, called Partitioned Sampling, for tracking articulated objects with particle filters without requiring an unreasonable amount of particles to cope with the resulting high dimensional state space. The solution is to first locate the base of the object and then determine the configuration of attached

links in a hierarchical way. As an example of this, a hand drawing application is presented. The partitioned sampling is used by first locating the palm and subsequently determining the angles between the palm and the thumb and index finger. These angles are used to differentiate between a small number of gestures corresponding to drawing commands. Tracking is based on a spline description of the contour of the hand being fitted to edges in the image and combined with skin color matching as presented in [19] and [20]. Detailed motion models and background subtraction is used to limit the effect of clutter.

Recognition

Usually the classical algorithms from the field of pattern recognition are applied. These are Hidden Markov Models, correlation, and Neural Networks. Especially the two first have been used with success while the Neural Networks often have the problem of modelling non-gestural patterns [17]. For more inside into how Hidden Markov Models and correlation can be applied in gesture recognition see, e.g., [32][17][35] and [4][1][6], respectively.

Chapter 4

Discussion

In this report we have done two things. Firstly, we have tried to give a brief overview over the different technologies which are applied in wearable HCI in order to include gestures. Secondly, we have tried to give a brief overview of computer vision-based gesture recognition.

We were surprised to see the great variety of different technologies used for including gestures into HCI. One reason might be the fact that the current state of the art in computer vision-based gesture recognition is not impressive.

Besides getting a good overview, reading through the literature also allowed us to identify desirable characteristics of a technology used in wearable HCI. These are listed below.

1. **Robust initialization and reinitialization:** The hand can be expected to enter and exit from the view frequently. Therefore the tracker must be able to quickly reinitialize itself, and a reliable estimation of whether the hand is present or not must be obtainable.

2. **Robustness to background clutter:** Objects in the background should not distract the tracker, not even if these objects are of skin color.

3. **Independence of illumination:** As the tracker is to be used in wearable applications, it must be able to cope with changing and mixed lighting conditions.

4. **Computationally effective:** Mobile processors tend to be significantly less powerful than their desktop counterparts. Algorithms requiring extensive computational resources should therefore be avoided.

Bibliography

[1] Henrik Birk, Thomas B. Moeslund, and Claus B. Madsen. Realtime recognition of hand alphabet gestures using principal component analysis. In *10th Scandinavian Conference on Image Analysis*, Lappeenranta, Finland, 1997.

[2] Michael J. Black and Allan D. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision*, 26(1):63–84, 1998.

[3] Justine Cassell. A framework for gesture generation and interpretation. In R. Cipolla and A. Pentland, editors, *Computer Vision in Human-Machine Interaction*, pages 191–215. Cambridge University Press, New York, 1998.

[4] James L. Crowley, Franois Berard, and Joelle Coutaz. Finger tracking as an input device for augmented reality. In *International Workshop on Gesture and Face Recognition*, Zurich, Switzerland, 1995.

[5] Sylvia M. Dominguez, Trish Keaton, and Ali H. Sayed. Robust finger tracking for wearable computer interfacing. In *Workshop on Perspective User Interfaces*, Orlando, FL, 2001.

[6] H. Fillbrandt, S. Akyol, and K.F. Kraiss. Extraction of 3D Hand Shape and Posture from Image Sequences for Sign Language Recognition. In *International Workshop on Analysis and Modeling of Faces and Gestures*, Nice, France, 17 October 2003.

[7] Eric Foxlin and Michael Harrington. Weartrack: A self-referenced head and hand tracker for wearable computers and portable vr. In *International Symposium on Wearable Computing*, pages 155–162, Atlanta, GA, 2000.

[8] Namita Gupta, Pooja Mittal, Sumantra Dutta Roy, Santanu Chaudhury, and Subhashis Banerjee. Condensation-based predictive eigentracking. In

Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP2002), 2002.

[9] Namita Gupta, Pooja Mittal, Sumantra Dutta Roy, Santanu Chaudhury, and Subhashis Banerjee. Developing a gesture-based interface. *IETE Journal of Research: Special Issue on Visual Media Processing*, 48(3):237–244, 2002.

[10] Giancarlo Iannizzotto, Massimo Villari, and Lorenzo Vita. Hand tracking for human-computer interaction with graylevel visualglove: Turning back to the simple way. In *Workshop on Perceptive User Interfaces*. ACM Digital Library, November 2001. ISBN 1-58113-448-7.

[11] Michael Isard and Andrew Blake. Contour tracking by stochastic propagation of conditional density. In *ECCV (1)*, pages 343–356, 1996.

[12] Tony Jebara, Bernt Schiele, Nuria Oliver, and Alex Pentland. Dypers: Dynamic personal enhanced reality system. In *Perceptual Computing Technical Report #463*. MIT, 1998.

[13] M. Kohler and S. Schroter. A Survey of Video-based Gesture Recognition - Stereo and Mono Systems. Technical Report Research Report Nr. 693, Fachbereich Informatik, University of Dortmund, 1998.

[14] Takeshi Kurata, Takekazu Kato, Masakatsu Kourogi, Jung Keechul, and Ken Endo. A functionally-distributed hand tracking method for wearable visual interfaces and its applications. In *IAPR Workshop on Machine Vision Applications*, pages 84–89, Nara, Japan, 2002.

[15] Ivan Laptev and Tony Lindeberg. Tracking of multi-state hand models using particle filtering and a hierarchy of multi-scale image features. In M. Kerckhove, editor, *Scale-Space'01*, volume 2106 of *Lecture Notes in Computer Science*, pages 63–74. Springer, 2001.

[16] J.J. LaViola. A Survey of Hand Posture and Gesture Recognition Techniques and Technology. Technical Report CS-99-11, Department oc Computer Science, Brown University, Providence, Rhode Island, 1999.

[17] H.K. Lee and J.H. Kim. An HMM-based Threshold Model Approach for Gesture Recognition. *Transactions on Pattern Analysis and Machine Intelligence*, 21(10):961–972, 1999.

[18] J. Lin, Y. Wu, and T.S. Huang. Capturing human hand motion in image sequences. In *Workshop on Motion and Video Computing*, Orlando, Florida, December 5-6 2002.

[19] John MacCormick. *Stochastic Algorithms for Visual Tracking: Probabilistic Modelling and Stochastic Algorithms for Visual Localisation and Tracking*. Springer-Verlag New York, Inc., 2002.

[20] John MacCormick and Andrew Blake. A probabilistic contour discriminant for object localisation. *Proc. Int. Conf. Computer Vision*, 1998.

[21] John MacCormick and Michael Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *European Conf. Computer Vision*, volume 2, pages 3–19, 2000.

[22] James P. Mammen, Subhasis Chaudhuri, and Tushar Agarwal. Simultaneous tracking of both hands by estimation of erroneous observations. In *British Machine Vision Conference (BMVC)*, Manchester, UK, 2001.

[23] Michael Nielsen, Thomas Moeslund, Moritz Storring, and Erik Granum. A procedure for developing intuitive and ergonomic gesture interfaces for hci. In *The 5th Int. Workshop on Gesture and Sign Language based Human-Computer Interaction*, Genova, Italy, 15-17 April 2003.

[24] Lau Nørgaard. Probabilistic Hand Tracking for Wearable Gesture Interfaces. Master's thesis, Laboratory of Computer Vision and Media Technology, Aalborg University, Denmark, 2003.

[25] Rochelle O'Hagan and Alexander Zelinsky. Finger track - a robust and real-time gesture interface. In *Australian Joint Conference on Artificial Intelligence*, Perth, Australia, 1997.

[26] K. Oka, Y. Sato, and H. Koike. Real-Time Tracking of Multiple Fingertips and Gesture Recognition for Augmented Desk Interface Systems. In *International Conference on Automatic Face and Gesture Recognition*, Washington D.C., USA, May 20-21 2002.

[27] V.I. Pavlovic, R. Sharma, and T.S. Huang. Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review. *Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.

[28] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *European Conference on Computer Vision*, volume 1, pages 661–675, Copenhagen, Denmark, 2002.

[29] Wayne Piekarski and Bruce H. Thomas. The tinmith system: demonstrating new techniques for mobile augmented reality modelling. In *Third Australasian conference on User interfaces*, pages 61–70. Australian Computer Society, Inc., 2002.

[30] James M. Rehg and Takeo Kanade. Digiteyes: Vision-based hand tracking for human-computer interaction. In *Workshop on Motion of Non-Rigid and Articulated Bodies*, pages 16–24, 1994.

[31] Jun Rekimoto. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *International Symposium on Wearable Computing*, pages 21–27, 2001.

[32] G. Rigoll, A. Kosmala, and S. Eickeler. High Performance Real-Time Gesture Recognition Using Hidden Markov Models. Technical report, Gerhard-Mercator-University Duisburg, 1998.

[33] Thad Starner, Jake Auxier, Daniel Ashbrook, and Maribeth Gandy. The gesture pendant: A self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *International Symposium on Wearable Computing*, Atlanta, GA, October 2000.

[34] Thad Starner, Bernt Schiele, and Alex Pentland. Visual contextual awareness in wearable computing. In *Second International Symposium on Wearable Computers*, pages 50–57, 1998.

[35] Thad Starner, Joshua Weaver, and Alex Pentland. 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, 1998.

[36] Koji Tsukada and Michiaki Yasumura. Ubi-finger: Gesture input device for mobile use. *Information Processing Society of Japan Journal*, 43(12), 2002.

[37] M. Turk and A. Pentland. Eigen Faces for Recognition. *Cognitive Neuro-Science*, 3(1), 1991.

[38] Norimichi Ukita, Yasuyuki Kono, and Masatsugu Kidode. Wearable vision interfaces: towards wearable information playing in daily life. In *1st CREST Workshop on Advanced Computing and Communicating Techniques for Wearable Information Playing*, pages 47–56, 2002.

[39] K.D. Ulhaas and D. Schmalstieg. Finger Tracking for Interaction in Augmented Environments. In *International Symposium on Augmented Reality*, New York, New York, 29-30 October 2001 2001.

[40] R. Watson. A Survey of Gesture Recognition Techniques. Technical Report TCD-CS-93-11, Department of Computer Science, Trinity College, Dublin, Irland, 1993.

[41] King Yuen Wong and Minas E. Spetsakis. Motion segmentation and tracking. In *15th International Conference on Vision Interface*, pages 80–88, Calgary, Canada, 2002.

[42] Y. Wu and T.S. Huang. Vision-based Gesture Recognition: A Review. In A. Braffort et al., editor, *International Workshop*, number 1739 in LNAI. Springer, 1999.

[43] Xiaojin Zhu, Jie Yang, and Alex Waibel. Segmenting hands of arbitrary color. In *International Conference on Automatic Face and Gesture Recognition*, pages 446 – 453, Grenoble, France, 2000. IEEE Computer Society.