Alternative User Interfaces

Johannes Sorger*
Vienna University of Technology

Abstract

A modern user interface should not only enable the user to handle the functions and take full advantage of a particular device, it also should provide control in the most simple and intuitive way. Because conventional user interfaces cannot guarantee to accomplish this task for every application, alternative user interfaces are being researched and developed. The possibilities for their application are vast, as is the amount of different techniques they are implemented in. The focus of this paper lies on motion tracking, as it is an important factor in the research and development of alternative user interfaces. An overview of the different motion tracking techniques is given, going into detail about optical and inertial tracking. As an alternative to motion tracking, biosignal sensors are introduced. Also, some applications of alternative user interfaces are presented, demonstrating how many tasks in different aspects of life can benefit from these techniques.

Keywords: unconventional human computer interfaces, optical tracking, inertial tracking, interaction, biosignal sensors

1 Introduction

The term user interface is defined as the aggregate of means by which people (the users) interact with a particular machine, device, computer program or other complex tool (the system). The user interface provides means of input, allowing the user to manipulate the system - and output, allowing the system to react to the user's manipulation in a desired manner. It represents a layer between the human who is operating the machine or system and the actual machine or system itself. User interfaces are not a phenomenon of our recent culture - they arose as soon as mankind began to design primitive apparatuses and have evolved since then. Machines grew in complexity and variation, making user interface design an increasingly challenging task. A modern user interface should not only enable the user to handle the functions and take full advantage of a particular device, it also should provide control in the most simple and intuitive way. The ease of control is an important factor for commercial success on the consumer market. To meet these criteria, particularly in the IT sector many user interfaces have been standardized. Take the mouse and keyboard as the most popular example. There are few other ways in which one can interact with a digital device.

Of course the spectrum of usages of electronic devices in our everyday world grows broader and not every situation is handled optimally via keyboard or button input. Many tasks require another approach, an alternative user interface. Generally an alternative user interface can be described as every means of interaction between a human and a system that is not executed via a conventional and commercially available interface. The range of this topic is wide and could never be covered entirely in a paper of this extent. Therefore the focus of this paper lies on a few selected approaches.

When researching about alternative user interfaces one will inevitably stumble over the term ”tracking”. Tracking in connection with user interfaces could mean a lot of things. The tracking of the user's keyboard input or mouse movement for example. Generally every recognition or monitoring of user actions falls under this term. You can even track the user's heartbeat and use it as input for a system, as we will see later. Normally, though, in this branch of science ”tracking” refers to the tracking of user movements. This tracking is achieved through different kinds of sensors or mechanical installations that register the user's movement in two or three dimensions. They are therefore logical choices for user interfaces in virtual reality (VR) or augmented reality (AR) applications. A list of different tracking technologies can be seen in Figure 1. All of these techniques have advantages and drawbacks caused by their principles of work. Most of them (exceptions will be explained later) can be deployed in two modes - outside-in- and inside-out-tracking - dependent on the constellation of their sensors. In outside-in tracking, the sensors are placed on fixed locations in the environment, pointed towards the object being tracked. These fixed sensors form their own coordinate system and register changes within this system. In inside-out tracking the sensors are positioned directly on the object that is being tracked, faced outward in order to capture certain features in the environment.

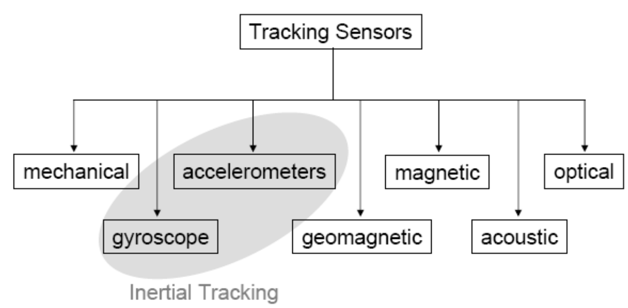

Figure 1: Most common tracking sensors.

Another distinction that can be made between different tracking systems is in how many degrees of freedom (DOF) the user is allowed to interact with them. The degrees of freedom that are supported by a system give information about the directions in which a body can be moved or tilted / turned, respectively the movements that the system is able to recognize. In general, a rigid body in d-dimensions has $d(d+1)/2$ degrees of freedom (d translations $+d(d-1)/2$ rotations) which results in a maximum of six DOF for a body in three dimensional space. Of course this can differ in practice, depending on the requirements of the user interface and technical limitations. A mouse, for example, can only be operated in two degrees of freedom while most tracking devices support three or more DOF. The six DOF in three dimensions are often referred to as:

*e-mail: johannessorger@gmx.net

1. Heaving: moving up and down
2. Swaying: moving left and right
3. Surging: moving forward and backward
4. Pitching: tilting up and down
5. Yawing: turning left and right
6. Rolling: tilting side to side

In the following sections of this paper different tracking techniques are introduced. Chapter two will give an overview of tracking techniques that are related to the techniques explained in chapter three and four. Chapter three will give a detailed insight into optical tracking, regarding its technical background and discussing two applications of optical tracking. In chapter four inertial tracking will be described, giving as well some information about its technical background and explaining an example of its application. In chapter five a different approach than tracking will be introduced: biosignal sensors as alternative user interfaces.

2 Related Work

What follows is an overview of different tracking sensors that are related to the techniques explained in the next two chapters as they are used for 2D or 3D pose estimation, as well. Their basic setup, functions, advantages and disadvantages will be explained according to Dorfmüller-Ulhaas [Dorfmueller-Ulhaas 2002]. Each of these techniques has their own advantages and drawbacks compared to the others. A flawless tracker does not exist but there are approaches called hybrid trackers that try to overcome the disadvantages of one technique by combining it with another in order to get an optimal solution for a specific tracking task.

2.1 Mechanical Sensors

Mechanical sensors require a physical connection to the object that is supposed to be tracked. These sensors can be compared to a robotic arm as they consist of a jointed structure with rigid links. Any movement of this object is precisely tracked by potentiometers or mechanical encoders. The advantages of this system are extremely good precision and fast response times provided through the mechanical sensors which also are not prone to jitter. The disadvantages on the other side arise through this physical connection which impairs the free movement of the user and also limits the area of operation.

2.2 Geomagnetic Sensors

Geomagnetic sensors are only able to measure an absolute orientation. The geomagnetic compass measures orientation corresponding to the earth's magnetic field which is cheap but not very accurate. In many environments it can easily be disturbed by ferrous metal. The gyrocompass works geomagnetically, too, but makes use of the earth's spin in order to align itself towards true north. This technique is more accurate than the geomagnetic compass and it works more robust. Currently the gyrocompass is too large for human motion tracking.

2.3 Magnetic Trackers (MTs)

Magnetic trackers generate magnetic fields by a source of three orthogonal coils of wire. These coils have to be activated in sequence in order to generate three orthogonal magnetic dipole fields that do not influence each other. There are two approaches for magnetic trackers, one using magnetic field coupling [Raab et al. 1979], the other using quasi DC fields [Blood 1989]. Both approaches require a special sensor to measure the magnetic field attenuation, the strength and also the direction of the magnetic field. There are many advantages to magnetic tracking as it allows several body parts to be tracked simultaneously. Further it is not sensitive to the line of sight problem, meaning it will still function correctly if the object that is supposed to be tracked is occluded by another object. On the downside MTs are inaccurate and they suffer from latency problems and distortion of data. They can be thrown off by large amounts of metal or other electromagnetic fields in their surroundings. Also MTs can only be used in a restricted area as the sensors may not be placed too distant from the source. Considering all these disadvantages, magnetic trackers are still a popular approach in a wide range of human-machine interface applications.

2.4 Acoustic Trackers

Acoustic trackers (also called ultrasonic trackers) use ultrasonic and consist of three high frequency sound wave emitters in a rigid constellation from the source. The three receivers that are worn by the user (also in a rigid arrangement) make it possible to determine the user's pose in six degrees of freedom. The emitters and receivers can also be placed the other way round to allow inside-out tracking. Position and orientation of the tracked object can be determined by two methods: phase coherence and time of flight (TOF) ranging. In phase coherence the range is determined by measuring the phase shift between the transmitted signal of a continuous wave source and the signal that is detected at the microphone. This method allows continuous measurement without latency but registers only relative distance changes. Also, the received signal is easily disturbed by reflections of the signal. Time of flight ranging on the other side measures the amount of time it takes for a sound to reach the sensors after it has been emitted by the transmitters. Ultrasonic trackers are widely used and also available in commercial products while being generally very inexpensive. The downside of acoustic tracking is that it also can only be used in a restricted working area and that it suffers from the line-of-sight problem as emitter and receiver may not be occluded. Also, acoustic trackers can be affected by temperature and pressure changes.

3 Optical Trackers (OTs)

The general scheme of optical trackers can be described as one or multiple optical sensors that, using an algorithm, recognize and track certain features or markers in an environment. Though it is important to note that there is a huge variety in hardware setups and tracking algorithms, as will be explained in the next section in more detail. Optical trackers provide an interesting alternative to magnetic trackers. While offering the same functionality they do not share the disadvantages of magnetic tracking systems. Optical trackers provide wireless interaction and high accuracy measurement. They can work in real time and have a reasonable update rate, plus they support tracking of multiple objects without the use of additional sensors. Beside all these benefits OTs are of course not flawless as they are very much dependent on the illumination of the environment and on scene constraints. Further they cannot

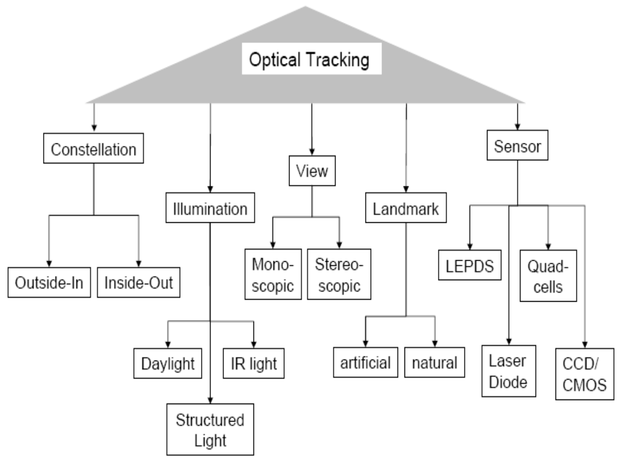

Figure 2: Categorization of optical trackers.

track objects that are concealed and they tend to be computationally expensive.

3.1 Technical Background

Referring to the above mentioned variety of hardware setups in optical tracking, according to Dorfmüller-Ulhaas [2002], the hardware of optical trackers can be categorized as shown in Figure 2. A general distinction of OTs is made in the constellation of their optical sensors: outside-in or inside-out. Outside-in trackers are used in cases that require the user to move freely in a non-obstructive environment, without being connected and hindered through wires. Especially if non-rigid objects or multiple non-occluding objects are tracked simultaneously, for example optical gesture recognition, hand-, head-, face- and human body tracking applications and motion capturing in general. Inside-out tracking is suited for applications that require only a few rigid objects to be tracked. For example, augmented reality applications where only the pose of the user's head - a single rigid object - needs to be tracked.

A further distinction between different OTs can be made in regard to the illumination type they are working with. As already mentioned, illumination is a critical problem in optical tracking applications. Many developers of optical tracking systems rely on daylight or natural illumination provided by the environment. But often the environment affects the illumination and due to these dynamic light sources, lighting cannot be assumed to be static, thus resulting in differing lighting conditions. To meet this challenge caused by the illumination circumstances and the dynamic sensitivity of camera sensors, specific image processing algorithms have to be applied [Naimark and Foxlin 2002].

Illumination in a spectrum of wavelengths higher (ultraviolet) or lower (infrared) than the human eye is able to perceive has the advantage of being independent of lighting conditions, for example in environments where projectors are being used. This is due to the fact that there is no noticeable interference in the light spectrum as long as there is no sunlight irradiating the scene.

When using structured light as source of illumination a known pattern of pixels (often grids or horizontal bars) is projected onto a scene. The way that these deform when hitting the surface allows the tracking system to calculate the depth and surface information of the objects in the scene.

Another distinction in optical tracking systems lies in the amount of sensors being used. One distinguishes between monoscopic and stereoscopic views. A stereoscopic system with two cameras is better suited for 3D motion tracking tasks, especially if the tracked object is not rigid and moving. If the two cameras are rigid and their calibration data are known, an estimation of the object's position in three dimensional space is available at any moment and the pose and structure of this object can be determined, if the correlation between images are known. Monoscopic systems, on the other side, require information about the structure of a rigid object in order to calculate its pose.

When designing an optical tracking system the developer has to decide whether the system should keep track of artificial or natural landmarks. When using artificial landmarks, the design of the marker is up to the developer. A marker is a point or structure that the system should be able to recognize in the environment and thus be able to register changes of position and orientation in relation to it. Since the properties of the fiducial are known to the system, it can be more easily detected than natural features whose structure and texture are defined by the environment. One way to circumvent the tracking difficulties that arise in the absence of known structure is to assume some features to be part of lines [Jiang and Neumann 2001] or within planes [Simon et al. 2000]. Another way would be the use of stereoscopic tracking because it is able to initialize object structure by triangulation which makes it unnecessary to assume a specific structure of scene parts.

Finally, one can distinguish optical tracking systems by the type of sensors they are using. There are lateral effect photo diodes (LEPDS), laser diodes, quad cells which are all non-imaging sensors, and CCD or CMOS sensors which are commonly used in photo and video cameras. Quad-cells and LEPDs are pure analog sensors that determine the centroid of light in the field of view. Quad-cells are simple optical direction sensors that operate without lenses and therefore do not have the disadvantage of the optical distortion caused by lenses. The drawback of quad-cells and LEPDs is that they are easily disturbed through reflections and other light sources in the environment as they cause the centroid to be shifted to their direction. Laser diodes use optical ranging techniques, similar to acoustic ranging. The laser beams' propagation time is used to calculate the distance from the laser diode to a reflecting target, which proves to be very accurate but also very expensive and only possible for single target tracking.

The accuracy of optical tracking systems is mainly dependent on the lens quality, the resolution of the sensors and the image size of the projected rigid object as opposed to the size of the object which does not directly affect accuracy.

3.2 Applications of Optical Traking

In this section of the paper examples of alternative user interfaces using optical tracking are given.

3.2.1 Tracking of Bare Fingers for Interactive Surfaces

Letessier and Berard [2004] developed an optical tracker for tracking of bare fingers, allowing multiple users to directly manipulate virtual objects with their hands. Their system is a monoscopic outside-in tracker using a normal off-the-shelf video camera (CCD sensors). Concretely the two dimensional position of fingertips on a planar display surface is being tracked.

The hardware of the system simply consists of a video camera that is connected via FireWire to a PC that does not necessarily require

Figure 3: The CED metric: a) background model with hand, b) difference map with the CED metric not thresholded, c) thresholded.

to have high end specs, a projector for displaying of the user interface and a planar surface that the interface is projected on. Compared to a simple mouse as input device which only requires the information about movement and button press and release, the finger as means of input also provides the event types "appear" and "disappear", as it can be hidden or outside the camera's field of view. Additionally, as the system supports tracking of multiple fingers, finger event parameters must include an ID of the finger that generated a certain event.

Letessier's and Berard's tracker falls into the category of pixel-oriented tracking methods, as opposed to model-oriented tracking methods. The latter methods aim at fitting a model of the finger on the frames coming from the video stream. These models vary from a very complex articulated skeleton of the hand, to parameterized contours and small images representing the finger. However, this approach tends to be very computationally expensive and not very user friendly [2004]. Pixel-oriented methods on the other side use pixel-level information to segment the frames. The result of this segmentation is a difference map where each pixel is associated to a class such as foreground and background. The drawback of these methods is that they are sensitive to camouflage since they are not able to segment contiguous objects of similar appearance.

Pixel-oriented methods are based on three main approaches: color models, region growing and image differencing. The latter has been implemented by Letessier and Berard [2004] because it provides a good resolution under various conditions and has been proven efficient for finger tracking [Hardenberg and Berard 1989]. The difference maps that are resulting from the image differencing segmentation (IDS) have to be post processed in order to extract finger positions. Letessier and Berard developed an original shape filtering algorithm that is optimized in processing and invariant to finger orientation. At first foreground extraction using IDS is applied to produce a grayscale difference map, next it is converted into a binary map through automatic thresholding. Then a shape filter extracts the fingertip positions and finally high-level events (such as motion) are generated through association for the client application.

Foreground extraction: The IDS model for foreground extraction is dependent on three factors: the form of the background model, the metric used to compare the background model and the current image, and finally the background maintenance process.

The background model is achieved by modeling each pixel variations. In this approach the background is modeled as a simple image where each pixel is the mean over time of the measured pixel values. This approach was taken because pixel variation due to camera- and lighting noise and background changes are determined as roughly the same.

As comparison metric a color based-approach is used. Luminance-based approaches have difficulties in making a distinction between the fingers and the shadows they cast on the background [Hardenberg and Berard 1989], thus registering the shadows as part of the fingers. Letessier and Berard propose their own purely color-based metric. This computationally inexpensive method calculates the Euclidian distance between corresponding pixels in the (r, g) normalized chromaticity plane and was named Chrominance Euclidian Distance (CED). Figure 3 b) shows a difference map with the CED metric that resulted from the frame shown in Figure 3 a). The similarity between pixels p and p' is computed as d(p, p') given by

$$p = [R, G, B] \tag{1}$$

$$r(p) = R/(R+G+B) \tag{2}$$

$$g(p) = G/(R+G+B) \tag{3}$$

$$d(p, p') = ||[r, g] - [r', g']|| \tag{4}$$

The process of adapting the segmentation to a changing background is called background maintenance. In this tracker the background model *Bg* is maintained by computing the average of recent pixel measurements which is approximated by a running average. If the current frame is *Im*, the background mode at time $t+1$ can be calculated with (5). The learning rate $\alpha^t_{(x,y)}$ is affected by the IDS results, for example small values for segmented objects and high values for the background.

$$Bg^{t+1}_{(x,y)} = \alpha^t_{(x,y)} \cdot Im^t_{(x,y)} + (1 - \alpha^t_{(x,y)}) \cdot Bg^t_{(x,y)} \tag{5}$$

The background displays the user interface that is projected onto it and changes therefore regulary. These dynamic changes of the background could cause false alarms in the difference map. A projected finger could be recognized as a real finger, for example. Because the visuals projected on a white surface are much brighter than the physical objects in the scene, this problem can be solved through augmentation of the camera gain which moves the projection's appearance out of the camera sensor range, as can be seen in Figure 4.

Automatic thresholding: A threshold is used to transform the obtained similarity map into a binary image, as seen in Figure 3 c). The threshold must be uniform, since luminance variations are being ignored by the segmentation method and it must be determined automatically, since the system is supposed to work autonomously. A typical histogram of the distance map contains two modes, the lower mode, corresponding to the background noise (roughly 80% of the pixels), and the higher mode, corresponding to the foreground. Ideally the threshold should be chosen to eliminate the first mode while preserving the other.

Shape filtering: Shape filtering for fingertip detection is the only method that can recognize theoretically any amount of fingers with

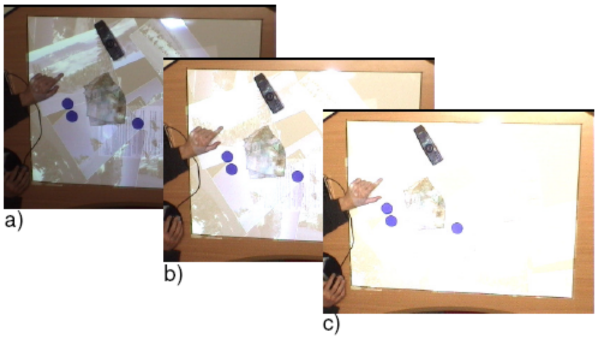

Figure 4: Effect of increased camera gain. Gain is set to ”auto” a), gain is increased b), until the projected image is overexposed c).

basically no performance loss because it processes each frame as a whole. The principle of shape filtering is to characterize the searched object with simple geometric criteria and verify these criteria for each pixel in the binary-image. The filter algorithm developed by Letessier and Berard [2004] rejects or accepts each pixel according to the set filter parameters and clusters the accepted pixels into groups of connected pixels, as can be seen in Figure 5. The output of the filter is a list of (x,y) coordinates of the medians of these vertical and horizontal clusters.

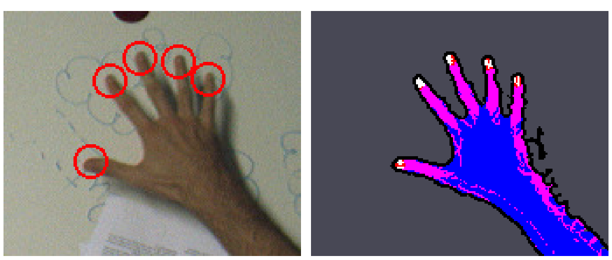

Figure 5: Output of the filter algorithm (right). Accepted pixels are white, their centers are represented by red circles on the left image.

Association: Finally the filter's output gets translated into events for the client application. The difficulty here lies in identifying the fingers consistently such that the same ID is provided with the appear-, motion-, and disappear-events for one particular finger. The ”closest neighbor algorithm” that was developed by Berard [Berard 2003] processes the data from the filter in the following way: each detected fingertip at the current frame t is matched to the closest memorized finger at the preceding frame. If no finger is memorized or the closest one is above a threshold distance, an appear event with a new finger ID is generated. A motion event occurs if a finger has moved farther than a given threshold distance. Memorized fingers that cannot be associated to any fingers from the current frame generate a disappear event for their ID and are deleted. As the tracker only operates in two degrees of motion, it is not able to detect the contact of a finger with the surface, corresponding to a mouse down event. As a substitute for this, an additional spatiotemporal filter detects pauses in finger movements and generates a button event when a pause lasts more than 300ms.

This optical finger tracker offers a solution to a more direct manipulation of digital objects. It allows multiple users simultaneous interaction with their two hands on large surfaces. The system could even be deployed in ”unsafe” locations such as public spaces, as all the costly and fragile equipment can be deployed out of reach from the users who are only interacting on the projected surface.

3.2.2 Camera-Based Interface for Interaction with Mobile Handheld Computers

Hachet et al. [Hachet et al. 2005] developed a new three degrees of freedom user interface for 2D and 3D mobile applications adapted to the characteristics of handheld computers. The interface is based on the movements of a target held by the user in front of a camera that is attached to or integrated in the mobile device as can be seen in Figure 6. The color-codes that are printed on the target make it possible to track the target's movements through analysis of the video stream that is captured by the camera. It can therefore be classified as a monoscopic inside-out tracker using a CCD/CMOS camera to track artificial landmarks (the color-code). Hachet et al. took into account the reduced screen size of mobile devices and aimed to provide an interface that supports the manipulation of data such as 3D scenes while assuring good visualization. Compared to other handheld interfaces this means concretely that the screen is never occluded like it happens with stylus control on a touch screen. Also, it must not be moved, turned or tilted simiar to some other approaches [Hinckley et al. 2000; Rekimoto 1996; Yee 2003]. This is problematic considering the limited viewing angle of these tiny displays. All this is assured through tracking of the color-code imprinted target sheet.

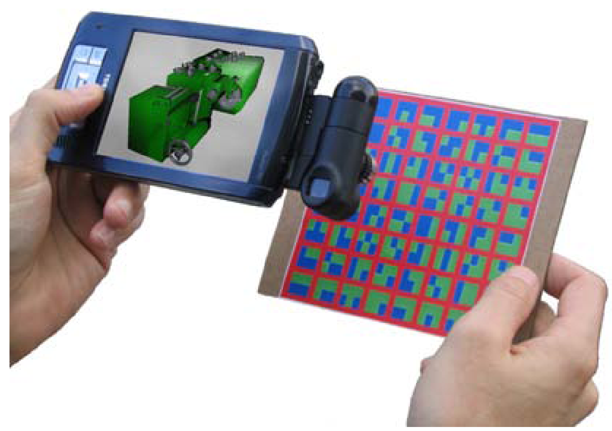

Figure 6: Camera-based interface with the target in front of the camera.

The tracking algorithm developed by Hachet et al. is so efficient and fast that it can be directly computed by the handheld device. The tracking technique has been based on RGB color-codes to allow a wide range of movements of the target and to limit issues due to variable light conditions. The target, shown in Figure 7 is composed of 64 cells, separated by red borders and measures 12x12 cm. Each cell is composed of two horizontal lines, the upper line relates to the target's x coordinate while the lower one relates to its y coordinate. Each line consists of a triplet of blue and green colors, corresponding to a binary code. Blue represents 0 and green represents 1. So the first line of any cell contains the code of its column number and the second line the code of its line number.

The algorithm analyzes the pixels of the video input and assigns each pixel to one of the three red, green and blue clusters according to the maximum of its components. The technique that determines the location on the target where the camera is pointing to is very simple. From the reference point, called the seed, that is initially set to the center of the input image the algorithm searches for the four borders of the cell (left, right, top, bottom), as seen in Figure 8 a). This is achieved by following the simple rule: while the pixel is not red, consider the next pixel. Once the borders of a cell have

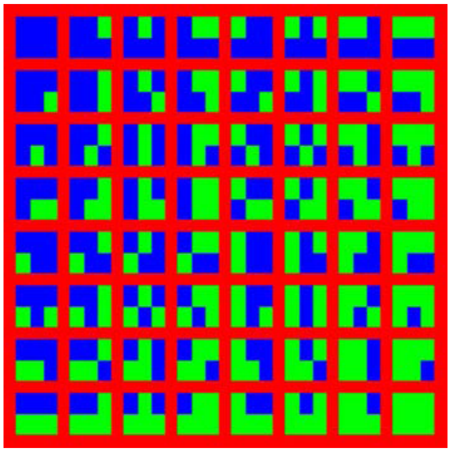

Figure 7: Color-codes on the target.

been identified, the color codes can be recovered simply by reading the color values of the six pixels encapsulated in the cell, shown in Figure 8 b). The color-codes tell the application towards which cell the device is pointed. The first cell in the first line, for example, corresponds to 000 (blue, blue, blue) for both code lines. If the initial seed does not lie within a cell (Figure 8 c)) which means that the starting pixel is red (A), the algorithm scans the next up-right pixel until a non-red pixel is found (B). From this non-red pixel the algorithm continues to search in up-right direction until another red pixel is found (C). The new seed (D) is then defined as the middle of the distance BC. Also, to make this technique more robust and to detect errors that could occur from tilting the target, a test [Hachet et al. 2005] is performed before determining the borders of the cell.

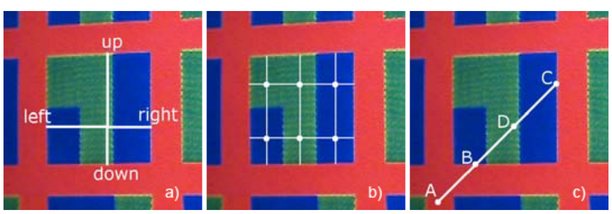

Figure 8: Different stages of the tracking algorithm.

Once the algorithm has determined the cell the camera is pointing towards, it calculates the position of the target in relation to the camera. This is achieved by calculating the x, y and z coordinates of the location on the target on which the camera is centered:

$$x_{target} = N * W + ls * \frac{W}{lr} \quad (6)$$

N represents the column number, W is the column's width on the target, ls stands for the number of pixels between the cell's left border and the seed, and lr corresponds to the number of pixels between the left and the right border. The y_{target} coordinate is calculated analogously. The target's (x, y) position in relation to the camera is directly given by x_{target} and y_{target} while the z coordinate is derived from lr. As the distance between target and camera increases, the distance from left to right decreases as well as the amount of pixels on this path. This enables the user to perform precise pan and zoom maneuvers with the target, as the algorithm calculates x, y and z coordinate for each frame. The coordinates are absolute since they are independent from previous records.

Hachet et al. developed a few test applications to exhibit the usability of this user interface. Visualization of large documents such as maps can be handled intuitively simply by moving the target up, down, left, right or towards or away from the camera which results in direct control over pan and zoom without occluding the screen. Also 3D interaction techniques are supported by the system. Translation and rotation of 3D objects (which one of these two is dependent on the press of a button) are directly mapped to the manipulation of the target which gives the user the feeling of really holding the object. For navigation in 3D scenes the forward-backward degree of freedom is used to control forward and backward velocity while the other two DOFs are used to alter the yaw and pitch angles. Even file selection and system control are possible [Hachet et al. 2005].

4 Inertial Tracking

Accelerometers and Gyroscopes fall both into the category of inertial tracking. An accelerometer, as the name might suggest, is a device for measuring acceleration on masses. In order to measure acceleration along three axes of an object in Euclidean space, three accelerometers are needed, each mounted perpendicular to an axis. A gyroscope keeps track of an object's orientation by measuring the vibration of oscillating masses. Both sensors need to be placed directly on the object that is supposed to be tracked, making them only usable as inside-out trackers, since there would be no use in placing them in the environment. This makes it possible, though, to measure movements and rotations in six degrees of freedom without hindering the user through wires or restricting the area of operation. Therefore inertial tracking enables the user to work effectively sourceless in a relatively large area compared to other tracking methods. The drawback of these sensors is that while they measure movement and rotation, they cannot give information about their position.

4.1 Technical Background

Describing accelerometers from a hardware point of view, they are quite simple devices. They contain a proof mass that consists of moving parts which are very small and light in order to reduce their moment of inertia. This proof mass is suspended by a hairspring taking up all the backlashes that are caused through movement of the device. The force that affects the proof mass on the spring is recorded to calculate the acceleration in the direction of an axis. But since gravity constantly affects the proof mass, the accelerometer when resting on a table, will still read acceleration due to earth's gravity at that location.

The first gyroscopes were built with a spinning wheel whose axis is free to take any orientation. Through the rotation the wheel maintains its position while the gyroscope itself is being turned. This effect is based on the principle of angular momentum. Unfortunately conventional gyroscopes are too large to be used in human motion tracking. Around 1990 a new class of smaller and also cheaper gyroscopes called coriolis vibratory gyroscopes (CVGs) became available. CVGs require no spinning mass and consist of a proof mass that oscillates at high frequency. The vibration of the proof mass is used to determine the angular velocity. This approach is based on the measurement of the Coriolis Force.

To calculate the relative position of a moving inertial sensor, a double integration is applied to the linear accelerometer output. The orientation, on the other side, is determined by single integration of the angular velocity rates. Periodic re-calibration of the actual positions and orientations is necessary as they become sensitive to drift because of the integration.

4.2 Using Inertial Tracking as a Human-Robot Interface for an Automated Wheelchair

Christian Mandel et al. [Mandel et al. 2007] developed a system that makes use of inertial tracking as controlling equipment for an autonomous wheelchair. The tracker is a small sized inertial measurement unit (IMU), like the commercially available XSens MTx. The IMU is an accurate 3 DOF orientation tracker that is mounted at the back of its operator's head, as seen in Figure 9. The IMU permanently measures posture values that are converted into appropriate steering commands. It provides drift-free 3D orientation as well as kinematic data about 3D acceleration, 3D rate of turn and 3D earth-magnetic field. Mandel et al. developed two approaches for steering the wheelchair, one by forwarding the head-posture dependent joystick-like signals for direct steering, and the other by interfacing an autonomous navigation module. The former will be explained in more detail.

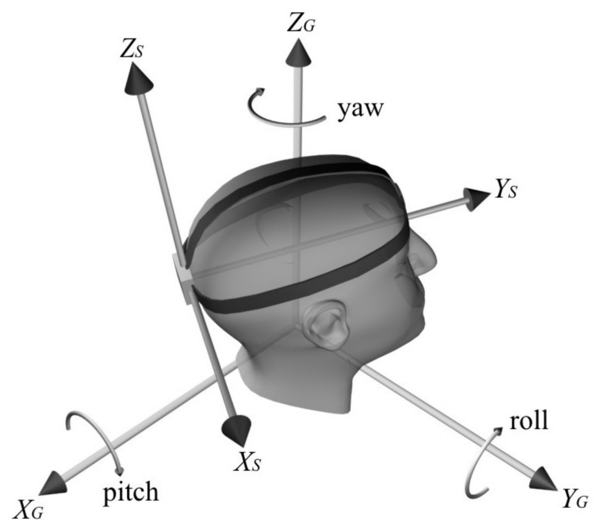

Figure 9: Schematic view of the IMU mounted at the back of the users head. Also visible are the global coordinate system G and the sensors local coordinate system S.

The hardware consists of a wheelchair (Figure 10) that is equipped with two laser range finders, mounted beneath the operator's feet. The laser range finders sense distances to nearby obstacles. Two incremental encoders measure wheel rotation for dead reckoning. An omnivision camera system is mounted behind the user's head and a laptop processes the data. The IMU is positioned at the back of the user's head. The IMU's configuration has been set to output Euler angles that describe the orientation of its local coordinate system S with respect to the fixed global coordinate system G. A single IMU reading is referred to as the triple $\Phi = (\psi, \varphi, \theta)$, as defined in (7).

$$\begin{aligned} \psi &= pitch = rotation\ around\ X_G \in [-90^\circ ... 90^\circ] \\ \varphi &= roll = rotation\ around\ Y_G \in [-180^\circ ... 180^\circ] \\ \theta &= yaw = rotation\ around\ Z_G \in [-180^\circ ... 180^\circ] \end{aligned} \tag{7}$$

In order to use the IMU as a joystick replacement, the device has to be calibrated so that the minimal, the mean and the maximal deflection of a specific person's head is adopted with regards to each

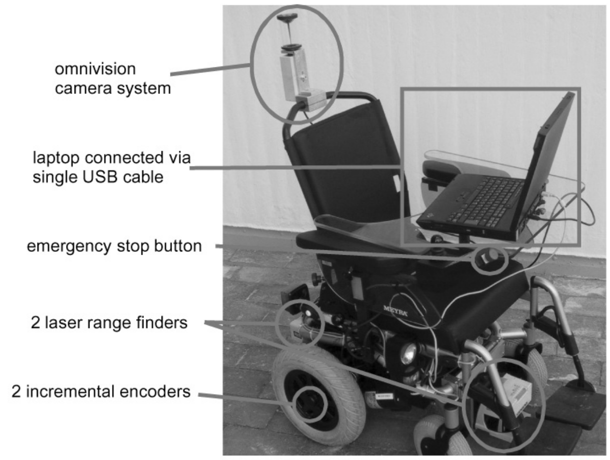

Figure 10: The autonomous wheelchair with its sensorial equipment and a laptop for processing.

axis in use. To achieve this P_{min} and P_{max} are assumed to be the IMU-readings of a user pitching his head forwards and backwards with maximal deflection. In analogy, R_{min} and R_{max} represent the minimal and maximal roll deflection of the user's head. Calculating the arithmetic mean of the ψ and φ components of each of the four sets results in an approximation for the user's minimal and maximal head deflections ψ_{max}, ψ_{min}, φ_{max}, and φ_{min}. Furthermore the rest position of the person's head is described as $\psi_0 = \frac{\psi_{max}+\psi_{min}}{2}$ and $\varphi_0 = \frac{\varphi_{max}+\varphi_{min}}{2}$. A dead zone around ψ_0 and φ_0 is defined by introducing ψ_0^+, ψ_0^-, φ_0^+, and φ_0^- (8), to allow the user to move his head a bit without causing unintended motion.

$$\begin{aligned} \psi_{valid} &\in [\psi_{max} ... \psi_0^+] \cup [\psi_0^- ... \psi_{min}] \\ \varphi_{valid} &\in [\varphi_{max} ... \varphi_0^+] \cup [\varphi_0^- ... \varphi_{min}] \end{aligned} \tag{8}$$

$$\begin{aligned} v &= c_v \frac{\psi_{valid} - \begin{cases} \psi_0^+ : \psi_{valid} > \psi_0^+ \\ \psi_0^- : \psi_{valid} < \psi_0^- \end{cases}}{\begin{matrix} +\psi_{max} - \psi_0^+ : \psi_{valid} > \psi_0^+ \\ -\psi_{min} + \psi_0^+ : \psi_{valid} < \psi_0^- \end{matrix}} \\ w &= c_w \frac{\varphi_{valid} - \begin{cases} \varphi_0^+ : \varphi_{valid} > \varphi_0^+ \\ \varphi_0^- : \varphi_{valid} < \varphi_0^- \end{cases}}{\begin{matrix} +\varphi_{max} - \varphi_0^+ : \varphi_{valid} > \varphi_0^+ \\ -\varphi_{min} + \varphi_0^+ : \varphi_{valid} < \varphi_0^- \end{matrix}} \end{aligned} \tag{9}$$

A valid head-joystick command (v, w) is now computed as in (9) if the input values $(\psi_{valid}, \varphi_{valid})$ apply to (8). The constants c_v and c_w are used to map v and w onto the velocity-domain of a particular vehicle. In order to prevent strong feedback effects between the translational acceleration v' of the wheelchair and the user's head pitch angle ψ, and therefore achieve a smooth acceleration behavior, a basic damping mechanism has been implemented. The damping mechanism replaces a velocity command v_t at a given point in time by the arithmetic mean of former velocity commands $\dot{v_t}$ (10).

$$\dot{v_t} = \frac{1}{n} \sum_{n=0}^{n_{max}} v_{t-n} \tag{10}$$

The approach that uses the information from the IMU as input for a geometric path-planning algorithm has the advantage of not putting the user in a continuous control loop like the previous approach does. Instead, the user only has to face his head to a desired target position once and issue a voice command to let the system autonomously execute the given task. In Mandel's et al. article [2007] this approach is described in further detail.

Compared to regular steering via joystick input there are still some disadvantages that need to be corrected. One problem that has been discovered is that users were not able to steer the vehicle on an almost straight line while moving with high translational speed. This is due to the dead zone disregarding head-joystick commands from within a static interval around the head's rest position.

Criterion	Common Joystick	IMU as Head-Joystick
∅ time of travel	30.73 s	55.03 s
∅ length of travel	22.45 m	25.03 m
∅ average speed	0.76 m/s	0.50 m/s
∅ safety layer interventions	111.04 ms	445.76 ms

Table 1: Results of a comparative experiment.

Table 1 displays the results of a comparative experiment in which 15 untrained participants first used a common joystick, and afterwards the IMU-based head-joystick. It is shown that driving the same path when using the IMU-based head-joystick the average travel time is about one third longer compared to the travel time using a normal joystick. Also safety layer interventions occur four times as often using the head-joystick. The safety layer is a software module that permanently monitors the user's input for translational and rotational velocity. It initiates a hard full stop if a situation occurs where a combination of the desired speeds would not allow for a safe breaking maneuver anymore.

All in all it goes to show that this control type does indeed allow people who suffer from high level quadriplegia but are still able to move their heads to take full control of a wheelchair. Although there obviously is room left for improvement in the system's intrinsic inaccuracies.

5 Biosignal Sensors

The tracking of biosignals can also be used to control certain applications. This sort of tracking does not fall into the category of conventional tracking as described earlier. However, biosignal sensors are a promising alternative user interface for specific applications.

"Biosignal" is a summarizing term for all kinds of signals that can be (continually) measured and monitored from biological beings. It is often used as synonym for bio-electrical signals, like Electroencephalogram (EEG), Magnetoencephalogram (MEG), Galvanic skin response (GSR), Electrocardiogram (ECG) and Electromyogram (EMG). In fact it refers also to non-electrical signals that can be monitored from human beings, like mechanical signals, acoustic signals and visual signals.

In the following two sections two user interfaces are presented that make use of measuring such signals.

5.1 Interfaces for Mobile Devices Using an EMG Controller

Costanza et al. [Costanza et al. 2005; Costanza et al. 2007] developed an electromyogram based wearable input device which recognizes isometric muscular activity related to very subtle or no movement at all. This activity is used as means of input for a mobile device like a cell phone or PDA and enables the user to interact with it without his surroundings noticing. According to Constanza et al. using a mobile device in a social context should not cause embarrassment and disruption to the immediate environment. They want interaction with mobile and wearable devices to be subtle, discreet and unobtrusive and therefore promote the idea of these "intimate interfaces". Vibrating alerts as replacement for ringtones represent an example of this trend.

The biosignals measured by the EMG are electrical signals that are generated through physiological activity, like flexing of a muscle. Their range lies in the order of 100μV to a few mV, therefore their sensing requires high sensitivity low noise amplifiers. These EMG signals can be measured by surface electrodes that basically are Ag/AgCl plates covered with conductive solid gel, similar to those used for standard electrocardiogram (ECG) measurements. A minimum of three electrodes is required: two of them form a differential input pair and the third one is for grounding. Figure 11 shows the constellation of the sensors.

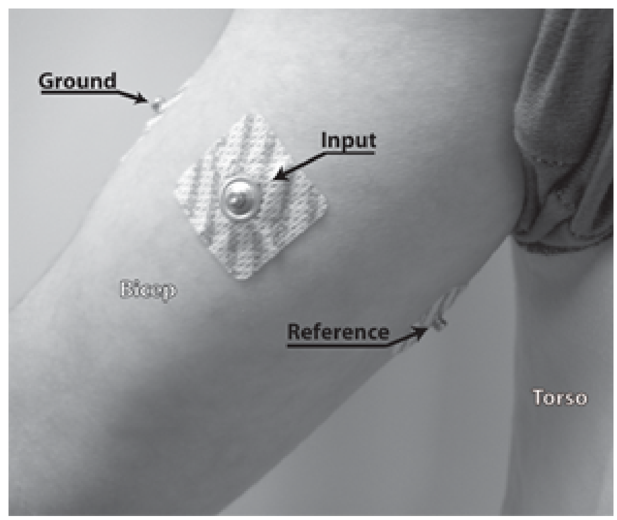

Figure 11: Electrode positions on the upper arm.

The system proposed by Costanza et al. is a small wireless armband controller called the Intimate Communication Armband which can be worn under clothes to make it unnoticeable, as can be seen in Figure 12. The device is completely self contained and senses, amplifies and analyzes the EMG signals from the bicep. As soon as the device recognizes a gesture, it transmits a message containing the device ID and the parameters describing the gesture via Bluetooth to the connected phone or PDA. If muscular activity not corresponding to the defined gestures is sensed, the device remains silent. This means that everyday-movements will not trigger unwanted commands. Further the algorithm that filters the measurements is designed in a way that it is not necessary to calibrate the device for individual people or to especially train certain muscles. Compared to other EMG based systems [Putnam and Knapp 1993; Wheeler and Jorgensen 2003], Constanza's et al. approach chooses to trade variety of gestures that the system is able to recognize for the minimizing of computational complexity, the robustness against false positives, use of only one input channel and avoiding calibration and system training on each user. From the hardware point of view the device consists of a high input impedance amplifier connected to the electrodes, an anti-aliasing filter, a microcontroller to sample and process the EMG signal and a Bluetooth communication module for transmitting the processing results.

The gesture for the recognition algorithm was chosen so that it satisfies two requirements: it has to be natural for people to perform and different enough from normal muscle activity to avoid misclassification by the algorithm. The muscle that was chosen to perform the input gestures is the biceps because it lies superficially, leaving the signal fairly uninfluenced by activity generated through nearby muscles, and it is well defined even in non-athletes. The deployed

Figure 12: The Intimate Communication Armband worn on the upper arm.

peak model, depictured in Figure 13 is based on the standard deviation of the EMG signal and is calculated with a sliding window of 0.2s duration overlapping for 75% of its duration. The standard deviation was chosen to smooth the data and emphasize discontinuities in the energy of the electromyogram while the window size has the longest duration possible that still does not filter out interesting features. A brief contraption of the biceps corresponds to a peak in the standard deviation of the signal. Because of the noisy characteristics of the EMG signal [3], standard peak detection techniques cannot be employed. Instead the algorithm was modeled to detect peaks that follow the pattern of a sequence consisting of three consecutive intervals: a "beginning" interval of low activity (silence) and duration T_B, followed by a "middle" interval of high activity and duration T_M and finally again a low activity "end" interval of duration T_E. High and low activity are defined respectively as the standard deviation of the signal being above a threshold H and below a threshold L. To allow certain tolerance values, the condition on the history is imposed on the average of its values. The condition in the middle has to be satisfied by 50% and the condition on the end by 70% of the samples. Also, it is important to note that because the duration of each contraction has been proven to be more consistent than their intensity, the model definition focuses rather on duration than intensity. A drawback of this technique is that it requires one gesture to be completed before it can be recognized by the algorithm.

Using this setup it is possible to operate mobile devices hands-free and in a subtle mode, as has been demonstrated by Costanza et al. with the navigation of an audio menu [Costanza et al. 2007]. The Intimate Communication Armband was either used to select the current item of a list that was narrated by a voice and advanced automatically, two seconds after the item description was read. Or in a two-arm mode one biceps was used to sequentially advance in the menu while the other biceps was in charge of selecting the desired item.

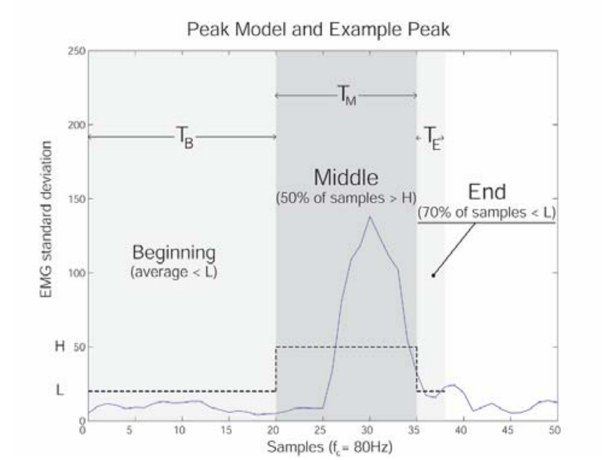

Figure 13: Peak model and example peak that the algorithm detects.

5.2 Using Heart Rate to Control an Interactive Game

The system that Nenonen et al. [Nenonen et al. 2007] proposed shows that even the heart rate, a seemingly uncontrollable biosignal, can be used as means of input for an interactive user interface. The system uses real time heart rate information to control a physically interactive biathlon game called Pulse Masters Biathlon. The underlying idea is that instead of interfacing the game to an exercise bike or other equipment with speed output, the player is to influence his heart rate by means of a freely chosen form of exercising.

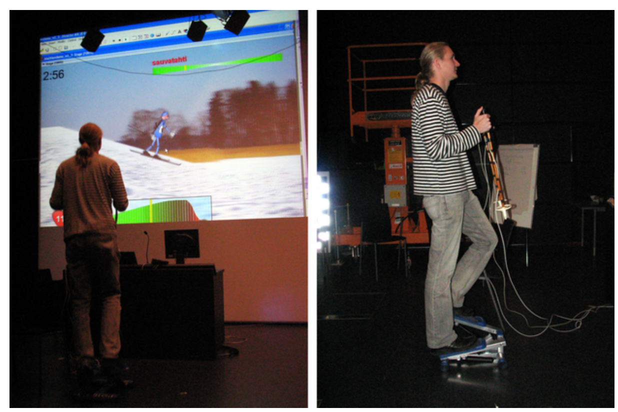

Figure 14: Exercising using a mini stepper and holding the two skiing sticks.

The game consists of two modes, cross-country skiing and target shooting, that alternative each level. The player is connected to an optical heart rate monitor and is also operating two buttons, each one situated at the top of a skiing stick. During the skiing mode the player has to adjust the skiing speed of his on-screen avatar that is skiing on a predetermined track (Figure 13). The skiing speed is directly proportional to the user's heart rate. This means, the higher the heart rate, the faster the on-screen avatar reaches the finish line. Additionally, timed button presses give the avatar a little speed up.

In each of the game's shooting levels the player has to shoot five targets. Only one try is given to hit each target. For each missed shot a ten second time penalty is added to the player's overall completion time. The crosshair is not controlled by the player but instead

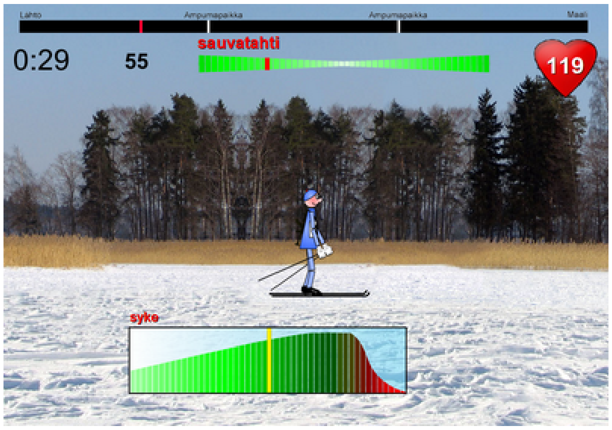

Figure 15: A skiing level in Pulse Masters Biathlon.

moves on a predetermined route on and off the targets. The player has to press one of the trigger buttons exactly when the crosshair is aligned with a target. To maintain gameplay balance the speed of the crosshair is dependent on the heart rate. The higher the heart rate, the faster the crosshair moves over the targets. This means that a player who exaggerates in the skiing part in order to complete the level in the minimum time will have a harder time hitting the targets as his heart rate will still be high from the skiing.

Figure 16: A shooting level in Pulse Masters Biathlon.

Also, for the sake of gameplay balance and to prevent players from exaggerating too extensively, heart rates in the skiing part that are greater than 150 beats per minute (bpm) do not influence the speed anymore. On the other side, if the player's heart rate was under 70bpm, the avatar would stand completely still so that operation of the game without exercising is rendered impossible. In the shooting part there is no upper limit to the heart rate but it has been regulated so that shooting at a heart rate less than 90bpm is supposed to be very easy, rising in difficulty as the bps increase.

From the hardware point of view the system is built with a projector, two trigger buttons and an optical heart rate monitor which are all connected to a standard PC. The optical heart rate monitor consists of a light sensor and a light which are attached on the opposite sides of a finger. The blood that flows through the finger affects the light that shines through it. These changes of the light are being recognized by the optical sensor and calculated into heart beats per minute. The disadvantage of the optical heart rate monitor is that it needs to be placed very cautiously in order to track the heart rate properly. This is problematic because of the physically active nature of the game which causes the sensor or the light to move and therefore disturb the measurements.

6 Conclusion

Alternative user interfaces are being researched and developed in a huge variety, though most of them have got something in common: they are about tracking the user's motion in some way or the other. Exceptions exists like we have seen in the previous chapter but it can be said that tracking in general is the method of choice for most alternative user interfaces, be it for virtual- or augmented reality applications, office or everyday situations, or as auxiliary means for disabled persons. It was shown that each of the mentioned tracking techniques has their own advantages and disadvantages compared to the others. Hybrid trackers that combine different tracking techniques in order to overcome their disadvantages will have an important role in the future of tracking and alternative user interfaces. The examples of alternative user interfaces that were shown in this paper merely scratched the surface of existing applications. But even these few examples give a good impression of how many of life's tasks can benefit from alternative user interfaces. They can help make complicated tasks easier and impossible tasks possible. Considering what will be possible with the perpetual advancements in research and technology one can only remain in thrilled anticipation.

References

BERARD, F. 2003. The magic table: Computer-vision based augmentation of a whiteboard for creative meetings. In *Proceedings of the IEEE International Conference in Computer Vision, Workshop on Projector-Camera Systems (PROCAMS03)*, CD–ROM.

BLOOD, E. 1989. Device for quantitatively measureing the relative position and orientation of two bodies in the presence of metals utilizing direct current magnetic fields. *U.S. Patent No. 4,849,692*.

COSTANZA, E., INVERSO, S., AND ALLEN, R. 2005. Toward subtle intimate interfaces for mobile devices using an emg controller. In *Proceedings of the 2005 ACM Conference on Human Factors in Computing Systems (CHI2005)*.

COSTANZA, E., INVERSO, S., ALLEN, R., AND MAES, P. 2007. Intimate interfaces in action: Assessing the usability and subtlety of emg-based motionless gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing systems*, 819–828.

DORFMUELLER-ULHAAS, K. 2002. *Optical Tracking - From User Motion To 3D Interaction*. PhD thesis, Vienna University of Technology.

HACHET, M., POUDEROUX, J., AND GUITTON, P. 2005. A camera-based interface for interaction with mobile handheld computers. In *Proceedings of the 2005 symposium on Interactive 3D graphics and games*, 65–72.

HARDENBERG, C. V., AND BERARD, F. 1989. Bare-hand human-computer interaction. *Workshop on Perceptive User Interfaces*.

HINCKLEY, K., PIERCE, J., SINCLAIR, M., AND HOROVITZ, E. 2000. Sensing techniques for mobile interaction. In *Proceedings of the 13th annual ACM symposium on User Interface Software and Technology (UIST 2000)*, 91–100.

JIANG, B., AND NEUMANN, U. 2001. Extendible tracking by line auto-calibration. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*, 97–103.

LETESSIER, J., AND BERARD, F. 2004. Visual tracking of bare fingers for interactive surfaces. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, 119–122.

MANDEL, C., ROEFER, T., AND FRESE, U. 2007. Applying a 3dof orientation tracker as a human-robot interface for autonomous wheelchairs. In *Proceedings of the 10th International Conference on Rehabilitation Robotics*.

NAIMARK, L., AND FOXLIN, E. 2002. Circular data matrix fiducial systems and robust image processing for a wearable vision-inertial self-tracker. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2002)*, 120–28.

NENONEN, V., LINDBLAD, A., HAKKINEN, V., LAITINEN, T., JOUHTIO, M., AND HAMALAINEN, P. 2007. Using heart rate to control an interactive game. In *Proceedings of CHI 2007 - Reach beyond : Conference on Human Factors in Computing Systems*, 853–856.

PUTNAM, W., AND KNAPP, B. 1993. The use of the electromyogram in a man-machine interface. In *Proceedings of the Virtual Reality and Persons With Disabilities Conference*.

RAAB, F., BLOOD, E., STEINER, T., AND JONES, H. 1979. Magnetic position and orientation tracking system. *IEEE Transactions on Aerospace and Electronic Systems 15*, 5, 709–718.

REKIMOTO, J. 1996. Tilting operations for small screen interfaces. In *ACM Symposium on User Interface Software and Technology (UIST 96)*, 167–168.

SIMON, G., FITZGIBBON, A., AND ZISSERMAN, A. 2000. Markerless tracking using planar structures in the scene. In *Proceedings of the International Symposium on Augmented Reality 2000 (ISAR00)*, 120–28.

WHEELER, K., AND JORGENSEN, C. 2003. Gestures as input: Neuroelectric joysticks and keyboards. In *IEEE Pervasive Computing*, vol. 2.

YEE, K.-P. 2003. Peephole displays: pen interaction on spatially aware handheld computers. In *Proceedings of the conference on Human factors in computing systems*, ACM Press, 1–8.