Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 14 (2012) 1 – 9

Procedia
Computer Science

Proceedings of the 4th International Conference on Software Development for Enhancing Accessibility and Fighting Info-exclusion (DSAI 2012)

AsTeRICS, a Flexible Assistive Technology Construction Set

Roland Ossmann[a]*, David Thaller[a], Gerhard Nussbaum[a], Franz Pühretmair[a], Christoph Veigl[b], Christoph Weiß[b], Blanca Morales[c], Unai Diaz[c]

[a]*Kompetenznetzerk KI-I, Altenberger Straße 69, 4040 Linz, Austria*
[b]*Fachhochschule Technikum Wien, Höchstädtplatz 5, 1200 Wien, Austria*
[c]*Fundación INGEMA, Paseo de Mikeletegi 1-3, 20009 Donostia - San Sebastian, Spain*

Abstract

Over the last decades a considerable number of information and communication technology based Assistive Technology devices have become available for people with disabilities. These Assistive Technology devices often ask for adaptation of software and/or hardware to fit the users abilities before they can be used.
Within the Project AsTeRICS, a flexible and affordable construction set for the implementation of user driven assistive technologies solutions will be developed. This allows the combination of different sensors to process and manipulate the sensor data to control any supported device. This paper will show how a webcam mouse (head tracker) and a single switch mouse can easily be created and tailored to the user needs and possibilities. Additionally, results of user tests with the head tracker will be presented.

Keywords: AT Construction Set; Headtracker; Single switch mouse

1. Introduction

More than 2.6 million people in Europe have problems with their upper limbs and therefore many of them depend on Assistive Technologies (AT) [1]. Over the last decades a considerable number of information and communication technology based AT devices have become available for people with disabilities. These AT devices often ask for adaptation of software and/or hardware to fit the user's abilities before they can be used. All too often, assistive tools that have been optimized for particular applications cannot be used in other

* Corresponding author. Tel.: +43-732-2468-3774; fax: +43-732-2468-3789.
E-mail address: ro@ki-i.at.

 doi:10.1016/j.procs.2012.10.001

situations out-of-the-box. Additionally some people cannot be supplied with AT devices at all, due to the limits of adaptability or unaffordable costs of the necessary adaptations.

The aim of the AsTeRICS (Assistive Technology Rapid Integration & Construction Set) project is to reshape this situation substantially. AsTeRICS provides a flexible and affordable construction set for building assistive functionalities which can be highly adapted to individual user's needs. The scalable and extensible system allows integration of new functions without major changes. AsTeRICS opens access for people with severe motor disabilities to a standard desktop computer but also to embedded devices and mobile services, which have not offered highly specialized user interfaces before [2]. Therefore AsTeRICS respects the strong need for flexible, adaptable AT functionalities accompanying people with disabilities away from the desktop, enabling them to interact with a diverse and fast changing set of deeply embedded devices in our modern environment.

Major parts of the software suite are provided as Open Source software and are published on the project website www.asterics.org.

1.1. Technical concept

The AsTeRICS Personal Platform implements a set of building blocks for the realization of assistive technology. The base of the concept is an embedded computing platform executing the AsTeRICS Runtime Environment with OSGi [3] plugins providing desired AT-functions. Sensors and actuators which allow the system to interact with its environment can be connected to the platform (see Fig. 1).

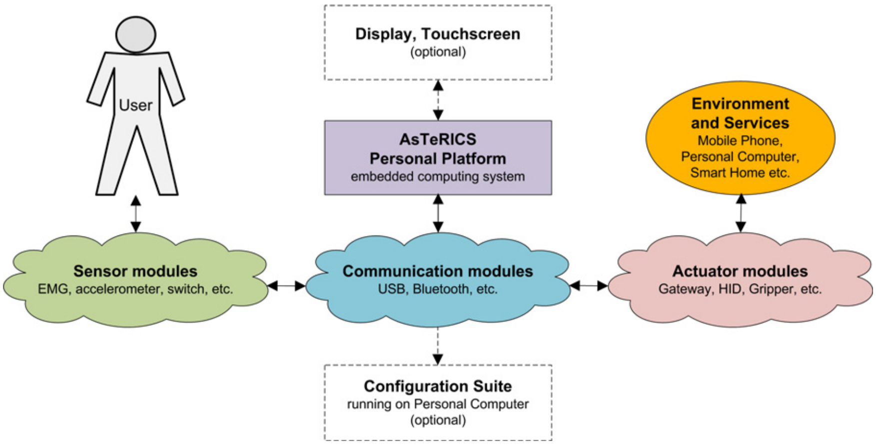

Fig. 1. Schematic concept of AsTeRICS

Sensors include classic AT-interfaces such as switches and special joysticks but also emerging sensor techniques like Computer Vision and Brain Computer Interfaces (BCI). Actuators include simple switches and digital-to-analogue conversion but also more complex modules like keyboard-, mouse- or joystick emulation, generic infrared remote control, KNX [4] interface to existing building automation systems or mobile phone access. If desired, an On-Screen Keyboard displayed on an optional LCD-touchscreen provides selection or adjustment of system parameters via scanning, voice-feedback and touchscreen interaction [5].

Data from the sensors are processed, translated and mapped on the platform and are used to control the actuators. The configuration of the system i.e. the data flow/processing can be done remotely using the

AsTeRICS configuration suite, which provides a graphical user interface to set up and tailor the components to the specific needs of the primary user.

2. Tailoring the System to the Users

To tailor a complex and powerful construction set to the user needs and requirements, a powerful configuration system is needed. Within the AsTeRICS project, the AsTeRICS Configuration Suite (ACS) will fulfill this task – see Fig. 2. This is a graphical programming system, where each component of the system is represented by a plugin in the ACS. These plugins can be configured within an integrated editor and are connected to each other by data channels of continuous data and event channels for events. Beside the implementation and modification of models, the ACS can fully control the AsTeRICS Runtime Environment (ARE). Additionally, the ACS can create the graphical user interface for the end users on the ARE.

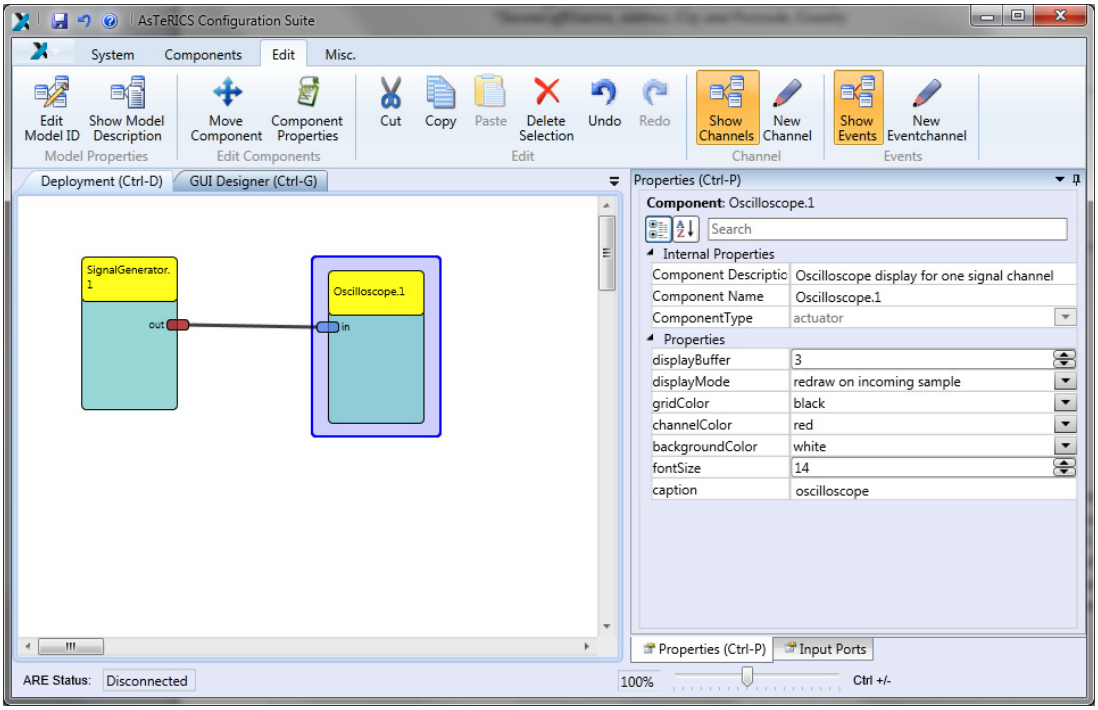

Fig. 2. Screenshot of the AsTeRICS Configuration Suite

The number of supported components is not limited and can be dynamically extended by the project partners and by the open source community. This means, if functionality or the support for any device is missing, a new component can be developed and easily included into the system. Thereby, the system is very flexible and extendable to future requirements. The following sections will give a few examples, how user’s needs and requirements can be achieved with different components, being available and combined.

2.1. A Simple but Efficient Webcam Mouse

Camera based systems have become powerful and cost efficient input alternatives for people with severe motor disabilities during the last decade. The spectrum of available solutions ranges from eye- and head tracking systems to gaze estimation via head-mounted camera frames or binocular remote tracking. Also the price of those systems varies from free DIY open source solutions to high quality systems in the range of 20.000 Euro and more. Some good examples are the Camera Mouse developed by Betke/Gips at Boston College [6], the ITU Gaze Tracker [7], the commercially available MyTobii P10 gaze tracker [8] and the research activities supported by the European network of excellence for Communication by Gaze Interaction (COGAIN, http://www.cogain.org/).

The different approaches in computer-vision-based input also imply different requirements for the users capabilities, as advantages and disadvantages depending on the individual situation: High quality gaze-controlled mouse cursor movement allows very efficient interaction for example fast key selection via on-screen keyboard. On the other hand, the precision of state-of-the-art systems mostly is not sufficient to use e.g. a normal Windows desktop with its tiny interaction elements. Additionally, the "midas-touch" problem [9] can be difficult to handle for some people (everything on the screen is selected by simply looking at it). The high cost and other disadvantages of remote gaze-estimation systems keeps head-tracking based input systems attractive for people with sustained fine motor control of the head.

In the AsTeRICS project, several plugins have been developed for webcamera-based head tracking support. Due to the flexible AsTeRICS architecture, these plugins can be combined in various ways to enable individualized head-tracking solutions including different clicking options and adjustment of parameters by the end user via a touch screen. In the following, some interesting AsTeRICS plugins will be described which can be combined to establish camera-mouse functionalities. Subsequently, a model for head-tracking based mouse control with clicking alternatives will be presented.

2.1.1. Computer Vision Plugins for Head Tracking

Currently, two computer vision plugins for head- and facetracking are available, which utilize the OpenCV computer vision library [10]. Both plugins can be used with various image acquisition systems including ordinary low-cost USB webcams, and deliver the current movement of the users nose and chin to other AsTeRICS plugins. As described in the next section, this information can be directly mapped to the x/y coordinates of the computer mouse.

- *FacetrackerLK*: A very efficient face tracking algorithm tailored to low-performance platforms. This algorithm first performs a face localization by applying a trained classifier of haar-like features [11], and then tracks certain feature points of the face (nose and chin) by a Lukas-Kanade optical flow algorithm [11]. This approach delivers about 30 updates per second on a standard Intel Atom processor which is used in notebooks or tablet computers.
- *FacetrackerCLM*: This plugin utilizes a sophisticated version of the constrained local model search (CLM) [12], which represents an alternative approach to fitting a deformable shape model. This allows tracking of various facial landmarks, but needs more processing power than the FaceTrackerLK plugin to deliver reasonable update rates for direct mouse cursor control.

2.1.2. Remote- and Local Mouse Emulation Plugins

These plugins transform incoming signals at input- or event ports into real mouse cursor activities (x/y-position changes, right- and left mouse button clicks, drag, or scroll-wheel activity). Whereas the *Mouse* plugin performs these actions via software emulation and API calls on the local system, the *RemoteMouse* plugin builds a connection from the AsTeRICS Personal Platform to the *HID-Actuator* USB dongle which has been developed in course of the project. The *HID-Actuator* can be plugged into any target computer including

Windows, Linux or Mac machines or even a Playstation 3 game console, and is capable of emulating a mouse, keyboard and joystick on these computers. Thus, no driver installation is necessary on the target system and desired input configurations (like a camera-controlled mouse cursor) can be used on any computer simply by plugging in the *HID-Actuator*. Both mouse emulation plugins provide input ports for x- and y- mouse cursor position and events port for the different clicking- and mouse-wheel activities.

2.1.3. Plugins for Click Control via Digital Input or Dwell Time

The creation of desired mouse clicks is essential for a useable mouse replacement, where a simple left click is most important as all other click types can be created by multiple left clicks. Various strategies for click creation are possible, including dwell-time (automatic clicking when no mouse cursor movement occurred for a defined timespan) or direct clicking via external switches. For the detection of no (or low) movement, a combination of the *DeadZone* and *Timer* plugins can be used: the *DeadZone* plugin is configured to detect a certain level of x/y activity of the connected Computer Vision plugin. Thus, the *Timer* plugin receives start- or stop events. The Timer plugin sends a click event to the mouse plugin if a defined time of inactivity is reached.

Direct click control via external switches is the desired option when the user can handle one or multiple switches (e.g. with finger, elbow, shoulder or toe). Switch-controlled clicking together with a head-tracking based cursor control allows input efficiency of 30 character selections per minute and more via on-screen keyboard [11]. To connect external switches to the system, different strategies can be used, including the buttons of a standard joystick (via the *JoystickCapture* plugin) or mouse (via the *MouseCapture* plugin) or special Assisitve Technology buttons like the JellyBean switches [13]. To connect external switches, either the *DigitalIn* plugin (and the associated *DigitalIn* CIM hardware module developed as part of the project) or the *Arduino* plugin (and the well-known Arduino microcontroller, see http://www.arduino.cc/) can be used. The advantage of the *DigitalIn* CIM is that standard AT-switches can be connected via a jack-plug and no soldering is needed. Both options allow the connection of multiple external switches and the utilization of the user control input in other AsTeRICS plugins.

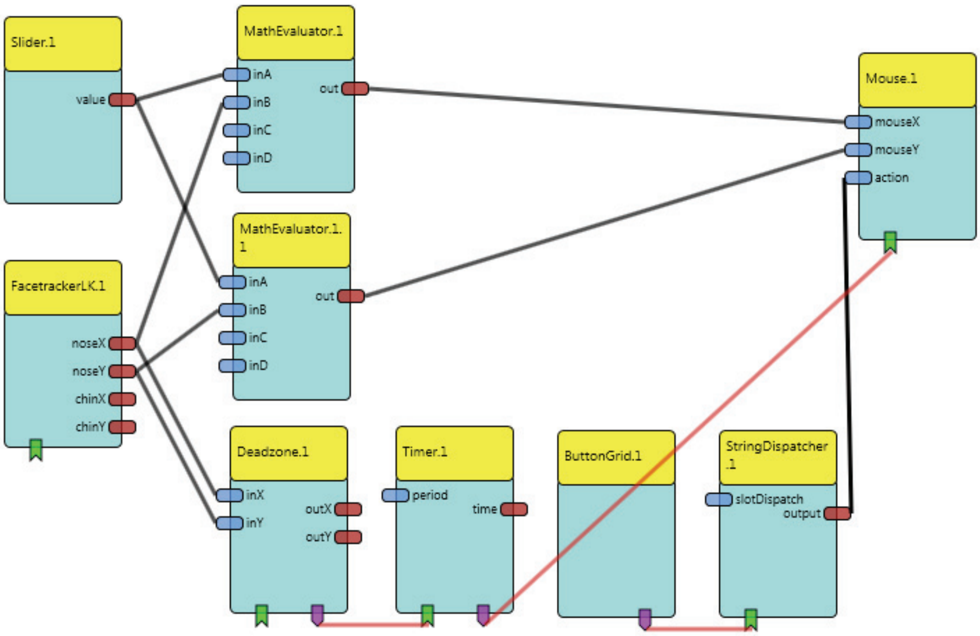

Fig. 3. Combination of AsTeRICS plugins to a functional webcamera-based mouse replacement

Fig. 3 shows a webcamera-controlled local mouse cursor based upon the connection of the described plugins. Left clicks are issued by a defined time of inactivity (dwelling). A graphical slider element allows the adjustment of the mouse acceleration. For this purpose, the *MathEvaluator* plugin is used to scale the x/y movement delivered by the *FacetrackerLK* plugin with respect to a slider position. Possible extensions to the above model include more GUI elements for parameter adjustment and click selection (double-, drag- and right clicks) and scroll-wheel control.

2.2. A Model for a User With Very Limited Possibilities

The following model allows writing texts and controlling the environment with just one switch. To write text, the on screen keyboard OSKA [14] will be used. The OSKA keyboard is very flexible and already included standard keyboard layouts can be extended with additional special keys and word prediction. It's also possible to create custom grids for special controls, like it's used in this model for a television control.

The model (see Fig. 4) for this requirement is quite simple, because many things, like scanning and button press (e.g. "trigger the nurse call") event handling, are already done by OSKA.

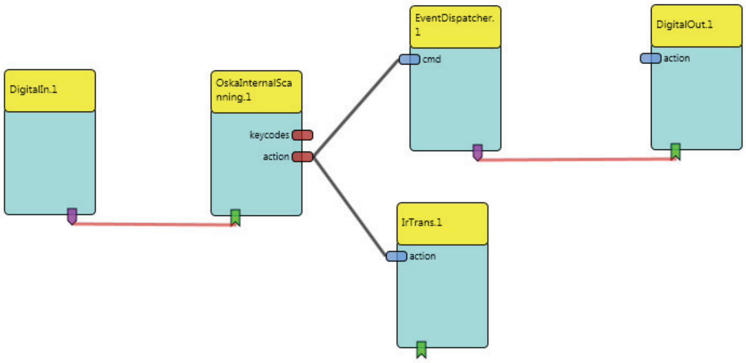

Fig. 4. Model for the usage of OSKA

The *DigitalIn* receives the switch input, which will be used by OSKA to press the currently selected button. The button commands will then be sent to the *EventDispatcher*, which triggers the corresponding actions like the nurse call at the *DigitalOut* component or the *IrTrans* module, an infrared transmitter which can send IR commands like TV control commands.

2.3. Increasing the Complexity – the One Switch Mouse

In the following model, the mouse on the PC will be controlled by just one switch. This allows users, who can press only one switch, to control the whole PC [15]. The model works as follows: while the button is pressed, the mouse pointer changes its cursor every n seconds, where n can be set depending on the speed of the user. In the beginning, the mouse cursor becomes a red cross (stop action), afterwards a left arrow (move mouse left), a top arrow (move mouse upwards), right arrow (move mouse right), bottom arrow (move mouse downwards), left click, double click and finally a right click symbol. By releasing the switch again, the mouse performs the command, specified by the cursor symbol which was visible when the button got released. For example when the user releases the button when the cursor was a left arrow, the mouse will start moving left, when the double click icon got selected a double click will be executed. If a mouse movement has been chosen, the mouse moves until the button is pressed the next time, then the mouse stops. When releasing the button again while the red cross was still visible, no further action will be executed. Otherwise the cycle of the 8 already mentioned icons starts again.

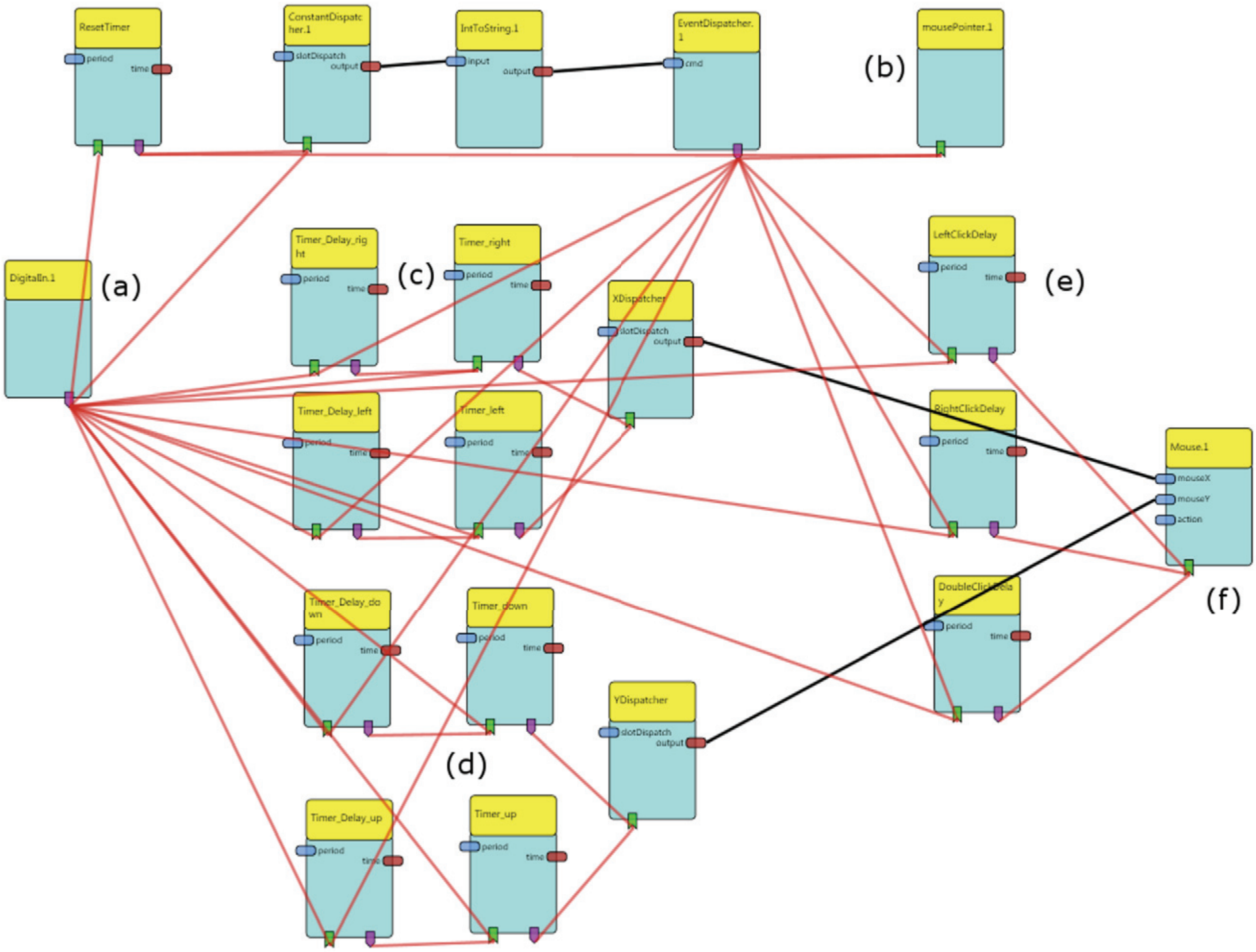

Fig. 5. Model for the single switch mouse

The model (see Fig. 5) for the one switch mouse looks very complex at a first view, but when going into detail, it is quite easy to understand. The model starts with the *DigitalIn* (a), where the switch is connected to and where an event will be fired, if this switch will be pressed. This interaction starts the changing procedure of the mouse pointer (b) and stops the timer for the mouse movement (c), (d). When the button is released again, depending on the selection, the mouse x movement (c), the mouse y movement (d), a mouse button click (e) or just a movement stop will be initiated. This command will be transmitted to the mouse control (f).

3. User Tests

The user tests were performed in different research and healthcare facilities from Austria, Poland and Spain, and with the participation of people with motor disabilities in their upper limbs. Specifically, the sample was composed of 51 users from three different countries (17 Austria, 10 Poland and 24 Spain, 25.5% females). The age range of participants was 19 to 61 years-old, with a mean age of 36.04 years (SD=11.7). The following results are focusing the webcam-mouse to present the usability of one of the presented models.

The diagnoses of the participants were very heterogeneous; leaded by cerebral palsy (29.4%), and followed by spinal cord injury (20.8%). The remaining users suffered from muscular dystrophy, spasticity, spastic tetraplegia, Friedreich ataxia, multiple sclerosis, spinal muscular dystrophy, stroke, traumatic brain injury, amyotrophic lateral sclerosis, and Parkinson disease. With regard to the type of disability 55% had tetraplegia, 11.8% suffered tetraparesia, 29.4% had either hemiplegia or other conditions (spastic, etc.), and the remaining users suffered from paraplegia.

Regarding the use of technology among the participants, in general all the participants reported feeling very interested in technologies, and specifically, 94% of the sample often used their personal computers almost daily. Moreover, 72% of them declared that they need some type of device adapted to interact with their computer.

3.1. Procedures

The study was approved by ethical committees in each user trials' country in order to follow the European ethical guidelines; moreover, the participants who agreed to participate gave signed confirmation (i.e. consent form) or oral consent.

The activities proposed to the users to test the AsTeRICS webcam-mouse sensor were the following: opening/closing audio/video files, and up/down volume and play/pause music in a music/video player program.

3.2. Results

For more than half of the sample (61%) the webcam-mouse model was easy and comfortable to use, but others (39%) had great difficulties to move the head and control the fine movements. In general, most of the users managed to carry out the tasks, and they did not show great difficulties to start using the system, but some usability flaws were observed with the interaction with the device.

Regarding the position of the participant with reference to the webcam, due to their motor impairment some users often had a twisted neck and the camera did not properly focus on the users' face. When this was observed the evaluators took the camera and they leaned it's to fit the specific face line of the user, but then the movement of the head did not correspond exactly with the natural head movement of the user.

Among the users with cerebral palsy, 67% of them had problems to use the webcam-mouse due to muscle control difficulties that this condition involves. In this case, involuntary movements of the head, or lack of control of voluntary movements were cause of not performing correctly the task proposed by the evaluators. Another difficulty found was that the corners of the screen were not easily reachable due to the need of forcing the movement of the neck to reach the desired position. In the cases in which the sensitivity of the pointer was customized in order to make less head movement to reach the corners, the movement on the centre of the screen became very fast for some of the users, and they showed difficulties to make precise movements.

The last problem found was that the face recognition software sometimes could choose another face that was in the range of the camera, changing the control to other persons in the room that could be at the back of the user.

Regarding the use and acceptability of the system, several differences of opinion were found among participants. On the one hand, users who interacted with the model correctly said that the idea was original, very useful and easy to use regardless their profile. Moreover, some users were used to using similar devices and really liked not having to put a sticker on the nose (i.e. like in the SmartNav application). On the other hand, the users with paraplegias derived from a cerebral palsy had problems to control the device. The rest of the users (61%), each one with a different type of disability (neurodegenerative diseases, tetraplegias or hemiplegias), concluded that this model is comfortable and not very tiring. In the last case, they also stated that maybe in the future with the worsening of their respective diseases, the device could become more useful.

4. Conclusion and Outlook

The AsTeRICS project provides a very flexible and tailorable AT construction set. Due to the openness of the system, the number of supported devices is more or less unlimited. The usage of standard hardware components like webcams or the Arduino microcontroller in combination with standard PC hardware, and the fact, that the software is open source, makes the system very affordable.

The results of the user tests are very promising, many users are well supported by the system. Nevertheless, not all users were pleased with the system, further development will take the user comments and requirements into account. Especially, developing and tailoring models to their requirements will be more in the focus. Lucky us, tailoring the system to the user requirements is one of the strengths of AsTeRICS.

Acknowledgements

The AsTeRICS project is partially funded by the European Commission under the Seventh (FP7 - 2007-2013) Framework Programme for Research and Technological Development, Grant Agreement number 247730.

References

[1] Eurostat: Population and Social Conditions: Percentual Distribution of Types of Disability by Sex and Age Group. All accessed on 13th of June 2005. http://epp.eurostat.cec.eu.int
[2] Jakob E. Bardram, Alex Mihailidis and Dadong Wan. Pervasive Healthcare: Research and Applications of Pervasive Computing in Healthcare. CRC Press. 2006
[3] OSGi Alliance - the Open Service Gateway Initiative. http://www.osgi.org
[4] KNX - the Worldwide Standard for Home and Building Control. http://www.knx.org
[5] Nussbaum, G., Veigl, C. et al. 2011. AsTeRICS - Towards a Rapid Integration Construction Set for Assistive Technologies, Proceedings of the AAATE 2011 - Everyday Technology for Independence and Care, pp.776-793. IOSpress, 2011
[6] James Gips, Margrit Betke, Peter Fleming: The Camera Mouse: Visual Tracking of Body Features to Provide Computer Access for People With Severe Disabilities, Ieee Transactions On Neural Systems And Rehabilitation Engineering, Vol 10, No 1, March 2002
[7] Javier San Agustin, Henrik Skovsgaard, John Paulin Hansen, and Dan Witzner Hansen. 2009. Low-cost gaze interaction: ready to deliver the promises. In Proceedings of the 27th international conference extended abstracts on Human factors in computing systems (CHI EA '09). ACM, New York, NY, USA, 4453- 4458. DOI=10.1145/1520340.1520682 http://doi.acm.org/10.1145/1520340.1520682
[8] MyTobii P10 Gaze Tracking System, Tobii Technology AB, S-182 17 Danderyd, Sweden; online: http://www.tobii.com/assistive-technology/germany/products/hardware/mytobii-p10/
[9] Jacob, R. J. K.: Eye-movement-based human-computer interaction techniques: Toward non-command interfaces, in H. R. Hartson \& D. Hix (editors): Advances in Human-Computer Interaction, 1993, Vol. 4, Ablex Publishing Corporation, Norwood, New Jersey, chapter 6, pp. 151-190.
[10] The Intel Open Computer Vision Library: http://www.intel.com/technology/computing/opencv
[11] Christoph Veigl: An Open-Source System for Biosignal- and Camera-Mouse Applications; Submission for the Young Researchers Consortium of the ICCHP 2006, Linz
[12] J. Saragih, S. Lucey and J. Cohn, Deformable Model Fitting by Regularized Landmark Mean-Shifts, International Journal of Computer Vision (IJCV), 2010
[13] The Jelly Bean Switch – a special Assistive Technology Button Switch: http://www.ablenetinc.com/Assistive-Technology/Switches/Jelly-Bean
[14] Clarosoftware, On Screen Keyboard Application (OSKA). 2012. http://www.clarosoftware.com/index.php?cPath=365
[15] Hatakeyama, T., Todoroki, T., Masaki, K., Itoh, J., And Okamoto, A.. 1993. The Single Switch Mouse Emulator For The Severely Physically Disabled, Proceedings of the 9th the Symposium on Human Interface, pp.279-282.