Development of Hands-free Interaction Techniques and Alternative Computer Access Solutions for People with Motor-neuron Impairments

vorgelegt von
M. Sc.
Cagdas Esiyok
ORCID: 0000-0003-0229-0948

an der Fakultät IV – Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
– Dr.-Ing. –

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Odej Kao
Gutachter:	Prof. Dr. Dr. h.c. Sahin Albayrak
Gutachter:	Prof. Dr.-Ing. Matthias Rötting
Gutachterin:	Prof. Dr. Aliye Tosun

Tag der wissenschaftlichen Aussprache: 18. August 2020

Berlin 2020

Intelligence is the ability to adapt to change.

-Stephen Hawking-

Abstract

In our increasingly digitalized world, computers have become indispensable tools by offering several useful services in many aspects of life such as communication, education, commerce, health, social interaction, and entertainment. Unfortunately, most people with motor-impairments have difficulties to access such services because conventional input devices are not concertedly designed for them. Hands-free Computer Accessibility Tools (CATs) help these people to achieve aforesaid useful services for a more inclusive and barrier-free life. However, hands-free computer access is still a challenging task for people with severe motor-impairments of the limbs such as quadriplegics. Especially, when it comes to the people who have only a single voluntary gesture above neck survived (such as blinks or tooth-clicks), hands-free computer access with a single-gesture becomes one of the most challenging tasks in human-computer interaction (HCI).

Through this thesis, we focus on the single-gesture based hands-free computer access problem. The existing HCI solutions on this problem are mostly based on expensive dedicated devices beyond standard computer peripherals. Although the aim of the universal access is enabling equal opportunity by reducing barriers, high-cost of current solutions creates a new barrier financially for the majority of target group. Furthermore, most of the existing single-gesture based hands-free HCI techniques are only compatible with a specific switch-accessible interface. To overcome these deficiencies of the existing HCI solutions, we propose our novel software switch approach. By following the principles of the software switch approach, we also propose four novel software switches which are single-gesture based HCI techniques named as the PuffCam, the PuffMic, the HeadCam, and the HeadGyro. Unlike the existing solutions, our proposed software switches don't require any dedicated devices, and they are compatible with most switch-accessible interfaces.

Although the proposed software switches can allow the users to interact with a computer by a single puff/head gesture, the users require a CAT which is capable of converting the emulated switch presses by software switches into meaningful

commands to operate a computer. In accordance with this requirement, we present a new single-switch accessible CAT called the GLOSTER 1.0 with a novel mouse pointing technique within the scope of thesis.

In addition to the above-mentioned contributions, the inadequacies of the existing evaluation tools promote us to design a novel evaluation tool namely the SITbench 1.0. As a benchmark tool, it is able to serve not only for the proposed software switches but also the other available switch-based interaction techniques (SITs). It provides a quicker and more accurate switch evaluation process by collecting and saving the objective data automatically.

Zusammenfassung

In unserer zunehmend digitalisierten Welt sind Computer zu unverzichtbaren Werkzeugen geworden, da sie in vielen Bereichen des Lebens wie Kommunikation, Bildung und Handel sowie Gesundheit, soziale Interaktion und Unterhaltung verschiedene nützliche Dienste anbieten. Leider haben viele Menschen mit motorischen Beeinträchtigungen Schwierigkeiten, auf solche Dienste zuzugreifen, da konventionelle Eingabegeräte nicht auf sie abgestimmt sind. Freihändige Computerzugangshilfen (Hands-free Computer Accessibility Tools, CATs) helfen diesen Menschen, die oben genannten Dienste zu nutzen und somit ein inklusiveres und barrierefreies Leben in der Computerwelt zu erreichen. Der freihändige Computerzugang ist jedoch immer noch eine Herausforderung für Menschen mit schweren motorischen Beeinträchtigungen der Gliedmaßen, wie z.B. Quadriplegiker. Besonders wenn es um Menschen geht, die mit nur einer einzigen freiwilligen Geste über dem Hals leben (wie Blinzeln oder Zahnklicken), wird der freihändige Computerzugang mit einer einzigen Geste zu einer der anspruchsvollsten Aufgaben in der Mensch-Computer-Interaktion (HCI).

In dieser Arbeit konzentrieren wir uns auf das Problem des freihändigen Computerzugangs mit einer einzigen Geste. Die existierenden HCI-Lösungen für dieses Problem basieren meist auf teuren dedizierten Geräten, die über die Standard-Computerperipherie hinausgehen. Obwohl das Ziel des universellen Zugangs darin besteht, durch den Abbau von Barrieren Chancengleichheit zu ermöglichen, stellen die hohen Kosten der derzeitigen Lösungen für die Mehrheit der Zielgruppe finanziell eine neue Barriere dar. Darüber hinaus sind die meisten der existierenden Single-Gesture-basierten HCI-Freisprechverfahren nur mit einer spezifischen, switch-kompatiblen Schnittstelle kompatibel. Um diese Mängel der bestehenden HCI-Lösungen zu überwinden, schlagen wir unseren neuartigen Software-Switch-Ansatz vor. Wir folgen den Prinzipien des Software-Switch-Ansatzes und schlagen außerdem vier neue Software-Switches vor, die auf Single-Gesture-basierten HCI-Techniken aufbauen und als PuffCam, PuffMic, HeadCam und HeadGyro bezeichnet werden. Im Gegensatz zu den bestehenden Lösungen benötigen unsere vorgeschlagenen Software-Switches keine

dedizierten Geräte und sind mit den meisten Switch-zugänglichen Schnittstellen kompatibel.

Obwohl die vorgeschlagenen Software-Switches den Benutzern die Interaktion mit einem Computer durch eine einzige Puff-/Kopf-Geste ermöglichen können, benötigen die Benutzer einen CAT, der in der Lage ist, die emulierten Switch-Pressen durch Software-Switches in sinnvolle Befehle zur Bedienung eines Computers umzuwandeln. Entsprechend dieser Anforderung stellen wir im Rahmen der Doktorarbeit einen neuen mit einer neuartigen Mauszeigertechnik sowie einem einzigen Switch zugänglichen CAT vor - namens GLOSTER 1.0.

Zusätzlich zu den oben genannten Beiträgen haben uns die Unzulänglichkeiten der bestehenden Evaluierungswerkzeuge dazu bewegt, ein neuartiges Evaluierungswerkzeug, nämlich die SITbench 1.0, zu entwerfen. Als Benchmark-Tool ist es in der Lage, nicht nur für die vorgeschlagenen Software-Switches, sondern auch für die anderen verfügbaren switch-basierten Interaktionstechniken (SITs) zu dienen. Es ermöglicht einen schnelleren und genaueren Bewertungsprozess der Switches, indem es die objektiven Daten automatisch sammelt und speichert.

Acknowledgements

This work would not have been possible without the support, help and encouragement of many people.

First of all, I would like to express my gratitude to my thesis supervisor Prof. Dr.-Ing. Sahin Albayrak for giving me the chance and support to work on my PhD topic at TU Berlin. I also take this opportunity to thank Prof. Dr.-Ing. Matthias Rötting and Prof. Dr. Aliye Tosun, for reviewing my thesis.

In particular I wish to thank Dr. Brijnesh.-J. Jain for giving me constant feedback and professional advice on my research, which ultimately led to improved results.

Likewise, I would like to thank Dr. Frank Hopfgartner and Dr. Fikret Sivrikaya for their precious comments, suggestions and support while forming this work.

In addition, I would like to express my appreciation to the Ministry of National Education Scholarship of the Turkish Republic for funding me during my PhD studies.

Special thanks go to great people who took part in the usability studies through this thesis.

Further, I would like to thank my colleagues from the Information Retrieval and Machine Learning research group at DAI-Labor for their support and for providing a friendly working environment within this period.

I would also like to express my deepest gratitude to my family. I always felt their support through my education.

Last but not least, I would like to deeply thank my wife. All of this would not have been possible without her unconditional support and sweet smile.

List of Publications

A large part of this thesis is based on the following publications by the author:

Journal Articles

Chapters 2 and 3:

- **Cagdas Esiyok**, Ayhan Askin, Aliye Tosun and Sahin Albayrak, "*Software Switches: Novel Hands-free Interaction Techniques for Quadriplegics Based on Respiration–machine Interaction*", Universal Access in the Information Society, 2019.

Chapters 2 and 4:

- **Cagdas Esiyok**, Ayhan Askin, Aliye Tosun and Sahin Albayrak, "*Novel Hands-Free Interaction Techniques based on the Software Switch Approach for Computer Access with Head Movements*", Universal Access in the Information Society, 2020.

Chapter 6:

- **Cagdas Esiyok** and Sahin Albayrak, "*SITbench 1.0: A Novel Switch-Based Interaction Technique Benchmark*", Journal of Healthcare Engineering, 2019.

Other publications by the author outside the scope of this thesis:

Conference Paper

- **Cagdas Esiyok**, Benjamin Kille, Brijnesh.-J. Jain, Frank Hopfgartner and Sahin Albayrak, *"Users' Reading Habits in Online News Portals"*, Proceedings of the 5th Information Interaction in Context Symposium, 2014.

Book Chapter

- **Cagdas Esiyok**, and Sahin Albayrak, *"Twitter Sentiment Tracking for Predicting Marketing Trends"*, Smart Information Systems, 2015.

List of Abbreviations

AAC	Augmentative and Alternative Communication
ALS	Amyotrophic lateral sclerosis
BCI	Brain-Computer Interaction
CATs	Computer Accessibility Tools
CG	Control group
CoorP	Coordinate-based Pointing
DG	Disability group
DS-NDG	Double switch Non-stop Driver Game
EEG	Electroencephalography
EMG	Electromyography
EOG	Electrooculography
FDP	Forefinger distal pulp
FN	False negatives
FP	False positives
FPIJ	Forefinger proximal interphalangeal joint
GDP	Gross Domestic Product
Gyro	Gyroscope
HCI	Human-computer interaction
HFG	Hungry Frog Game
ILO	International Labour Organisation
LED	Light-emitting diode
MEMS	Microelectromechanical system
Mic	Microphone
MMSE	Mini-Mental State Examination
NDG	Non-stop Driver Game
SITs	Switch-based interaction techniques
SS-NDG	Single switch Non-stop Driver Game
SUS	System Usability Scale
TN	True negatives
TP	True positives
TSMG	Tie-Smiley Matching Game
WLAN	Wireless local area network

Table of Contents

List of Tables

List of Figures

1

Introduction

Within the scope of this thesis, we provide comprehensive solutions to single-gesture based hands-free computer access problem for people with severe-motor impairments. In this introductory chapter, we begin with the motivation of the thesis which explains why single-gesture based hands-free computer access is important for people with severe motor-neuron impairments in Section 1.1. Then, the existing problems of the current solutions are stated in Section 1.2. Subsequently, addressed research questions through this thesis are given in Section 1.3. Afterwards, we state the main contributions of this thesis in accordance with the aim of providing comprehensive solutions for single-gesture based hands-free computer access problem. Lastly, Section 1.5 presents the overall structure of the thesis.

1.1 Motivation

Computers have become indispensable tools for the general public, facilitating many essential services in our increasingly digitalized world. Unfortunately, most people with motor-impairments lack these services, since the conventional computer interaction ways such as keyboards are generally inaccessible for them. The ability of operating a computer opens the door for these people to achieve several useful services in many aspects of life such as communication, commerce, health, entertainment, social interaction, and education for a more inclusive and barrier-free life, which leads to an increased quality of life by accessing internet [1]. Hands-free CATs play a vital role in achieving these services [2, 3]. In principle, they can enable the users to operate a computer with their unimpaired physical abilities such as head movements instead of conventional ways. The famous theoretical physicist Stephen Hawking —diagnosed with amytrophic lateral sclerosis (ALS) — is a well-known example of a person who utilized CATs in several aspects. Even after the loss of his mobility and speech, he was still able to communicate and conduct scientific research by means of a CAT with an Augmentative and Alternative Communication (AAC) technology throughout his life.

Unfortunately, Stephen Hawking was not an exceptional case. There are millions of people worldwide who require CATs. It is estimated that there have been about one billion people with several disabilities according to the World Report on Disability in 2011 [4]. Besides, about 2% of the world population —between 110 and 190 million people— have severe disabilities in functioning. Even only in United States, it is predicted that nearly 5.4 million people (between the ages of 18 and 64) live with paralysis in 2013 [5]. People with motor-impairments —as a result of ALS, carpal tunnel syndrome, spinal cord injury or degenerative diseases— require assistive technology solutions to have a more independent life.

On the other hand, according to the International Labour Organisation (ILO) statistics [6] published in 2007, an estimated 470 million of the world's working age people live with several disabilities. Although there have been many jobs that are dependent on computer usage like software coding, exclusion of millions of working age people with disabilities from labour force leads to an increase in the Gross Domestic Product (GDP) lost worldwide. Furthermore, as it is expected, the ones who can perform a paid-job feel more confident and independent both financially and psychologically.

1.2 Problem Statement

Hands-free computer access is a challenging task for people with severe motor-impairments of the limbs such as quadriplegics, since they have serious difficulties to control any body parts under neck. Especially, when it comes to the people who have only one voluntary gesture above neck survived (such as blinks or tooth-clicks), hands-free computer access with a single-gesture becomes one of the most challenging tasks in human-computer interaction (HCI).

In this section, we identify the existing problems under three domains in line with our ultimate goal to provide comprehensive solutions for single-gesture based hands-free computer access problem.

Software Switches: Single-gesture based Hands-free Interaction Techniques

Recent studies on single-gesture based hands-free computer access problem are mostly based on expensive dedicated devices beyond standard computer peripherals. Considering that 80% of people with disabilities accommodate in poor and middle income countries [4], the majority of these people have difficulties to afford most of current solutions [7-9]. Although the aim of the universal access is enabling equal opportunity and access to a service or a product regardless of people's physical disabilities by reducing the barriers, high-cost of current solutions creates a new barrier financially for the majority of target group.

The other problem is that the majority of the current single-gesture based hands-free HCI techniques are only compatible with specific switch-accessible interfaces. While some switch-accessible interfaces expect to receive a specific keyboard character like *enter*, the others expect to receive a mouse click. There is not any commonly agreed standard on this.

The GLOSTER 1.0: A Single-gesture Accessible Hands-free CAT

Although software switches proposed within this thesis can enable to interact with a computer by a single-gesture, a CAT is required to control a computer. As stated in Section 1.1, the CATs are capable of providing several useful services for the ones who cannot control a computer with conventional ways. Typing and clicking are generally achieved by using switches via a scanning method, while mouse pointing is generally performed by head or eye tracking methods. But, if the user has an only single-gesture unimpaired, all three functions have to be performed with a single-gesture/single-switch. Especially, single-switch based mouse pointing is a very challenging task in CATs.

The SITbench 1.0: An Evaluation Tool

Evaluation process of a switch-based interaction technique (SIT) —like the proposed software switches— requires an interdisciplinary team effort and takes a considerable amount of time, since a SIT setup depends on many variables such as switch type or switch site. Although collecting subjective evaluation data from the users is a very common approach, we considered that the subjective evaluation data alone might be manipulated and unreliable for comparing SIT performances in many cases. Because it is hard to evaluate the measurable performance by collecting subjective data instead of objective data, determining the optimum SIT setup (i.e., the most appropriate combination of setup variables) could not be achieved at first attempts.

On the other hand, although collecting objective data is the most appropriate method for performance evaluation, the existing objective evaluation methods in literature are far from being a benchmark. They are mostly designed to evaluate just a specific SIT, which makes them ineligible to be a benchmark where the other SITs could be evaluated via standardized test.

1.3 Research Questions

In this section, we present the research questions addressed through this thesis.

Software Switches: Single-gesture based Hands-free Interaction Techniques

To deal with the related problems stated in Section 1.2, we first focus on the following research question:

Q1: How to devise an efficient approach enabling single-gesture based hands-free HCI?

Then, we address the following research questions according to the most suitable hands-free gestures identified by us.

Q2: How to devise a better technique that enables a person to interact with a computer by a single puff-gesture?

Q3: How to devise a better technique that enables a person to interact with a computer by a single head-gesture?

The GLOSTER 1.0: A Single-gesture Accessible Hands-free CAT

In line with the aim of controlling a computer via the proposed interaction techniques, we address the following research question:

Q4: How to devise a better CAT that enables a person to control a computer with a single-gesture?

The SITbench 1.0: An Evaluation Tool

In the evaluation stage of the proposed interaction techniques, the requirement of an evaluation tool leads us to focus on the following research question:

Q5: How to devise a better tool that enables objective evaluation of switch-based interaction techniques?

1.4 Thesis Contributions

Our contributions through this thesis and related publications are given in this section under three domains.

Software Switches: Single-gesture based Hands-free Interaction Techniques

In accordance with our efforts to find an efficient solution to single-gesture based hands-free computer access problem, we start with a literature review of the current hands-free interaction techniques —in terms of the gestures used— to identify the current problems. To overcome the related existing problems stated in Section 1.2, we propose our novel software switch approach following our efforts to answer the research question Q1. To sum up, our software switch approach has two principles: an interaction technique based on software switch approach (1) should not require any dedicated devices, and (2) should be configurable to be compatible with the other switch-accessible interfaces. We also identify the most suitable hands-free gestures as puff and head gestures in accordance with the principles of our software switch approach. These gestures are then employed to interact with a computer via the proposed software switches.

Then, the research questions Q2 and Q3 lead us to devise four novel software switches —called the PuffMic, the PuffCam, the HeadCam, and the HeadGyro— by following the principles of our software switch approach. Two usability studies —conducted with 82 participants in total— demonstrate that the proposed software switches can allow interacting with a computer by a single-gesture in a way that they receive the user's gesture as an input signal to translate them into emulated switch presses. While the PuffMic and the PuffCam are based on puff-gesture, the HeadCam and the HeadGyro depend on head-gestures (e.g., a head tilt). The PuffMic and the PuffCam software switches were presented in the following article:

- **Cagdas Esiyok**, Ayhan Askin, Aliye Tosun and Sahin Albayrak, "*Software Switches: Novel Hands-free Interaction Techniques for Quadriplegics Based on*

Respiration–machine Interaction", Universal Access in the Information Society, 2019.

The following article is partly based on the HeadCam and the HeadGyro software switches:

- **Cagdas Esiyok**, Ayhan Askin, Aliye Tosun and Sahin Albayrak, "*Novel Hands-Free Interaction Techniques based on the Software Switch Approach for Computer Access with Head Movements*", Universal Access in the Information Society [under review].

The GLOSTER 1.0: A Single-gesture Accessible Hands-free CAT

As a result of focusing the research question Q4, we present a new single-switch accessible CAT called the GLOSTER 1.0 with a novel mouse pointing technique called the Coordinate-based Pointing (CoorP). It allows the users to easily control a computer with a single-gesture by employing the proposed software switches. By means of the proposed CAT, emulated switch presses by software switches can be converted into meaningful commands to operate a computer. As an all-in-one solution, the GLOSTER 1.0 provides all three functions —pointing, clicking, and typing— performed by a mouse and a keyboard. Following a usability study with 20 participants, it is revealed that the CoorP performed better than the CrossHair which is the most preferred technique by the existing CATs.

The SITbench 1.0: An Evaluation Tool

Within the development process of the proposed software switches in accordance with the research questions Q2 and Q3, during evaluation step, we encounter the related problems identified in Section 1.2. Therefore, the research question Q5 leads us to propose a novel benchmark tool for performance evaluation called the SITbench 1.0. It is demonstrated by a usability study with 8 participants that the SITbench 1.0 provides a quicker and more accurate switch evaluation process by collecting the objective data automatically. Part of the SITbench 1.0 was published in the following article:

- **Cagdas Esiyok** and Sahin Albayrak, "*SITbench 1.0: A Novel Switch-Based Interaction Technique Benchmark*", Journal of Healthcare Engineering, 2019.

1.5 Thesis Structure

The rest of this thesis is structured as follows:

In Chapter 2, we propose the software switch approach enabling single-gesture based hands-free HCI. In addition, we also identify the most suitable gestures to interact with a computer in line with the software switch approach. In Chapters 3 and 4, we present four novel interaction techniques —called the PuffMic, the PuffCam, the HeadCam, and the HeadGyro software switches— by following the principles of our software switch approach. While Chapter 3 introduces the PuffMic and the PuffCam software switches, which are based on a single puff-gesture, Chapter 4 presents single head-gesture based software switches namely the HeadCam and the HeadGyro. In Chapter 5, we introduce a new single-switch accessible CAT —namely the GLOSTER 1.0— with a novel mouse pointing technique which allows to easily control a computer by employing the proposed software switches. In Chapter 6, a novel benchmark tool for performance evaluation —called the SITbench 1.0— is proposed to provide a quicker and more accurate switch evaluation process by collecting the objective data automatically. Lastly, Chapter 7 summarizes and discusses the main contributions and provides an outlook to future research directions.

2

The Software Switch Approach

In line with our efforts to find an efficient solution to single-gesture based hands-free computer access problem, we begin with a literature review of the current hands-free interaction techniques in terms of the gestures used. Following the literature review, we identify two major problems of current single-gesture based hands-free interaction techniques: (1) the majority of current single-gesture based hands-free HCI techniques depend on dedicated devices beyond standard computer peripherals; (2) current single-gesture based hands-free solutions in literature are only compatible with specific switch-accessible interfaces. To overcome these problems, we propose our novel software switch approach. Briefly, the software switch approach has two principles: an interaction technique based on software switch approach (1) should not require any dedicated devices, and (2) should be configurable to be compatible with switch-accessible interfaces. In this chapter, we also identify the most suitable hands-free gestures as puff and head gestures in accordance with the principles our software switch approach. These gestures are then employed to interact with a computer via the proposed software switches in Chapters 3 and 4.

2.1 Related Works

Many hands-free solutions which are mostly based on dedicated devices (e.g., switches, sensors) or just standard computer peripherals (e.g., a camera) have been developed up to now for computer access. People with motor-impairments are able to interact with these devices by using their unimpaired body gestures like head movements or eye-blinks. In this section, to look from a broader perspective, we review hands-free solutions —instead of focusing just the single-gesture based hands-free solutions— that provide alternative means for computer access in terms of the body gestures used. We separated them into two main groups according to the condition whether any dedicated hardware is required except for standard computer peripherals.

- **Hands-Free HCI Techniques with Dedicated Devices:** These systems require additional dedicated sensors, switches or devices beyond standard computer peripherals to control a computer such as Brain-Computer Interaction (BCI) based systems via Electroencephalography (EEG) sensors [10, 11]; eye movement operated systems based on Electrooculography (EOG) sensors [12, 13] or dedicated cameras [14, 15]; head movement operated systems based on traditional switches [16], inertial sensors [17, 18] or special cameras [19, 20]; sip-and-puff operated systems [21-23]; facial mimics operated systems based on Electromyography (EMG) sensors [24, 25]; tongue operated systems [26, 27]; tooth-click operated systems [28]; mouth/lip joystick operated systems [29-31]; and chin operated systems [32-34].
- **Hands-Free HCI Techniques with Standard Computer Peripherals:** These systems can be divided into two main groups as camera and microphone based systems. In camera based systems, a camera is set to focus on eye-gaze [35] or head movements [36-38] to transform them into mouse cursor movements on a computer screen. Facial gestures (such as eye blink or eyebrow-raising) can also be captured by a camera to trigger a mouse click or a key stroke [39, 40]. On the other hand, the majority of microphone based hands-free interaction techniques for computer access depend on speech gestures as commands where speech recognition techniques are applied to control a mouse pointer [41, 42] or a

keyboard [43]. Similarly, non-speech gestures such as humming, can be recognized by speech recognition algorithms to control a mouse pointer [44, 45] or a keyboard emulator [46]. Additionally, puff as a respiration-gesture can also be detected by a microphone and employed for clicking task in computer access [47].

2.2 Existing Problems

Following a literature review in Section 2.1, having a broader point of view helped us to recognize the existing problems. We identified two major problems of the current single-gesture based hands-free interaction techniques:

- **Requirement of Dedicated Devices**: The majority of current single-gesture based hands-free HCI techniques depend on dedicated devices beyond standard computer peripherals. Besides, The World Report on Disability reveals that 80% of people with disabilities accommodate in poor and middle income countries [4], which means that most of these people might have serious difficulties to afford dedicated devices [7-9]. Although the aim of the universal access is enabling equal opportunity and access to a service or product regardless of people's physical disabilities by reducing barriers, high-cost of dedicated devices creates a new barrier financially.
- **Compatibility with Switch-accessible Interfaces**: Current single-gesture based hands-free solutions in literature are only compatible with specific switch-accessible interfaces. To make it clear, first the mechanism of a scanning-based interface and standardization problem should be understood. In principle, unlike direct selection (such as typing on a keyboard), the scanning-based interface highlights items one-by-one on the computer screen, and the user activates the switch when the desired item is highlighted. Between switch-accessible interface and the switch, there is a switch adapter which is a dedicated device to transform switch activation signals into meaningful keyboard presses or mouse clicks. Following a switch activation, switch adapter emulates a specific keyboard character or a mouse click event (depending on the manufacturer of

switch interface) and send it to the computer in order to communicate with switch-accessible interface. But the main problem here is that there has not been any commonly agreed standard for the communication between switches and switch-accessible interfaces; while some switch-accessible interfaces expect to receive a specific keyboard character like space, the others expect to receive a mouse click. This standardization problem is partially solved by a switch driver software permitting the users to assign a specific character or mouse click — following a switch activation— which is expected by the target switch-accessible interface. However, these switch driver software are only compatible with a limited number of switch adapters of specific brands, which makes them partial solutions for the standardization problem. In other words, each switch adapter requires its specific switch driver software. To the best of our knowledge, there is not any complete solution for this standardization problem in literature.

2.3 Principles of the Software Switch Approach

To overcome the problems identified in Section 2.2, we propose our software switch approach. Two principles of the software switch approach are presented below according to related problems:

- **Requirement of Dedicated Devices**: As the first principle of our software switch approach, any interaction techniques based on our approach should not require any dedicated device beyond standard computer peripherals like a microphone or a camera. At this point, as the only reasonable exception, we decided to exclude smartphones from dedicated devices list; because the total number of smartphones —3.2 billion in 2019 [48]— got ahead of the total number of computers in recent years worldwide [49], which makes them easy to access for people in even low-income countries. Besides, smartphones are able to provide several services to the users unlike dedicated devices which are produced with a specific aim.
- **Compatibility with Switch-accessible Interfaces:** As the second principle of our software switch approach, any interaction techniques based on our approach

should be configurable to generate any expected keyboard characters or mouse clicks, which makes them compatible with most switch-accessible interfaces. In this way, they can provide a better solution to the standardization problem than the current solution where a switch driver and a traditional switch are required to purchase. In other words, they are able to both detect a single-gesture like a traditional switch and allow the users to assign the expected keyboard characters or mouse clicks —which will be sent to the switch-accessible interface following a detected single-gesture— like a switch driver.

To sum up, our software switch approach has two principles: an interaction technique based on software switch approach (1) should not require any dedicated devices, and (2) should be configurable to be compatible with switch-accessible interfaces.

2.4 Gesture Selection

Although the software switch approach is flexible enough to be employed with any physical gesture, within this thesis, we focus on the HCI techniques based on hands-free gestures to provide comprehensive computer access solutions for the ones who are only able to move from the neck up. To select the proper types of hands-free gestures which will be employed by the proposed software switches, we begin with the evaluation of gesture types of current hands-free interaction techniques given in Section 2.1.

HCI techniques which require dedicated devices are left out of the focus in accordance with the first principle of our software switch approach. Then, we exclude eye-gaze and facial gestures (e.g., eye blink gesture), since they might be highly affected by Midas Touch problem [50]. For example, if the eye blink gesture is used to interact with a computer, it is hard to distinguish whether the user blinked consciously as a trigger signal or it was just a regular eye blink performed unconsciously. Although the Midas Touch problem could be resolved by employing multi-modal inputs such as performing an eye blink with an eye brow raising simultaneously as a trigger signal, employing a multi-modal input is not an option in the scope of this thesis since we aim to solve the single-gesture based computer access problem. As another gesture types which don't require a dedicated device, the speech and non-speech gestures are also omitted, because the speech recognition is mostly preferred for more complicated tasks than just a single-gesture recognition. Unlike these gestures, we revealed that the respiration (i.e., sip and puff) and head movement (e.g., a head tilt) gestures are not affected by Midas Touch problem, and they can be easily recognized via simple algorithms. Therefore, we considered them as the most suitable hands-free gesture types among existing solutions to interact with a computer by following the principles of our software switch approach.

2.5 Summary

In this chapter, first we begin with a literature review of the current hands-free interaction techniques in terms of the gestures used. Then, two major problems of current techniques are identified.

To overcome the identified problems, we propose the software switch approach and detect the most suitable gestures. Afterwards, we also propose four novel software switches which are single-gesture based human-computer interaction (HCI) techniques —namely the PuffCam, the PuffMic, the HeadCam, the HeadGyro— by following the principles of our software switch approach. Unlike the existing solutions, the proposed software switches don't require dedicated devices and compatible with most switch-accessible interfaces.

3

The PuffMic and the PuffCam:

Novel Interaction Techniques Based on a Single Puff-Gesture

In this chapter, we focus on how to devise a better method that enables a person to interact with a computer by a single puff-gesture. For the ones with a very limited motor activity but a complete respiration activity, interacting with a computer by a single puff-gesture is considered a challenging task. To overcome this challenge, we propose two novel interaction techniques as software switches —the PuffCam and the PuffMic— for single-gesture based CATs by following the principles of our software switch approach. Both software switches are respiration operated where a strong puff, detected non-invasively by a microphone or a modified camera, is considered as a pressed switch. A usability study —conducted with 46 participants with/out disabilities— reveals that the accuracy, precision, recall and false positive rate of our interaction techniques are quite impressive, and the PuffCam performs better than the PuffMic for all metrics. According to questionnaire findings, comfort assessment of interaction techniques by participants is rated quite satisfactory.

3.1 Introduction

As stated in Section 1.1, there have been millions of motor-impaired individuals worldwide who have difficulties to operate a computer via conventional ways [4, 5]. For the ones with a very limited motor activity but a complete respiration activity, interacting with a computer by a single respiration-gesture is considered as a challenging task for computer access.

The majority of current solutions addressing this challenge is based on invasive sip-and-puff devices [21, 23, 51, 52]. They are capable of recognizing the users' sip and puff gestures to interact with a computer. Recognition of these gestures is performed invasively in a way that the users inhale (sip gesture) or exhale (puff gesture) through a tube/straw which is placed in his/her mouth. Then, air pressure originated from the user's sip or puff is recognized by the device to serve as a double input switch for switch-accessible interfaces like CATs. In literature, as a different approach, we were able to find just one solution called the BlowClick [47] which doesn't require a dedicated device. It provides a non-invasive solution in a way that the users' puff gesture is recognized by a computer after the user puffs on a standard microphone connected to the computer.

Although current respiration-based HCI solutions are useful to enhance people's quality of life in many aspects, we identify two main problems of the existing solutions below to be handled. To overcome these problems, we propose two novel HCI techniques based on the user’s puff gesture —called the PuffMic and the PuffCam software switches— by following the principles of our software switch approach. Two problems of current interaction techniques identified and how we address them by applying our software switch approach are explained below:

- Requirement of Dedicated Devices: While the BlowClick [47] doesn't require any dedicated device; sip-and-puff devices are commercial dedicated hardware beyond standard computer peripherals, which makes them hard to afford for people with low-income. Furthermore, the users might have hygiene problems with sip-and-puff devices, since a tube between mouth and environment might cause hygiene risks [53, 54]. In accordance with the first principle of our software switch approach, both proposed software switches depend on just standard computer peripherals (a microphone or a camera) like the BlowClick.

- Compatibility with Switch-accessible Interfaces: Sip-and-puff devices are only compatible with a specific group of switch-accessible interfaces, while the BlowClick is not compatible with any switch-accessible interface. In other words, the Blowclick is unable to serve as a switch for any other switch-accessible interface. At this point, the PuffMic differs from the BlowClick as being configurable to be compatible with switch-accessible interfaces, although their input method is similar. In line with the second principle of our software switch approach; the PuffMic and the PuffCam software switches are configurable to generate any expected keyboard characters or mouse clicks, which makes them compatible with most switch-accessible interfaces.

To sum up, the proposed solutions are based on the assumption that individuals' strong puffs can be distinguished by employing a standard microphone (the PuffMic) or a modified webcam (the PuffCam) via the proposed software switches running on a standard computer. In this way, a puff-gesture captured by a standard microphone or a webcam is considered as a pressed switch to make a selection in a switch-accessible software like a CAT.

We conducted a usability study with 46 individuals (23 motor-impaired, 23 able-bodied) to collect objective and subjective data by employing the SITbench 1.0 [55] (described in Chapter 6) and a five-point Likert scale questionnaire, respectively. As a result of usability study, the PuffCam method showed better performance than the PuffMic method in all conditions. Accuracy, precision, recall and false positive rate of both interaction techniques were found quite impressive. Moreover, comfort assessment results of a five-point Likert scale questionnaire were satisfactory. The idea to control a computer via breathing without purchasing any dedicated device was considered very promising by all participants.

Studies within this chapter are expected to stimulate new studies by motivating researchers to place a greater emphasis on non-invasive respiration methods since the non-invasive voluntary respiration is an underrated activity for computer access in comparison with invasive systems. The proposed interaction techniques can be used instead of traditional invasive sip-and-puff devices in many cases. Considering that 80% of the people with disabilities live in low and middle income countries [4], the proposed software switches can help to meet the cost-free switch requirements of individuals with motor-impairments worldwide in an open access manner without any additional device. There is not any other alternative solution as a switch currently for the ones who have only respiration activity and cannot afford any dedicated device. They can be integrated with any assistive systems where voluntary respiration is utilized. For example, the users can operate a wheelchair. Similarly, an assistive living system might be developed where the users could interact with smart home devices. On the other hand, the application areas of the proposed interaction techniques can be quite flexible and should not be considered just for assistive technology area. For example, it is possible to utilize them in the entertainment area like computer game industry beyond the assistive technology area.

This chapter proceeds with introducing the proposed software switches in Section 3.2. Then, we evaluate both interaction techniques by presenting objective and subjective evaluation results of our usability study in Section 3.3. Finally, we conclude and discuss our study in Section 3.4.

3.2 Software Switches

In this section, first we introduce the user interface of the PuffMic and the PuffCam. Then, we present both software switches.

3.2.1 The User Interface

Figure 3.1 illustrates the interface of both proposed software switches during a puff activity. It includes a puff meter where the green bars on it show the puff level of the user, and the middle red bar represents the threshold level detected during calibration step (calibration steps of each software switches are explained in Sections 3.2.2 and 3.2.3.). The puff level depends on how strong the user puffs. In principle, if the puff level exceeds the threshold value, a switch press is emulated. The user can monitor the puffs on run-time via puff meter, which helps the user to estimate how strong s/he needs to exhale.

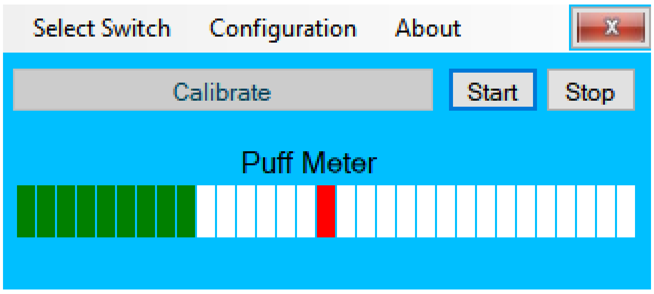


Figure 3.1: The user interface of the PuffMic and the PuffCam.

The interface can be configured according to the selected software switch. For both the PuffMic and the PuffCam, any keyboard character or mouse click can be assigned in configuration for the target CAT. For example, if the CAT waits to receive *enter* character, the user assigns *enter* character to be sent via our software switches. Following a strong puff detected, the proposed software switches send *enter* character to the target CAT. This way, the proposed software switches let the users to control any compatible switch-accessible software. If the PuffCam is selected to be configured, the colour of the tracked object is also assigned by the user. Both software switches were developed under .NET 4.5 framework, and they are compatible with Windows-based operating systems.

3.2.2 The PuffMic

For the first interaction technique proposed, a standard microphone is placed under the user's nose or in front of the mouth in a way that puff-gestures can be captured easily (Figure 3.2). Following positioning, the threshold level needs to be calibrated since the strength of puff activity varies by person.

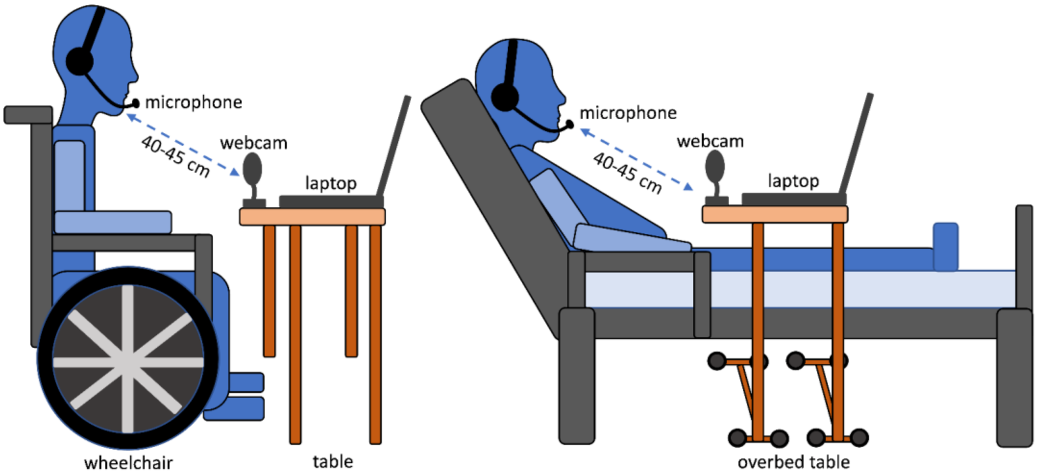


Figure 3.2: Camera, laptop and microphone positions during the experiments.

The user first adjusts the scanning time of the software switch less than the scanning time of the target switch-accessible software. For example, if the scanning time of the target CAT is 2 seconds (i.e., if the target CAT is able to receive the trigger signal in every 2 seconds), the user should assign the scanning time less than two seconds such as 0.9 second, and thus the software switch becomes capable of sending the expected trigger signal to the CAT in every 0.9 second. Then, the user assigns the smoothing time to be applied on the audio signal during signal smoothing process.

Following the assignment of scanning and smoothing times, the software switch records an audio wave for assigned scanning time period as illustrated in Figure 3.3. Within this time period, the user performs a strong puff on a microphone in order to set a threshold value. In this example, scanning time is considered as 0.9 second, while the smoothing time is set as 0.1 second.

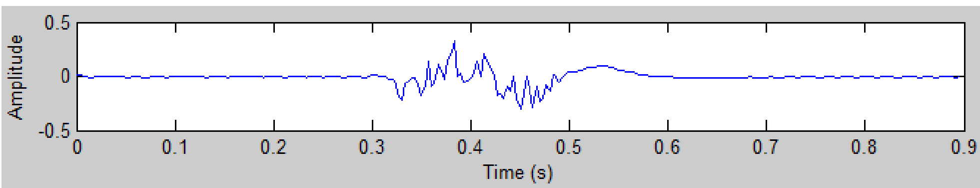


Figure 3.3: Raw audio signal.

Then, the software switch gets the absolute value of audio signal (Figure 3.4).

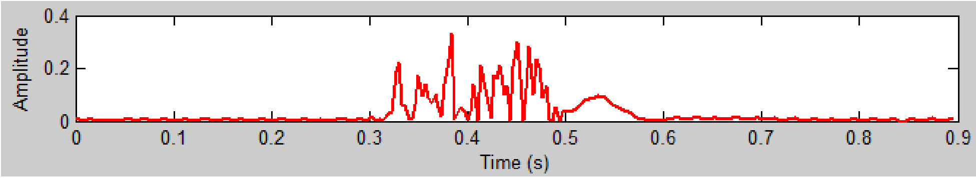


Figure 3.4: Absolute value of audio signal.

Afterwards, the software switch extracts the smoothed audio signal by calculating the average value for each smoothing time period (i.e., for each 0.1 second) as can be seen in Figure 3.5. The peak puff level (between the 0.4 and 0.5 seconds) is assigned as the threshold value. Then, for the visual feedback, the peak puff level is represented by the middle red bar in the puff meter (Figure 3.1) as a threshold indicator.

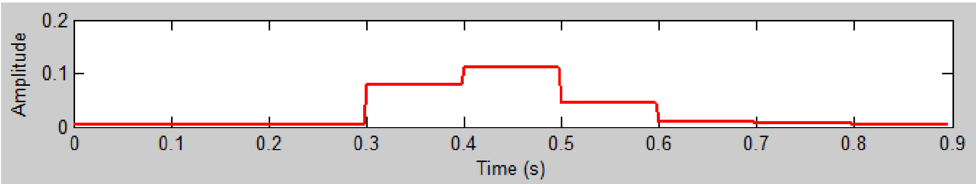


Figure 3.5: Smoothed audio signal.

Following a successful calibration, the PuffMic can serve as a puff switch for a switch-accessible software. To do this, the user's respiration activity is tracked in real-time. Like in calibration step; (1) first the absolute value of the audio signal is extracted. (2) Then, signal smoothing is applied to calculate the peak puff level depending on the scanning time and the smoothing time. (3) The peak puff level is represented as green bars in the puff meter according to the strength of the puff. (4) Lastly, if the peak puff level exceeds the threshold value following a strong puff, a switch press is emulated by the PuffMic to send a trigger signal to the target CAT.

3.2.3 The PuffCam

The PuffCam software switch basically translates the motion of an object —as a result of a strong puff— captured by a webcam into a trigger signal for a switch-accessible software. A modified webcam with a post-it (or a piece of paper) that is placed just to the opposite of the camera lens by means of an adhesive tape is employed as can be seen in Figure 3.6. The only requisite of this modification is that the object to be tracked (i.e., a black rectangle figure drawn on post-it in our example) should be kept in the visual field of the camera in any case, even if the user puffs very strongly on the post-it.

Figure 3.6: A standard webcam modified with a post-it.

Following a proper positioning as it is illustrated in Figure 3.2, the PuffCam should be calibrated. The user first adjusts the scanning time of the software switch less than the scanning time of the target switch-accessible software like adjusted in PuffMic. Then the real-time video motion tracking algorithm is employed as follows:

- For scanning time period assigned, video frames are taken by a webcam with a frame rate of 15 frames per second and a frame size of 320x240 pixels (Figure 3.7 (a));
- Euclidean colour filtering is applied for each video frames according to colour of the target object assigned during configuration (Figure 3.7 (b));
- Video frames are converted to grayscale following Euclidean colour filtering (Figure 3.7 (c));
- Object detection is performed on video frames via connected-component labeling method. (Figure 3.7 (d));

- The position of the detected object on post-it is tracked through run time (Figure 3.7 (e)).
- Within this scanning time period, the user puffs strongly on the post-it to change the position of the detected object.
- The difference in pixels between the position prior to puff and the position during strong puff is considered as threshold value, and this threshold value is represented by the middle red bar of the puff meter in Figure 3.1 as a visual indicator.

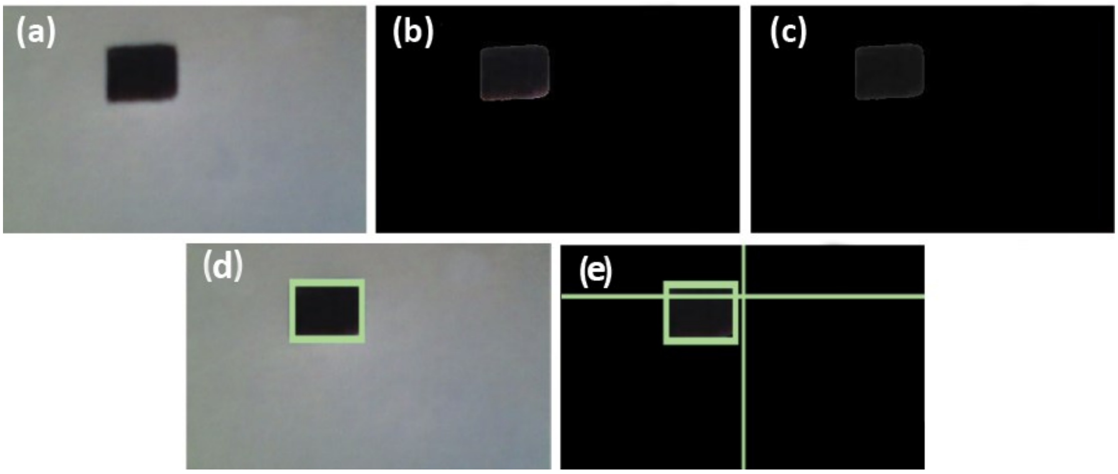


Figure 3.7: Main steps of motion tracking algorithm. (a) take video frames via camera; (b) apply Euclidean colour filter for each frame; (c) convert video frames to grayscale; (d) detect the object; (e) track the position of the detected object.

For filtering and object detection, image processing library called AForge.NET was employed. Following a successful calibration (i.e., assignment of the threshold value), the PuffCam becomes capable of recognizing the strong puffs to serve as a puff switch. To do this, every motion of the detected object is calculated for each scanning time period and then represented as green bars in the puff meter depending on the motion level. If the user puffs strongly enough, threshold value is exceeded, and the PuffCam sends a trigger signal to the target CAT.

3.3 Evaluation

We conducted experiments in Turkey at Medical Faculty of Izmir Katip Celebi University in order to evaluate our interaction techniques by collecting objective and subjective data. The study had been approved on 13.09.2017 by the Ethical Committee of Izmir Katip Celebi University with a reference number of 179. The informed consent form was signed by all participants prior to the experiments.

In this section, firstly we introduce the participants who are separated into two main groups as able-bodied and motor-impaired individuals. Thereafter, we present the apparatus employed throughout the experiments and the procedure to be applied on two novel interaction techniques by means of our evaluation software the SITbench 1.0. Lastly, the experimental findings are shared.

3.3.1 Participants

Overall, 46 participants including 24 females and 22 males took part in this study. Of all participants, 23 (8 females, 15 males) had motor disabilities (hereafter: disability group (DG)) whose ages ranged between 17 and 78. Two of them can be seen in Figure 3.8. The participants without disabilities (hereafter: control group (CG)) included 23 people (16 females, 7 males) whose ages ranged between 17 and 71. Age statistics of all participants are summarized in Table 3.1. All participants with motor disabilities, who volunteered to test our interaction techniques, had difficulties controlling their hands. They were all receiving hospital treatment for several motor disabilities while the experiments were conducted. Just five of the participants had previous experience with switch interfaces. The majority of the participants of CG were volunteers responding to the call made by the Physical Medicine and Rehabilitation Department as well as the relatives and friends of DG. The main characteristics of all participants are listed in Table 3.2.

Figure 3.8: Two participants previous to the experiments.

Table 3.1: Age statistics of the participants according to the groups.

Groups	Gender	Mean Age	Number of Participants
Mix	Mix	47.5 (sd = 17.4)	46
Mix	Female	42.2 (sd = 16.2)	24
Mix	Male	53.3 (sd = 17.1)	22
DG	Mix	52.9 (sd = 19.3)	23
DG	Female	45.8 (sd = 24.2)	8
DG	Male	56.6 (sd = 15.7)	15
CG	Mix	42.3 (sd = 13.7)	23
CG	Female	40.5 (sd = 10.9)	16
CG	Male	46.3 (sd = 19.1)	7

Table 3.2: Main characteristics of the participants.

The User	Age	Gender	Disability
DG1	75	Male	Hemiplegia
DG2	21	Female	Hemiplegia
DG3	65	Male	Hemiplegia

DG4	27	Male	Tetraplegia
DG5	58	Male	Hemiplegia
DG6	57	Male	Hemiplegia
DG7	31	Female	Hemiplegia
DG8	17	Female	Neuromyelitis Optica
DG9	34	Female	Hemiplegia
DG10	63	Female	Guillain-Barre Syndrome
DG11	77	Female	Hemiplegia
DG12	38	Male	Hemiplegia
DG13	46	Female	Hemiplegia
DG14	77	Male	Hemiplegia
DG15	67	Male	Hemiplegia
DG16	34	Male	Tetraplegia
DG17	75	Male	Hemiplegia
DG18	58	Male	Hemiplegia
DG19	53	Male	Hemiplegia
DG20	65	Male	Hemiplegia
DG21	38	Male	Hemiplegia
DG22	62	Male	Hemiplegia
DG23	78	Female	Hemiplegia
CG24	27	Male	None
CG25	59	Female	None
CG26	25	Female	None
CG27	32	Male	None
CG28	41	Female	None
CG29	50	Female	None
CG30	28	Female	None
CG31	40	Female	None
CG32	34	Female	None
CG33	45	Female	None
CG34	45	Female	None

CG35	58	Male	None
CG36	52	Male	None
CG37	71	Male	None
CG38	49	Female	None
CG39	49	Female	None
CG40	32	Female	None
CG41	40	Female	None
CG42	17	Female	None
CG43	51	Female	None
CG44	43	Female	None
CG45	62	Male	None
CG46	22	Male	None

All participants were required to meet the following criteria in order to evaluate our interaction techniques. All participants are supposed to be able:

- to breath voluntarily;
- to find a target on a grid;
- to follow a moving target;
- to maintain gaze on a stable target;
- to stay focused on tests during experiments.

The Mini-Mental State Examination (MMSE), which is a 30-point questionnaire to assess cognitive impairments in clinical studies, was applied to all participants in order to validate whether they fulfil the cognitive ability related requirements stated above prior to the experiments.

3.3.2 Apparatus

The test apparatus consists of a laptop Lenovo G505S (CPU: AMD A8-4500M 1.9 GHz; RAM: 6 GB DDR3; Screen: LCD 15.6; OS: Windows 10 64 bits; Resolution: 1600 ×

900), a Digicomm Headset 9088 (Impedance: 32 ohms at 1kHz; Sensitivity: 105 dB / mW; Frequency range: 8 ~ 22,000Hz) and an A4Tech ViewCam pro PK-635M (Max Digital Video Resolution: 640 x 480; Image Sensor Type: 0.35 MP CMOS; Horizontal Field of View: 54).

3.3.3 Procedure

First of all, we ensured that the participants and devices (i.e., microphone, webcam, laptop) were positioned properly as illustrated in Figure 3.2 before starting the experiments, since a good positioning allows:

- the stability to enhance motor functions;
- easing abnormal reflexes due to voluntary movements;
- being compatible with long-term sessions;
- preventing a significant increase in abnormal muscle tone.

A headset-type standard microphone is located as seen in Figure 3.2 in a way that the system can recognize the strong puffs. In our experiment, we placed the microphone approximately 5 cm ahead of the user's mouth. On the other hand, the webcam should be positioned in a way that the user can move the post-it easily by puffing on it. The distance between webcam and the user's mouth, which depends on the properties of attached post-it or paper (i.e., size, hardness, etc.), was adjusted as about 40-45 cm. in our experiments.

After providing a good positioning, information was given to the participants about the test, and we conducted some trials in counterbalanced order until they understand the concept and become ready for tests. This training process lasted for approximately ten minutes for each participant.

Following positioning and training steps, we applied the first prototype of the Tie-Smiley Matching Game (TSMG) test of the SITbench 1.0 described in Chapter 6 to collect the objective evaluation data. Each proposed software switch was tested by each participant (n = 46) with the first three templates of TSMG where scanning time was

1000 milliseconds. Tests were applied in counterbalanced order to avoid interaction effects due to learning and fatigue. We gave participants some time (1 to 5 minutes) to rest during experiments so as to prevent excessive mental or physical fatigue.

The quantitative subjective data of the proposed interaction techniques was collected by applying a comfort assessment questionnaire containing four statements with a five-point Likert scale (Figure 3.9) after completing tests. The rating ranged from 1 to 5 (1 = strongly disagree, 2 = disagree, 3 = neither agree nor disagree, 4 = agree, 5 = strongly agree) for the comfort assessment questionnaire during experiments. In order to collect the qualitative subjective data, we received responses of open-ended questions and feedback by participants about interaction techniques in addition to our observations.

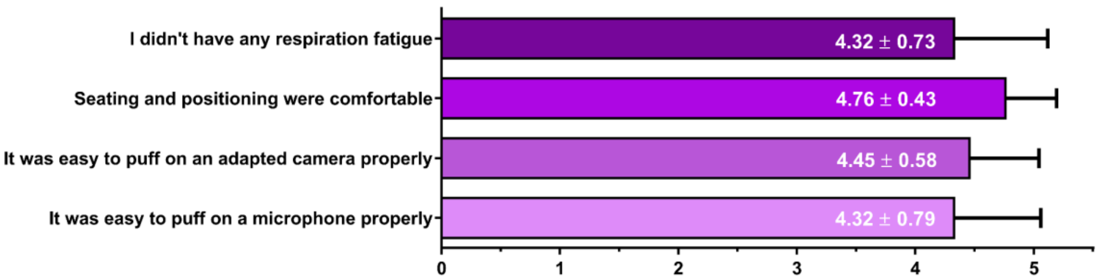


Figure 3.9: The comfort assessment questionnaire with mean values (x-axis represents the rating range).

3.3.4 Objective Data based Results

Mean values of the proposed software switches through evaluation metrics (accuracy, precision, recall and false positive rate) for all participants are shown in Figure 3.10. For all evaluation metrics, the PuffCam showed better performance. Results presented in Figure 3.10 are listed below:

- In terms of accuracy mean values, the PuffCam showed better performance with a mean of 0.926 in comparison to the PuffMic (m = 0.868). According to the

Student's t-test for both techniques, p was found less than 0.05, which means that difference among means in terms of accuracy is statistically significant;

- For precision mean values, the ranking was the same as accuracy: the PuffCam (m = 0.885) and the PuffMic (m = 0.859). As a result of the t-test for both techniques, p was determined greater than 0.05, which means that the difference among means in terms of precision is not statistically significant;
- Regarding recall mean values, the PuffCam came first (m = 0.897) and the PuffMic followed it (m = 0.806). In consequence of the t-test for both techniques, p was calculated as less than 0.05, which means that the difference among means in terms of recall is statistically significant;
- The PuffMic (m = 0.089) was followed by the PuffCam (m = 0.065) based on false positive rate. According to the t-test for both techniques, p was detected as greater than 0.05, which means that the difference among means in terms of false positive rate is not statistically significant.

Mean values of each interaction techniques depending on participant groups (Mix, DG, CG) are presented in Figure 3.11. In terms of mean values of accuracy, precision and recall evaluation metrics, CG members performed better than DG members for both software switches except for the PuffMic in recall. Regarding false positive rate score, DG had higher scores than CG for both interaction techniques, which means that DG members made false selections more frequently when compared to CG members. We applied the Student's t-tests for both interaction techniques through all evaluation metrics to see whether there is a significant difference between the performance of DG members and CG members. The difference among means between DG and CG was found significant in three conditions where: (1) metric type = accuracy, interaction type = the PuffCam; (2) metric type = precision, interaction type = the PuffMic; (3) metric type = false positive rate, interaction type = the PuffMic.

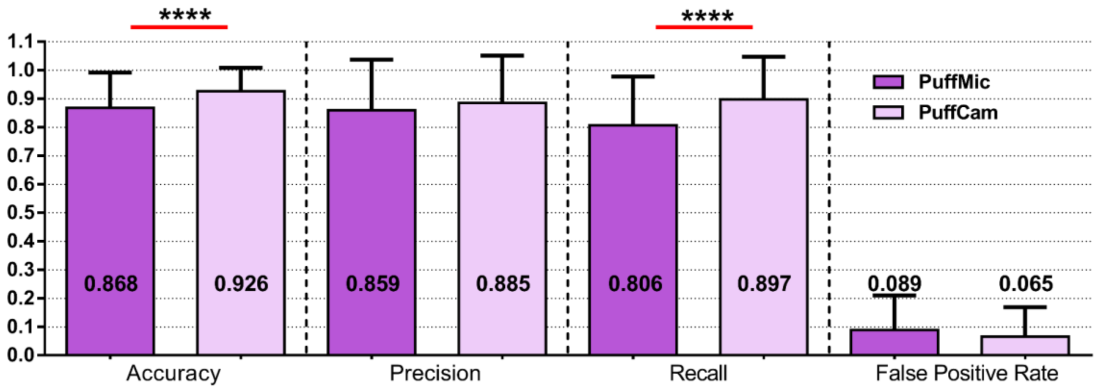


Figure 3.10: Mean values of the proposed software switches for all participants through evaluation metrics (accuracy, precision, recall, false positive rate) (****$p < 0.0001$).

3.3.5 Subjective Data based Results

The quantitative subjective data was collected from the filled-in questionnaire containing four statements ranked in a five-point Likert scale (Figure 3.9), while the qualitative subjective data was collected by the participants' responses to the open-ended questions, their feedback about interaction techniques and researchers' observations. The rating ranged from 1 *'strongly disagree'* to 5 *'strongly agree'* in the questionnaire for comfort assessment during experiments. With regard to the four statements, the mean value and standard deviation were calculated for all 46 participants. As can be observed in Figure 3.9, the statement *'I didn't have any respiration fatigue'* (m = 4.32) was rated lower than the statement *'Seating and positioning were comfortable'* (m = 4.76). Puffing on a modified camera (m = 4.45) was found easier than puffing on a microphone (m = 4.32). Overall, all statements were scored quite satisfactory by the participants. Prior to the experiments, all participants were quite confident that they can handle it, and they seemed excited to experience it.

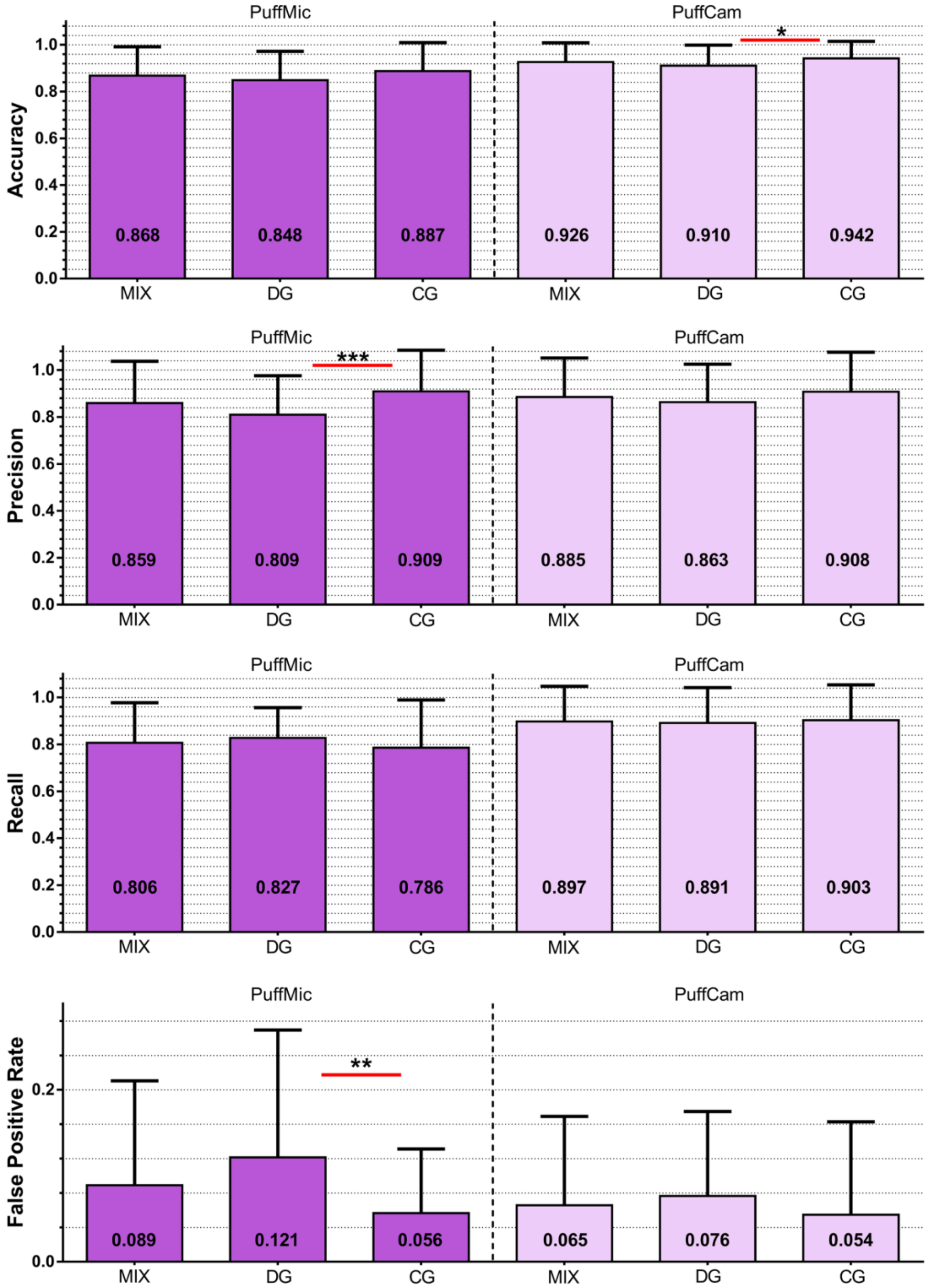


Figure 3.11: Mean values of interaction techniques through evaluation metrics (accuracy, precision, recall, false positive rate) according to the participant groups (Mix, DG, CG) (*$p < 0.05$; **$p < 0.01$; ***$p < 0.001$).

After the experiments were completed, all participants agreed that both interaction techniques were easy to use and stated that they would be looking forward to operating a computer access application based on these interaction techniques. They also all declared that they liked the techniques and that controlling a computer via breathing only without purchasing any dedicated device sounded very promising. Many participants stated that the respiration-based methods proposed would be an efficient alternative for the people with severe motor-impairments. On the other hand, some participants underlined that respiration-based systems might be a problem for elderly people who might have respiration difficulties. We even had to exclude two volunteers because of their upper respiratory infection. Most participants were pleased with the scanning time in the evaluation software SITbench 1.0 (1000 milliseconds), though two of them suggested to adjust it slower to track highlighted objects easier.

3.4 Conclusion and Discussion

Most of respiration-based solutions for people with motor disabilities depend on expensive dedicated hardware called sip-and-puff switches beyond standard computer peripherals. Moreover, each sip-and-puff switch emulates different keyboard characters or mouse clicks based on the decision of its manufacturer, which leads to a standardization problem. A commonly agreed standard is lacking on the switch-accessible software side as well; while some switch-accessible software expect to receive *enter* character, the others expect to receive different characters. On the other hand, non-invasive interaction techniques based on respiration activity are under-explored in comparison to invasive techniques like sip-and-puff devices. In other words, respiration operated interaction techniques are mostly limited to invasive sip-and-puff devices. Although they might meet some requirements of their users, sip-and-puff devices are expensive systems and have tubes inside the users' mouth that have to be changed regularly due to hygiene concerns. As a different approach, to the best of our knowledge, there is just one solution called the BlowClick [47] which doesn't require a dedicated device. But it is unable to serve as a switch for any switch-accessible interface.

In order to overcome the above-mentioned problems, we propose two novel non-invasive interaction techniques as software switches (the PuffCam and the PuffMic) based on the puff gesture alone where a strong puff detected by a microphone or a modified camera is considered as a pressed switch. Furthermore, both software switches provide the same functions of a traditional hardware switch and its software driver (i.e., an object on the screen can be selected, as it could be done by means of a hardware switch; and expected characters can be assigned, as they could be assigned by a software driver) without the need for a dedicated hardware and its software driver. Although the the PuffMic is similar to the BlowClick in terms of input method, the PuffMic differs from the BlowClick as being configurable to be compatible with switch-accessible interfaces.

The usability study conducted with 46 participants demonstrated that the accuracy, precision, recall and false positive rate of the proposed interaction techniques were quite impressive. For all evaluation metrics (accuracy, precision, recall and false positive rate), the PuffMic exhibited the worst performance. The reasons behind this based on our observations is considered as follows:

- Holding position of the headset microphone stable during the experiments is very important since the threshold value is assigned in the calibration step and highly depends on the initial position of the microphone. Any possible change in the position (angle or distance between the microphone and the user's mouth or nose) during the experiments might lead to a minor or a major change in the air pressure (originated from the exhalation pressure) applied on the microphone, which is considered as the main reason of high false positive rate. Although high resolution Uni-Directional-Cardioid Microphones can ease this problem, such microphones might be very expensive. Employing them is definitely out of the scope of this study because they are not standard computer peripherals;

In terms of mean values of accuracy, precision and recall, CG members performed better than DG members for both software switches except for the PuffMic in recall. DG members had a higher score of false positive rate than CG members for both interaction techniques, which means that DG members are more inclined to make false selection.

We have also collected the subjective data with a questionnaire containing four statements ranked in a five-point Likert scale (Figure 3.9), responses of open-ended questions, feedback from participants and observations of researchers. According to the questionnaire for comfort assessment, all statements were rated quite satisfactory by participants. All participants agreed that they enjoyed using the proposed software switches. Moreover, they stated that the idea of controlling a computer via breathing without purchasing any dedicated device sounded very promising. On the other hand, each software switch is affected by the external factors differently. While the PuffMic is not affected by the light level (i.e., it can even work in the darkness), the PuffCam is highly robust to the external voices. Additionally, the PuffCam also is not affected by the user's speech.

This study provides a preliminary evidence of our software switch approach. Because non-invasive respiration-based systems are underestimated considering the limited number of previous studies in comparison with invasive respiration-based sip-and-puff devices, this study can also encourage the researchers to place a greater emphasis on non-invasive respiration-based systems. Both software switches can be replaced with invasive puff devices in many cases. Because our system is cost-free (except for a laptop with a standard webcam or a microphone), the budget allocated for supplying expensive alternative assistive devices can be used for other requirements of people with disabilities, improving cost-efficiency for government social services. As a future study, we aim to compare the proposed software switches with traditional sip-and-puff switches. It is also worth noting that, the application area of interaction techniques we proposed should not be considered just for computer access systems. These interaction techniques can be employed and integrated with any other system where a puff-gesture of the users can be used. We compared the performance of CG with DG to see whether there is any significant difference between the groups, because we aim that our approach will be utilized by able-bodied people as well. For example, beyond assistive technology requirements, computer-game industry might employ our methods in a way that players use their respiration activity as a new input way while playing.

4

The HeadCam and the HeadGyro:

Novel Interaction Techniques Based on a Single Head-Gesture

Within this chapter, we focus on how to devise a better method that enables a person to interact with a computer by a single head-gesture. Head-operated CATs are useful solutions for the ones with complete head control; but when it comes to the people with only reduced head control, computer access becomes a very challenging task since the users depend on a single head-gesture like a head nod or a head tilt to interact with a computer. Therefore, two novel interaction techniques called the HeadCam and the HeadGyro are proposed within this chapter. In a nutshell, both interaction techniques are based on our software switch approach and can serve like traditional switches by recognizing head movements via a standard camera or a gyroscope sensor of a smartphone to translate them into emulated switch presses. A usability study with 36 participants (18 motor-impaired, 18 able-bodied) is also conducted to collect both objective and subjective evaluation data in this study. While the HeadGyro software switch exhibits slightly higher performance than the HeadCam for each objective evaluation metrics, the HeadCam is rated better in subjective evaluation. All participants agree that the proposed interaction techniques are promising solutions for computer access task.

4.1 Introduction

Head-operated CATs are considered as one of the most efficient examples of assistive solutions enabling hands-free computer access. They are generally based on human-computer interaction (HCI) techniques where a mouse cursor is operated by the user's complete head control ability. However, for the people who have only reduced head control ability (i.e., the ones who cannot operate the mouse cursor by moving head or any other activity), computer access is a very challenging task since the users have to interact with a computer by a single head-gesture. It is obvious that any new efficient interaction techniques based on a single head-gesture will play an important role to develop better CATs for the people with only reduced head control.

In accordance with our efforts to find a solution for people with only reduced head control to interact with a computer by a single head-gesture, we reviewed the current head-operated solutions in Section 4.2. We noticed that the majority of interaction techniques requires a complete head control ability. In other words, there are limited solutions which are capable of supporting single head-gesture access for people with reduced head movements. Considering these limited solutions which are single head-gesture based HCI techniques that provide alternative means for computer access task (given in Section 4.2.2), we identify two main problems. To overcome these problems, we propose two novel interaction techniques namely the HeadCam and the HeadGyro by following the principles of our software switch approach. The main problems of the existing systems and how we address them by applying our software switch approach are explained below:

- Requirement of Dedicated Devices: Traditional button switches are dedicated devices beyond standard computer peripherals, while software-based techniques [56, 57] do not require any dedicated devices. As low-cost solutions, the HeadCam and the HeadGyro software switches are based on a standard camera and a gyroscope sensor of a smartphone, respectively.

- Compatibility with Switch-accessible Interfaces: Although current software-based techniques [56, 57] support single head-gesture and do not require any dedicated device, they are unable to serve as a switch for any other switch-accessible interfaces. In other words, they are not compatible with any switch-accessible interface. They can only emulate mouse clicks within their interface. Both interaction techniques proposed can be configurable to generate any expected keyboard characters or mouse clicks, which makes them compatible with most switch-accessible interfaces.

In a nutshell, both interaction techniques can serve like traditional switches by recognizing the head movements via a standard camera or a gyroscope sensor of a smartphone to translate them into virtual switch presses. Furthermore, they don't require a dedicated device, and they are compatible with most of switch-accessible interfaces. As low-cost alternatives, they can be replaced with expensive traditional head switches for computer access. Currently, they are the only options as switches for the ones with a limited head control alone (i.e., the ones who have to use a switch-based system for computer access) who cannot afford any dedicated device. They are also capable of recognizing any motion of the other body parts such as the user's shoulder or leg, which makes them quite flexible switches. By this way, different physical gestures can be targeted easily, when the user becomes tired. Besides, neither of the proposed software switches require a physical strength to be activated unlike physical switches; especially the HeadGyro can even detect a minimal head movement to transform it into an emulated switch press. Since the HeadGyro software switch isn't affected by external factors like light or wind, it could be also employed for outdoor activities (e.g., operating a wheelchair).

A usability study with 36 participants (18 motor-impaired, 18 able-bodied) was conducted in order to evaluate the proposed software switches. The SITbench 1.0 benchmark [55] was employed for objective evaluation. Besides, we also applied a System Usability Scale (SUS) [58] questionnaire for subjective evaluation. While the HeadGyro showed slightly higher performance than the HeadCam for each objective evaluation metrics, the HeadCam was rated better than the HeadGyro in subjective evaluation. All participants agreed that the idea of controlling a computer via a single head-gesture without requiring any dedicated device sounded very promising.

This chapter proceeds with a literature review to summarize the current head-operated interaction techniques for computer access in Section 4.2. Subsequently, we introduce our software switches called the HeadGyro and the HeadCam proposed in Section 4.3. Then, we evaluate both interaction techniques by presenting objective and subjective evaluation results of our usability study in Section 4.4. Finally, we conclude and discuss our study in Section 4.5.

4.2 Related Works

In this section, to look from a broader perspective, we review the current head-operated HCI solutions that provide alternative means for computer access. We preferred to separate them into two main groups according to the condition whether they have a single head-gesture access support.

4.2.1 Head-operated Interaction Techniques without a Single Head-gesture Access Support

Interaction techniques in this group require a complete head control ability for hands-free computer access. In principle, they translate the users' head movements into mouse cursor movements in several ways.

One of the most popular techniques is wearing inertial sensors such as a gyroscope or an accelerometer on head (via a helmet or a cap) to control a mouse pointer [17, 18, 59-67]. These inertial sensor-based systems are mostly combined with a different sensor/switch to perform a mouse click task (e.g., in a way that head movements are detected by inertial sensors to control mouse pointer, and mouse clicks are performed by a puff switch). Another sensor-based solution called Headmaster Plus [68], which was evaluated in LoPresti et al.'s work [69], consists of ultrasonic sensors. Briefly, the user wears a headset including three ultrasonic sensors that wait an ultrasonic signal from a stationary transmitter on the user's computer. In this way, ultrasonic sensors determine the orientation of the user's head to convert them into mouse pointer coordinates.

Using a head pointer —a head-worn stick in principle— is another solution which permits the users to control, press or touch any target [70] by head, although this method is rarely preferred nowadays. Similarly, head-operated joysticks are alternative tools which enable the users to point a mouse cursor on the screen [30].

On the other hand, a specific part of the user's face (e.g., the tip of the nose) or the user's whole head can be tracked by a standard camera in order to transform head movements into mouse cursor movements on a computer screen[36-38, 71-88]. Mouse click tasks such as left or right clicks are generally performed with dwelling method (i.e., the user holds the mouse cursor steady for a given amount of time to perform click tasks) or with multi-modal approaches by means of other gestures like eye-blinks or tooth-clicks.

In addition to abovementioned approaches, head movements can also be followed by special camera-based systems to control a mouse cursor. In such systems, the user wears small reflective dots on his/her head/face or an infrared LED (light-emitting diode) which is placed on a helmet or a pair of glasses. These reflective dots are illuminated by an infrared or near infrared light source, and then a standard camera [89-91] or an infrared camera [19] tracks the position of target signals (coming from reflective dots or an infrared LED) for mouse cursor pointing. Likewise, RGB-D cameras as new vision sensor technologies are also able to do 3D mapping of head position to control mouse pointer [92].

4.2.2 Head-operated Interaction Techniques with a Single Head-gesture Access Support

For the ones with only reduced head control, there have been limited solutions which are able to support single head-gesture access. Using a traditional button switch via a scanning interface is a common technique where a head switch is mounted close to the user's head in a way that the user can hit it by tilting head (or by any activity moving head) [16, 93]. In addition to traditional hardware switches, there are just a few software-based techniques [56, 57] where a single head-gesture is employed to perform mouse clicks. In such techniques, the users are enabled to navigate the mouse cursor to the desired location by vision-based head tracking methods, and then mouse clicks are emulated according to the users' head-gestures as an alternative to dwelling method.

4.3 Software Switches

This section begins with the introduction of the common user interface of both software switches proposed. Afterward, The HeadCam and the HeadGyro software switches are explained respectively.

4.3.1 The User Interface

We designed a user interface, as shown in Figure 4.1 (a), which is employed for both software switches. Gamification techniques were applied to make software switches more engaging and fun. An initial state of the interface —where the user has a stable head position— can be seen in Figure 4.1 (a). The interface includes three dynamic game elements: (1) the earth, (2) the left and (3) the right red border lines. All three elements can be controlled by the user's head movements called pitch, yaw and roll as illustrated in Figure 4.1 (b).

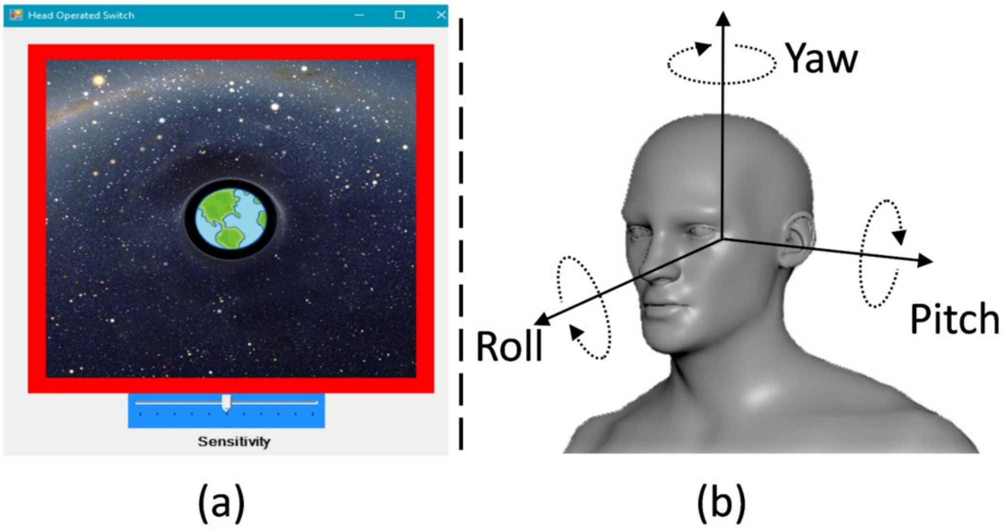


Figure 4.1: (a) The initial state of the interface of the HeadCam and the HeadGyro software switches. (b) Rotational movements of a head.

The sensitivity to control the game elements can be set according to the user's head control capability. As the sensitivity level gets higher, the user can move the game elements with a slower and minor head movement. The mission of the game is to save the earth from the gravity of a black hole by moving these three game elements until the earth intersects with the red border lines. Switch press and switch release are emulated according to this intersection situation. In other words, as soon as the earth intersects with the red border lines, a switch press is emulated until the end of intersection; while a switch release is emulated once the intersection between the earth and the red border lines is terminated. The intersection (i.e., switch press) is followed by a visual or an auditory sensory feedback provided to the user.

At the beginning, the interface can be configured to emulate any keyboard press or mouse action. Following an intersection detected, the interface sends the expected characters or mouse clicks to the target switch-accessible software like a CAT. This way, the proposed software switches let the users to control any compatible switch-accessible software. If the HeadCam is selected to be configured, the colour of the tracked object (i.e., the user’s head) is also assigned. Both software switches are compatible with Windows-based operating systems and were developed under .NET 4.5 framework.

In order to calibrate the earth's position, we simulated a gravity function that pulls the earth toward the black hole constantly. The gravity function becomes ineffective during the intersection (i.e., switch press). Once the intersection is over (i.e., switch release), the gravity function is reactivated. In this way, if the user keeps his/her head stable for a while when there is not any intersection, the earth will be pulled to its initial position eventually by gravity (i.e., to the centre).

As it is illustrated in Figure 4.2, each of six different head-gestures (i.e., rotational movements of the head) results in six different intersection states. While pitch (Figure 4.2 (a)) and yaw (Figure 4.2 (b)) movements control the earth's position, roll movements (Figure 4.2 (c)) operate the position of the right and the left red border lines.

4.3.2 The HeadCam

The HeadCam is based on a real-time video motion tracking algorithm which is similar with the PuffCam described in 3.2.3. In principle, the user's head is tracked by a built-in camera or a standard webcam to translate the roll movements of the user's head (as can be seen in Figure 4.2 (c)) into emulated switch presses. The algorithm of the HeadCam is listed step-by-step below:

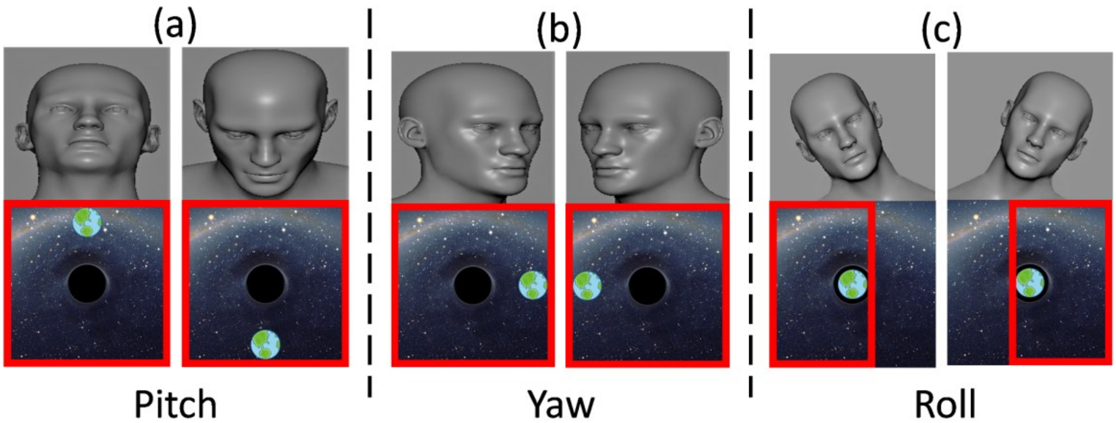


Figure 4.2: Six different intersection states of the interface according to rotational movements of a head.

- Video frames are taken by a camera with a frame rate of 15 frames per second and a frame size of 320x240 pixels (Figure 4.3 (a));
- Euclidean colour filtering is applied for each video frames according to assigned colour of the target object to be tracked during configuration (Figure 4.3 (b));
- Following Euclidean colour filtering, video frames are converted to grayscale (Figure 4.3 (c));
- All objects are detected in video frames (Figure 4.3 (d));
- The greatest object is chosen if there is more than one object detected (Figure 4.3 (e));
- The greatest object is tracked in real-time (Figure 4.3 (f));
- Every motion of the greatest object is transformed into the motion of the right or left red border lines as it is depicted in Figure 4.2 (c);
- Once the earth intersects with the red border lines, a switch press is emulated.

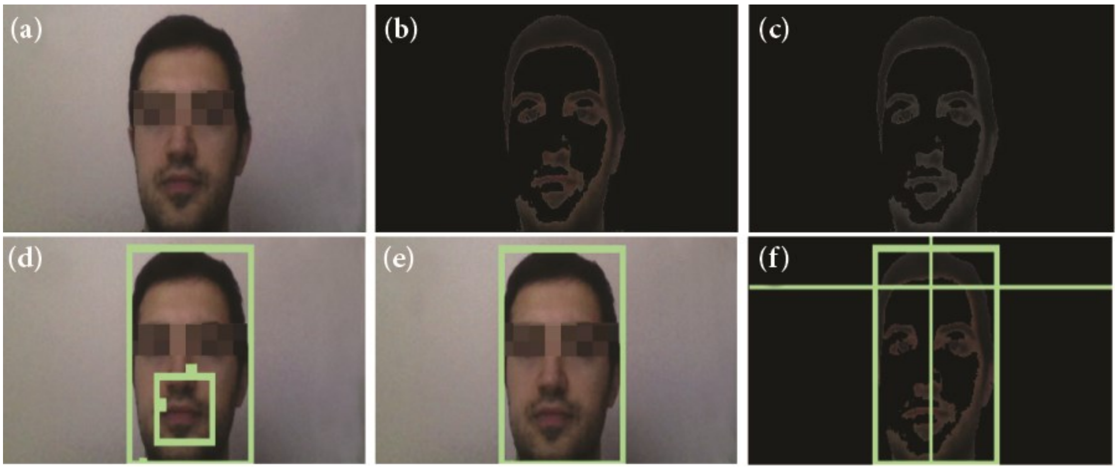


Figure 4.3: Steps of head tracking algorithm: (a) take video frames via camera; (b) apply Euclidean colour filter for each frame; (c) convert video frames to grayscale; (d) detect all objects in each frame; (e) choose the greatest object for each frame; (f) track the position of the greatest object.

Image processing library called AForge.NET was employed for filtering and object detection. Two roll movements of the user's head (right and left head tilts) can be easily recognized by the HeadCam, which makes our software switch capable of supporting double switch inputs for switch-accessible interfaces.

4.3.3 The HeadGyro

The HeadGyro interaction technique, basically, employs 3-axis gyroscope data of a smartphone —where the smartphone is placed on the user's head— to convert the rotational movements of the user's head into emulated switch presses. The smartphone can be placed on the user's head in several ways. For example, the user can wear a cap which is attached to the smartphone or a modified belt holding the smartphone as can be seen in Figure 4.4.

Figure 4.4: The placement of the smartphone on the user's head for the HeadGyro software switch.

The gyroscope is an important inertial sensor and mainly used to measure angular velocity of the sensor in inertial space. In other words, it measures the rate of change of the sensor's orientation. Today, inertial sensors like gyroscope are based on microelectromechanical system (MEMS) technology. They are employed in modern smartphones frequently since they are small, cheap, light, and offer low power consumption. In spite of all these advantages, because of the electromagnetic interference and the influence of semiconductor thermal noise, MEMS based sensors can exhibit high frequency noise like jitter, which affects the accuracy of the detected angular velocity. There have been several filtering methods used so far to reduce the noise of gyroscope data such as high/low-pass filter, forward linear filter, wavelet filter and Kalman filter. We preferred the Kalman filter to avoid jitter, considering the real-time requirements and its feasibility. We also developed a mobile application depending on Android operating system —which communicates with the computer in a wireless local area network (WLAN)— to convey the stream gyroscope data to the computer. The algorithm behind the HeadGyro is basically described step-by-step below:

- Real-time gyroscope stream data of the smartphone's 3-axis gyroscope sensor is drawn by our Android application;
- The Android application conveys this stream gyroscope data wirelessly to the computer;
- A simple Kalman filter is applied to this stream data as shown in Figure 4.5;
- Every motion of the user's head is translated into the motion of the game elements as illustrated in Figure 4.2;
- Once the earth intersects with the red border lines, a switch press is emulated.

4.4 Evaluation

A usability study was conducted to collect objective and subjective data. In this section, firstly we introduce the characteristics of participants. Then, we present the apparatus used within this study. Afterward, we briefly explain the tests and the procedure applied during the evaluation of the HeadCam and the HeadGyro. At last, we conclude the section with our experimental findings.

4.4.1 Participants

Following the approval of the Ethics Committee of the Izmir Katip Celebi University (Turkey) on 10.10.2018 (with a decision number: 332), the usability study was conducted at Medical Faculty of the University. All participants gave their informed consent before they participated in the study. A total of 36 participants, including 18 females and 18 males, took part in the evaluation of the proposed systems. While, disability group (DG) comprises 18 participants (6 females, 12 males) with motor-disabilities whose ages ranged between 18 and 68, control group (CG) without disabilities includes 18 people (12 females, 6 males) whose ages ranged between 18 and 59.

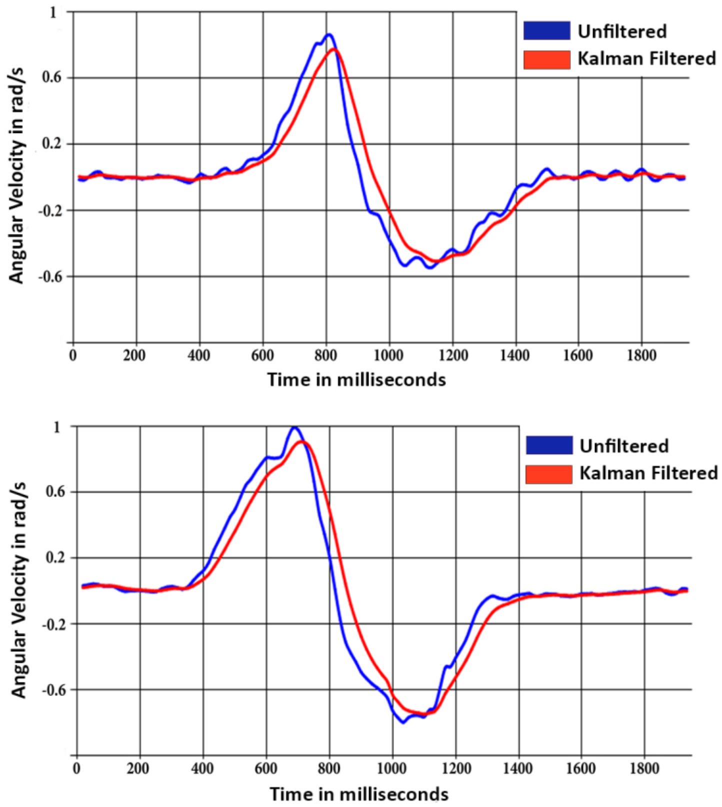


Figure 4.5: Two different stream data graphs based on the x-axis of the gyroscope sensor of two different participants when participants nod their head. Blue and red lines represent unfiltered and Kalman filtered gyroscope data, respectively.

In Table 4.1, age statistics of all participants are summarized according to groups. Main characteristics of the participants are listed in Table 4.2. All participants in DG were under medical treatment for several motor disabilities, while the experiments were conducted. On the other hand, participants of CG were generally accompanies of DG or staff working at the Physical Medicine and Rehabilitation Department. All participants met the following inclusion criteria: they are supposed to (1) find a target on the screen; (2) follow a moving target; (3) maintain gaze on a stable target; (4) stay focused on tests during experiments. As an inclusion criteria, all voluntary participants in DG had several difficulties in controlling their hands and thus couldn't operate a computer with conventional ways (i.e., with a mouse and a keyboard). Besides, there were five participants in DG who have reduced head control. Prior to experiments, we applied the MMSE —30-point questionnaire for cognitive assessment— to validate whether the participants can meet the cognitive ability to complete our tests.

Table 4.1: Age statistics of the participants according to the groups.

Groups	Gender	Mean Age	Number of Participants
Mix	Mix	43.2 (sd = 15.3)	36
Mix	Female	39.3 (sd = 14.2)	18
Mix	Male	47.1 (sd = 15.7)	18
DG	Mix	46.1 (sd = 17.3)	18
DG	Female	38.0 (sd = 19.5)	6
DG	Male	50.1 (sd = 15.4)	12
CG	Mix	40.3 (sd = 12.8)	18
CG	Female	39.9 (sd = 11.8)	12
CG	Male	41.3 (sd = 15.7)	6

Table 4.2: Main characteristics of the participants.

The User	Age	Gender	Disability
DG1	68	Male	Hemiplegia
DG2	21	Female	Hemiplegia
DG3	59	Male	Hemiplegia
DG4	27	Male	Tetraplegia
DG5	58	Male	Hemiplegia
DG6	57	Male	Hemiplegia

DG7	31	Female	Hemiplegia
DG8	18	Female	Hemiplegia
DG9	34	Female	Hemiplegia
DG10	63	Female	Hemiplegia
DG11	53	Male	Hemiplegia
DG12	65	Male	Hemiplegia
DG13	38	Male	Hemiplegia
DG14	62	Male	Hemiplegia
DG15	61	Female	Hemiplegia
DG16	34	Male	Hemiplegia
DG17	23	Male	Hemiplegia
DG18	58	Male	Hemiplegia
CG19	18	Female	None
CG20	51	Female	None
CG21	43	Female	None
CG22	55	Male	None
CG23	22	Male	None
CG24	27	Male	None
CG25	59	Female	None
CG26	25	Female	None
CG27	32	Male	None
CG28	41	Female	None
CG29	50	Female	None
CG30	28	Female	None
CG31	40	Female	None
CG32	34	Female	None
CG33	45	Female	None
CG34	45	Female	None
CG35	58	Male	None
CG36	52	Male	None

4.4.2 Apparatus

A laptop (Lenovo G505S; CPU: AMD A8-4500M 1.9 GHz; RAM: 6 GB DDR3; screen: LCD 15.6; OS: Windows 10 64 bits; resolution: 1600 x 900), an integrated camera of laptop (max digital video resolution: 1280 x 720; Image Sensor Type: 0.3 MP CMOS), and a smartphone with gyroscope sensor (Sony Xperia XZ1 Compact; CPU: Qualcomm Snapdragon 835; RAM: 4GB; OS: Android Oreo 8.0) were employed for experiments.

4.4.3 Tests

We used the SITbench 1.0 benchmark —presented in Chapter 6— which helps researchers to evaluate switch-based systems objectively. By means of this tool, objective evaluation data can be collected and saved automatically with standardized tests. To this end, we employed the Tie-Smiley Matching Game (TSMG) and Hungry Frog Game (HFG) tests of the SITbench 1.0.

For subjective evaluation, we applied the System Usability Scale (SUS) questionnaire [58] which consists of ten statements with a five-point Likert scale as can be seen in Table 4.3. Scale values range from 1 to 5 (1 = strongly disagree, 2 = disagree, 3 = neither agree nor disagree, 4 = agree, 5 = strongly agree). A SUS score (ranging from 0 to 100) is calculated based on scale value of the statements in a way that: (1) score contributions of each statement are summed where the score contribution is the scale value minus 1 for statements 1, 3, 5, 7, 9; the score contribution is 5 minus the scale value for statements 2, 4, 6, 8, 10; (2) the sum of the score contributions is multiplied by 2.5 to calculate the SUS score.

Table 4.3: Statements of the SUS questionnaire with average scale values of all participants. Δ symbol is replaced with the HeadGyro and the HeadCam, respectively, during assessments.

Statements	The HeadGyro Average Scale	The HeadCam Average Scale
1. I think that I would like to use Δ frequently	4.11	4.07
2. I found Δ unnecessarily complex	1.16	1.14
3. I thought Δ was easy to use	4.41	4.30
4. I think that I would need the support of a technical person to be able to use Δ	2.22	1.52
5. I found the various functions in Δ were well integrated	4.30	4.30
6. I thought there was too much inconsistency in Δ	1.19	1.22
7. I would imagine that most people would learn to use Δ very quickly	4.33	4.33
8. I found Δ very cumbersome to use	1.41	1.11
9. I felt very confident using Δ	3.97	4.27
10. I needed to learn a lot of things before I could get going with Δ	1.13	1.13

4.4.4 Procedure

At the beginning, the participants were informed about the test verbally. Then, we ensured that the participants and devices were positioned properly. Following a proper positioning, we let them to practice the tests (in a counterbalanced order) under our guidance, until they feel confident to start the tests. Afterwards, we applied two tests of the SITbench 1.0 to collect objective data: (1) TSMG: each software switch was tested by each participant (n = 36) with the first three templates of TSMG where scanning time was 1000 milliseconds. (2) HFG: each software switch was tested by each participant (n = 36) with the first three scenarios of HFG.

We applied the tests in the counterbalanced order to avoid learning and repetition effects. In order to prevent the mental or physical fatigue, we allowed the participants to get rest up to 5 minutes between the experiments. At the end of the SITbench 1.0 experiments, we also applied the SUS questionnaire to the participants for quantitative subjective evaluation. Besides, we collected the qualitative subjective data via our observations and participants' responses of open-ended questions about two software switches proposed within this study.

4.4.5 Objective Data based Results

As can be seen in Figure 4.6, according to the results of TSMG experiments, the HeadGyro demonstrated slightly better performance than the HeadCam in all performance evaluation metrics (accuracy, precision, recall, and false-positive rate). In terms of accuracy, mean value of the HeadGyro (m = 0.938) was greater than the HeadCam (m = 0.904), and the difference among mean values was found statistically significant ($p < 0.05$) according to Student's t-test for both software switches. For precision, the HeadGyro (m = 0.921) exhibited better performance than the HeadCam (m = 0.872), and there was a significant difference among means ($p < 0.05$). Regarding recall, the HeadGyro (m = 0.910) was followed by the HeadCam (m = 0.863) with a significant difference among means ($p < 0.05$) of both interaction techniques. For false-positive rate, the HeadCam (m = 0.077) was ahead of the HeadGyro (m = 0.048), and the difference among means was significant ($p < 0.05$).

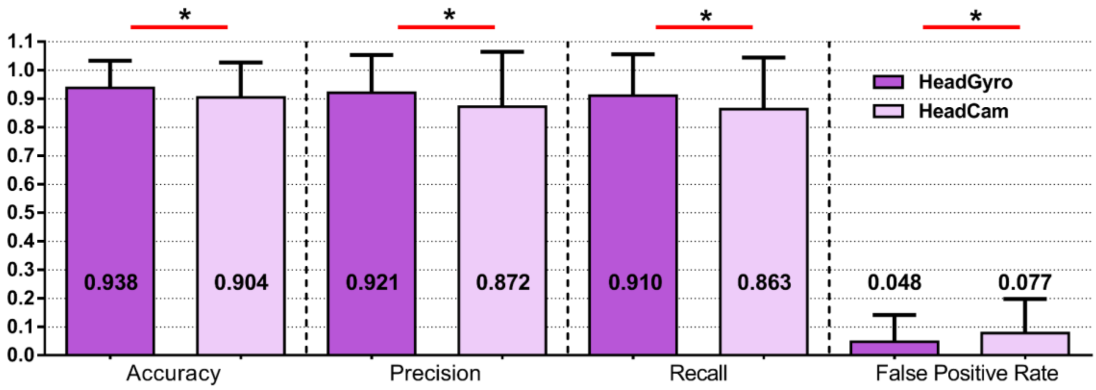


Figure 4.6: Mean values of interaction techniques acquired from all participants through evaluation metrics of TSMG including accuracy, precision, recall, and false positive rate (*$p < 0.05$).

Figure 4.7 presents the mean values of each software switches for TSMG depending on the participant groups (Mix, DG, CG). CG members performed better than DG members for both software switches according to the mean values through accuracy, precision and recall evaluation metrics. In false positive rate score, DG had higher scores than CG for software switches, which means that DG members made false selections more frequently when compared to CG members. The Student's t-tests for both interaction techniques through all evaluation metrics was applied to check whether there is a significant difference between the performance of DG members and CG members. The difference among means between DG and CG was not significant for all metrics.

Likewise, the HeadGyro proved a better performance in comparison to the HeadCam for all evaluation metrics of HFG (Figure 4.8) (average press time, average release time, the fastest press time, the slowest press time, the fastest release time, and the slowest release time). Mean values of both interaction techniques were presented in Table 4.4 depending on HFG experiments. According to p-values based on the Student's t-test results of all participants for both interaction techniques, it is demonstrated that there is a statistically significant difference among the means of the HeadGyro and the HeadCam through all evaluation metrics.

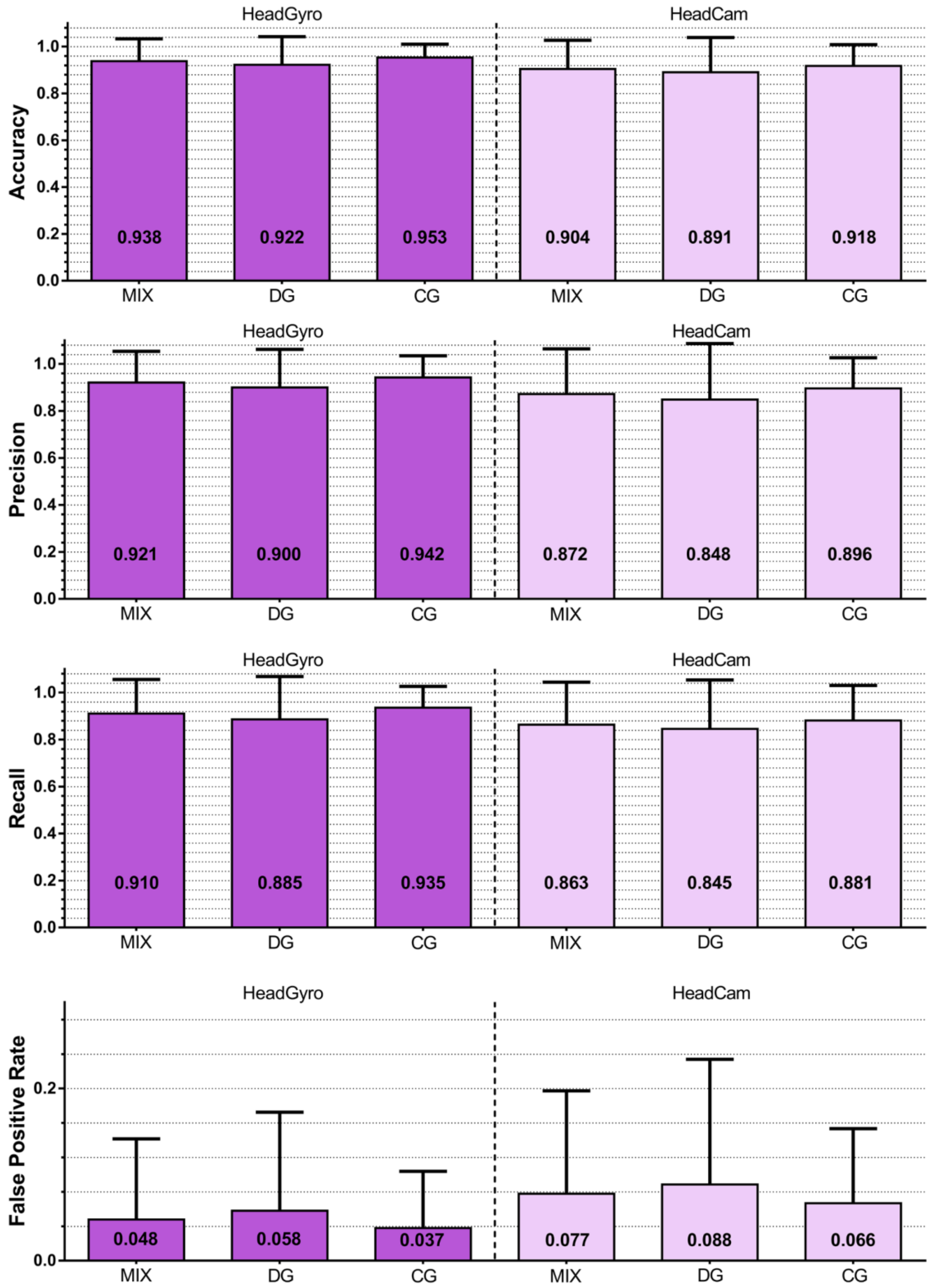


Figure 4.7: Mean values of the software switches through evaluation metrics (accuracy, precision, recall, false positive rate) according to the participant groups (Mix, DG, CG).

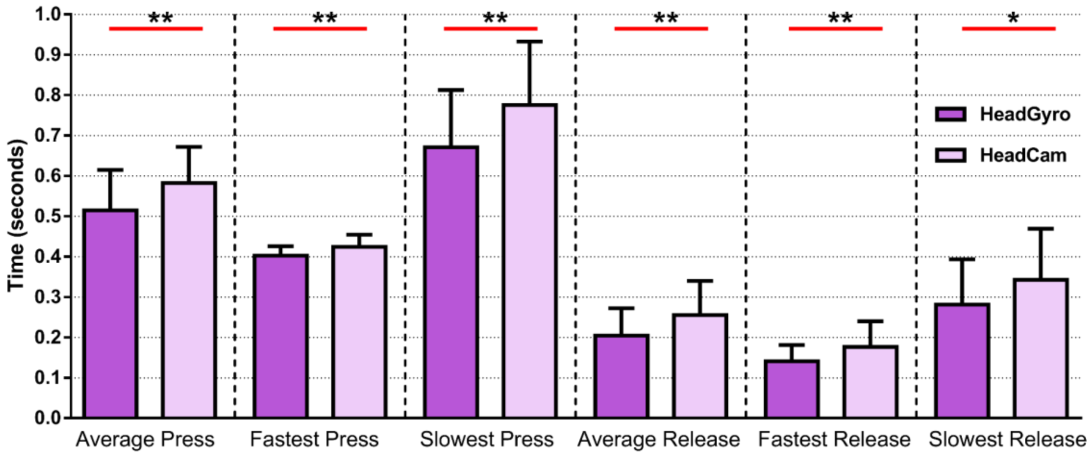


Figure 4.8: Mean values of two software switches for all participants through evaluation metrics of HFG (average press time, the fastest press time, the slowest press time, average release time, the fastest release time, and the slowest release time) (*$p < 0.05$; **$p < 0.01$).

Table 4.4: Mean values of the HeadGyro and the HeadCam through evaluation metrics of HFG (average press time, the fastest press time, the slowest press time, average release time, the fastest release time, and the slowest release time) for all participants.

Metric Type	The HeadGyro	The HeadCam
Average Press	0.514	0.582
The Fastest Press	0.402	0.424
The Slowest Press	0.670	0.775
Average Release	0.204	0.255
The Fastest Release	0.140	0.176
The Slowest Release	0.281	0.342

4.4.6 Subjective Data based Results

Results of the SUS questionnaire as quantitative subjective data are listed in Table 4.3. The average scale values acquired from all participants are represented according to the

HeadGyro and the HeadCam. The average SUS scores were calculated as 85.0 and 87,9 for the HeadGyro and the HeadCam, respectively.

According to the SUS adjective rating scale [94], both SUS scores can be considered as excellent. After the experiments, all participants agreed that both proposed interaction techniques are promising solutions for computer access tasks. They also declared that they were looking forward to experience both software switches to control a computer. Regarding to experiments with the SITbench 1.0, five participants stated that they would perform better if the scanning time/speed of TSMG test was set to a slower value, while four participants suggested to increase the size of smileys. All participants were pleased with the visual and auditory sensory feedback provided to the user during tests once the switch is activated or the target is appeared. While 31 of all participants declared that they would prefer to use the HeadCam for computer access, 5 of them chose the HeadGyro as their favourite software switch. They all agreed that gamification techniques made software switches more engaging. None of the participants experienced any fatigue during tests.

4.5 Conclusion and Discussion

Hands-free computer access via head movements is already a challenging task in comparison to conventional ways, but when it comes to the people who have limited head control, computer access becomes a more challenging task since the users are obligated to interact with a computer by a single head-gesture like a head nod or a head tilt. On the other hand, high-cost of dedicated devices —employed by the majority of current head-operated HCI solutions— creates a new barrier, although the aim of universal access is to break the barriers to enable equal opportunity and access for people with disabilities.

Alternative computer access methods can provide several useful services for the ones with motor disabilities in every part of life such as communication and education. Any new interaction techniques enabling computer access with minimal head movements will obviously help to enhance the quality of life and the self-sufficiency of people with reduced head control ability alone. Therefore, we proposed two novel interaction techniques called the HeadGyro and the HeadCam which depend on the gyroscope sensor of a smartphone and a standard camera, respectively. Both interaction techniques are based on our software switch approach that provides a comprehensive solution to the following problems of the current single head-gesture based interaction techniques: (1) requirement of dedicated devices, (2) compatibility with switch-accessible interfaces. In accordance with two principles of our software switch approach, the HeadGyro and the HeadCam software switches (1) don't require any dedicated devices; and (2) are configurable to be compatible with switch-accessible interfaces. In a nutshell, both software switches can serve like traditional switches by recognizing head movements via a standard camera or a gyroscope sensor of a smartphone to transform them into virtual switch presses.

According to the evaluation data of conducted usability study with 36 participants, HeadGyro showed slightly better performance than the HeadCam in objective evaluation, while the HeadCam was rated better than the HeadGyro in subjective evaluation. Furthermore, 31 of all participants declared that they would prefer to use the HeadCam for computer access, while 5 of them selected the HeadGyro. Based on our observations, the reasons behind this situation are considered as follows: (1) The head control ability is the key factor for this situation. The ones who have complete head control ability (31 participants) rated the HeadCam, while the ones with reduced head control (5 participants) preferred the HeadGyro since the HeadGyro is more sensitive and thus capable of recognizing tiny head movements. (2) The ones with complete head control can easily activate the software switch via a standard camera. As it was expected, wearing a smartphone on the head was found an unnecessary solution by the participants as long as their head control capability remains unimpaired or their head movements can be detected by the HeadCam. However, the HeadGyro can be advantageous if (1) the users cannot move their head enough to be recognized by a camera, or if (2) the external factors (e.g., low/high light or any moving object behind the user) cannot be tolerated by camera-based tracking. As can be concluded from the results of objective evaluation, the HeadGyro works in a more sensitive way in comparison to the HeadCam. On the other hand, CG members performed slightly better than DG members for both software switches through accuracy, precision and recall metrics. According to false positive rate score, DG members are more inclined to make false selections.

Both software switches can serve as the only low-cost options for the ones with limited head control who cannot afford the systems depending on high-cost dedicated devices. Beyond the head motions, the proposed software switches can be quite flexible by recognizing the other body motions to transform them into emulated switch presses. This flexibility also permits the user to change the targeted body motion once the user became tired. On the other hand, we didn’t find any significant difference between the performance of DG and CG members, which means that the proposed software switches could be employed for able-bodied people efficiently. They can be employed in multi-modal systems as new input techniques beyond the assistive technology area (e.g., as a new input for a computer video game). As another application domain, the HeadGyro software switch might be preferred during outdoor activities, since it is quite durable against the external factors like low light, high noise, and air conditions. As a future work, any other physical gesture —which is well-controlled by the user— can be targeted to evaluate the efficiency and usability of the proposed interaction techniques.

5

The GLOSTER 1.0:

A new Single-switch Accessible All-in-one CAT with a Novel Mouse Pointing Technique

In this chapter, we focus on how to devise a better CAT that enables a person to control a computer with a single-gesture. Therefore, we propose a new single-switch accessible CAT called the GLOSTER 1.0 with a novel mouse pointing technique called the CoorP. By means of the GLOSTER 1.0, trigger signals coming from the switches can be translated into meaningful commands to control a computer. As an all-in-one solution, it provides all three functions —pointing, clicking, and typing— conventionally performed by a mouse and a keyboard. A usability study conducted with 20 participants suggests that the CoorP provides a better solution than the CrossHair technique —which is the most popular mouse pointing technique in literature— to the single-switch based mouse pointing problem. Furthermore, the overall SUS score as 86,1 also demonstrates the high usability of the GLOSTER 1.0 in a real-life scenario.

5.1 Introduction

In conventional use, computer access is achieved by means of three basic functions as pointing, clicking, and typing; where pointing and clicking are performed by a mouse, and typing is performed by a keyboard. In case the users cannot access a computer through these conventional input methods because of motor-disabilities, the CATs are employed as useful solutions which allow the users to perform these three functions indirectly. Several types of CATs, which vary according to the CATs' abilities based on these three functions, have been developed so far. For example, while all-in-one CATs are capable of providing all three functions [3, 95-99], some CATs can only provide a specific function such as typing [100-107].

In CATs, typing and clicking are generally carried out using switches via a scanning method, while mouse pointing is mostly performed in a way that the user's head or eye movements are translated into the mouse cursor movements. But, if the user has an only single-gesture unimpaired, computer access becomes a harder task since all three functions have to be performed with a single-gesture/single-switch. Especially, single-switch based mouse pointing is a very challenging task in CATs.

We review the existing CATs which allow single-switch based mouse pointing in Section 5.2. We notice that they all depend on scanning techniques for mouse cursor pointing where the whole screen is scanned in several directions (i.e., vertical, horizontal or rotational) until the desired location is reached. Besides, there is just one non-commercial CAT allowing single-switch based mouse pointing, which means that the majority of existing solutions are not accessible for the ones who have financial difficulties to afford such CATs.

Within this chapter, we present a new cost-free single-switch accessible all-in-one CAT called the GLOSTER 1.0. As a different approach, we propose a novel single-switch based mouse pointing technique called the CoorP. It aims to improve the most popular mouse pointing technique called the CrossHair (described in Section 5.2) by navigating the mouse cursor to the target point more accurately at the same scan-line sensitivity. A usability study was conducted with 20 participants to evaluate the proposed CAT with the CoorP. We also develope a new test tool called the PointingChallenge for this usability study. The results of the usability study revealed that the CoorP performed more accurate mouse pointing than the CrossHair. On the other hand, for the ones who have difficulties to speak, the GLOSTER 1.0 also includes a speech-generator module as an AAC tool which converts the text typed by the user to speech.

This chapter proceeds with a section that reviews the existing solutions. Afterwards, we introduce the proposed CAT with the novel CoorP mouse pointing technique in Section 5.3. Then, we evaluate the GLOSTER 1.0 in Section 5.4 according to the conducted usability study. Lastly, we conclude and discuss our study in Section 5.5.

5.2 Related Works

In this section, we summarize the existing CATs that allow single-switch based mouse pointing in Table 5.1 according to the following three properties:

Is All-in-one: (Yes / No) The all-in-one CATs allow the users to perform all three functions —pointing, clicking, and typing— by using a single-switch, while the other CATs do not support single-switch based typing. In such cases, the user has to employ an on-screen keyboard for typing.

Is Commercial: (Yes / No) The ACAT [95] is the only cost-free CAT among the existing ones described in Table 5.1.

Scanning Technique: There have been six different scanning techniques which are employed by the current single-switch based mouse pointing techniques. In principle, they help to navigate the mouse cursor position to the desired point on the screen:

> The CrossHair: This is the most preferred mouse pointing technique by the existing CATs. It is generally performed in 4 steps:

(1) A horizontal scan-line scans up/down the screen continuously.
(2) The user presses the switch to stop the scanning when the horizontal scan-line intersects with the target point.
(3) Then, a vertical scan-line starts to scan the screen from the left/right side to the opposite side.
(4) The next switch press stops the vertical scan-line when it intersects with the target point.

The Inverse CrossHair: Although this scanning technique is similar with the CrossHair method described above, horizontal and vertical scanning depend on the press time of the switch.

(1) A horizontal scan-line scans up/down the screen as long as the user presses the switch.
(2) The user releases the switch to stop the scanning when the horizontal scan-line intersects with the target point.
(3) The user holds the switch pressed in order for a vertical scan-line scans the screen from the left/right side to the opposite side.
(4) The next switch release stops the vertical scan-line when it intersects with the target point.

The RadarMouse: This technique scans the screen like a radar as described in steps below:

(1) A line arrow rotates clockwise/counter-clockwise around the centre of the screen.
(2) The user presses the switch to stop the rotational scanning when the line arrow intersects with the target point.
(3) Then, the mouse cursor moves in the direction of the arrow.
(4) A final switch press stops the mouse cursor at the target point.

The Divide'nConquer: This technique aims to approach the target point by dividing the screen several times. For example, initially the entire screen is divided into two sub-parts. These two sub-parts are highlighted one-by-one until the user presses the switch to select the sub-part covering the target point. These divisions are repeated until the last divided screen is small enough.

The SelectDirectionFirst: The user presses the switch to select a direction offered in a scanning menu according to the target point. These directions are generally represented by arrows. Following the first switch press, the mouse cursor starts to move through the selected direction until the user presses the switch to stop it. If the mouse cursor is not at the target point, these steps are repeated by the user.

The Inverse SelectDirectionFirst: In a similar way with the SelectDirectionFirst technique, the user first presses the switch to select a direction offered in a scanning menu according to the target point. Then the user keeps the switch pressed to move the mouse cursor through the selected direction until the user releases the switch to stop the mouse cursor. These steps are repeated by the user, if the mouse cursor is not at the target point.

Table 5.1: Summary of the existing CATs allowing single-switch based mouse pointing.

Name	Is All-in-one	Is Commercial	Scanning Technique
ACAT [95]	yes	no	CrossHair
Grid 3 [96]	yes	yes	RadarMouse
SwitchXS [3]	yes	yes	RadarMouse, SelectDirectionFirst
EZ Keys [97]	yes	yes	CrossHair, RadarMouse
WINSCAN 3.0 CS [98]	yes	yes	Inverse SelectDirectionFirst
QUALIWORLD Basic [99]	yes	yes	CrossHair, SelectDirectionFirst
CrossScanner [108]	no	yes	CrossHair, Inverse CrossHair

ScanBuddy [2]	no	yes	Divide'nConquer
GUS [109]	no	yes	CrossHair, RadarMouse

5.3 Design of the GLOSTER 1.0

In this section, first, the user interface of the proposed CAT is introduced. Then, we explain how pointing, clicking and typing functions are achieved to control a computer by a single-switch.

5.3.1 The User Interface

As can be seen from the Figure 5.1, the GLOSTER 1.0 includes 4 modules to perform the pointing, clicking and typing functions. The interface locks itself to the right side of screen once the computer is started. At the beginning, in the configuration step, the expected key can be assigned according to the trigger signal which is sent by the preferred switch. For mouse cursor pointing function, in the configuration step, the user configures the scan-line sensitivity which is the time required in milliseconds to scan 1 pixel. For example, if the scan-line sensitivity is set to 10 milliseconds/pixel, then the vertical/horizontal scan-line moves one pixel for each 10 milliseconds. In other words, the scan-line scans 100 pixel for each second. The scanning time is also assigned in the configuration step which defines how fast the icons, the letters and the numbers are highlighted in the modules.

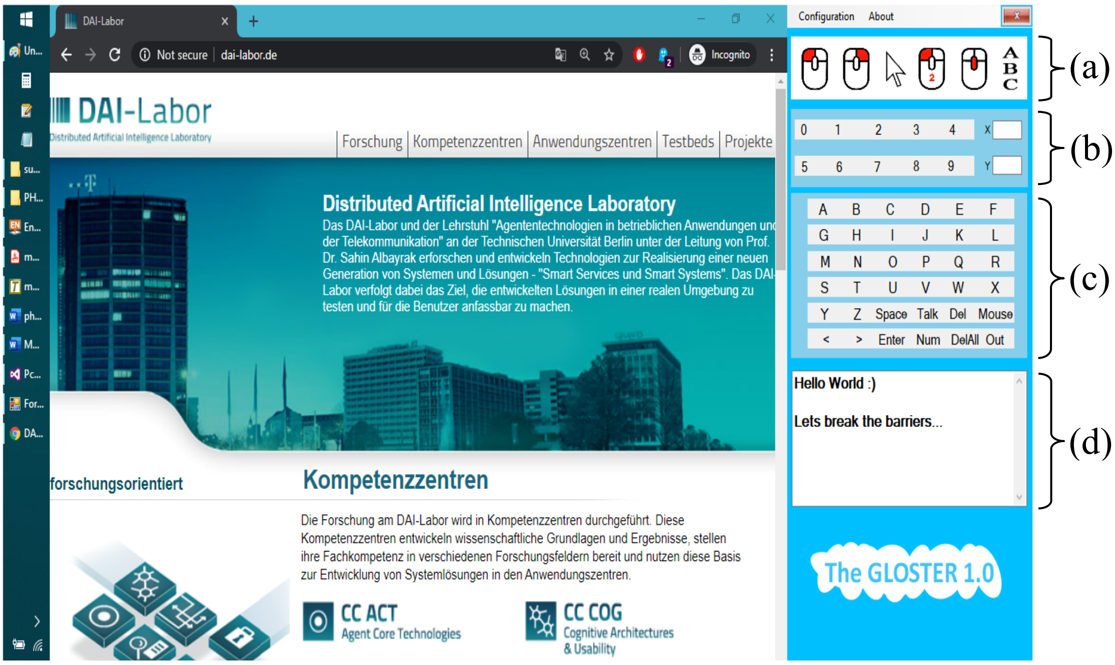
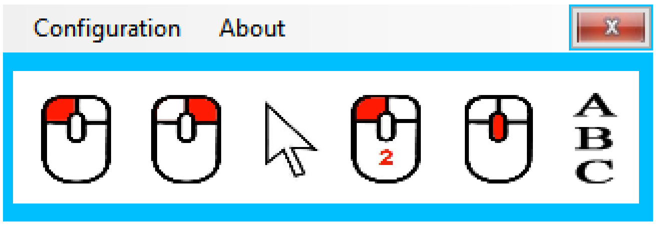


Figure 5.1: The user interface of the GLOSTER 1.0 including 4 modules. (a) The main menu module to select the pointing, clicking, and typing functions. (b) The numbers module to type the numbers and define the coordinates of the target point. (c) The speller module to type the letters. (d) The speech-generator module to convert the typed texts to speech.

In the main menu module (Figure 5.1 (a) and Figure 5.2), pointing, clicking, and typing functions are represented by icons. An automatic linear scanning method is employed to select these functions. In this method, first, the represented functions are highlighted one-by-one on screen for a scanning time; then, the user activates the switch if the highlighted icon is what s/he intends to select.



(a) (b) (c) (d) (e) (f)

Figure 5.2: The main menu module to select the pointing, clicking, and typing functions. (a) Mouse left-click. (b) Mouse right-click. (c) Mouse pointing function. (d) Mouse double-click. (e) Mouse middle-click. (f) Typing function.

By means of the numbers (Figure 5.1 (b)) and the speller (Figure 5.1 (c)) modules, the user can type the letters and the numbers; while the speech-generator module (Figure 5.1(d)) enables to convert the text typed by the user to speech.

5.3.2 Pointing: The CoorP Technique

The mouse cursor pointing function of the GLOSTER 1.0 is based on our novel CoorP technique. The user first selects the arrow icon (Figure 5.2 (c)) —which represents the mouse pointing function— in the main menu. Then, the CoorP technique is employed to navigate the mouse cursor. In principle, the CoorP improves the CrossHair technique in 7 steps. The first 3 steps reduce the whole screen to a more manageable size and help to approach to the target point more quickly than the CrossHair technique.

(1) At first, the whole screen (except the GLOSTER 1.0 interface) is divided into 100 sub-parts according to X-Y coordinate plane as can be seen in Figure 5.3. Thus, the user can recognize the target sub-part which includes the target point.
(2) Then, the user defines the X-coordinate of the target sub-part (between 0 and 9) in the numbers module (Figure 5.1 (b)) by pressing the switch if the highlighted number is the desired one. The numbers are highlighted according to the scanning time assigned at the configuration step.
(3) Afterwards, the second switch press defines the Y-coordinate of the target sub-part. Thus, the target sub-part can be focused in the following steps by employing the CrossHair technique.

(4) Following the identification of the target sub-part, a vertical scan-line starts to scan the identified sub-part from left to the right according to the scan-line sensitivity assigned in the configuration step.

(5) The third switch press stops the vertical scan-line when it intersects with the target point. Thus, the X-coordinate of the target point is identified.

(6) Then, a horizontal scan-line scans the identified sub-part from the bottom to the top.

(7) The user presses the switch when the horizontal scan-line intersects with the target point. This fourth switch press identifies the Y-coordinate of the target point. At the end of this step, the mouse cursor is navigated to the identified target point.

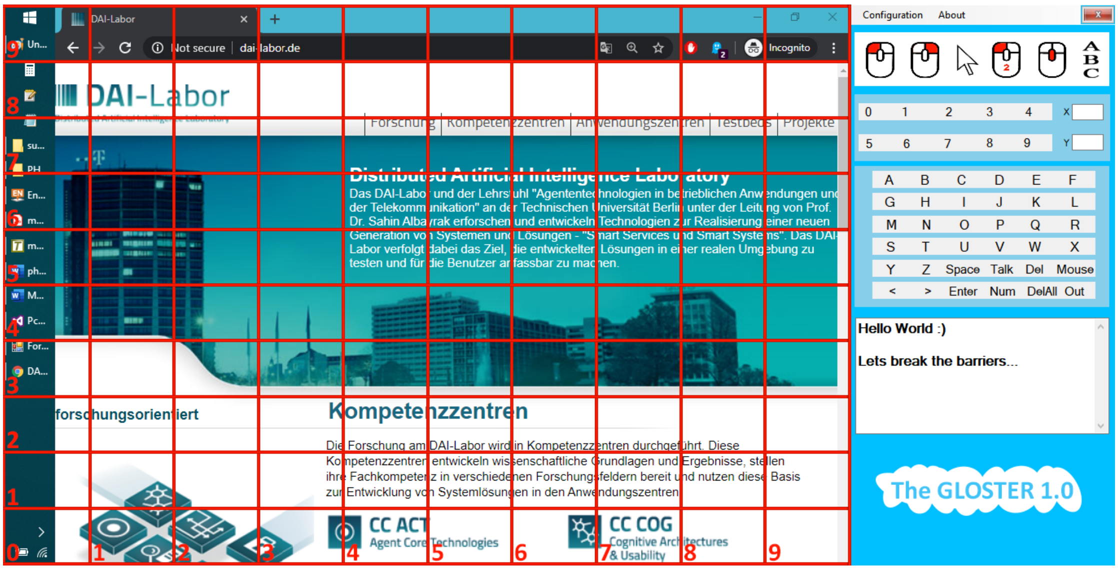


Figure 5.3: The user interface of the GLOSTER 1.0 which is divided into 100 sub-parts once the mouse cursor pointing function is selected.

5.3.3 Clicking

Clicking is performed by selecting the desired mouse click function within the main menu via the automatic linear scanning technique. For example, once the user selects the mouse left-click (Figure 5.2 (a)), the GLOSTER 1.0 emulates a left-click at the current position of the mouse cursor on the screen.

5.3.4 Typing

For typing function, the user selects the ABC icon (Figure 5.2 (f)) which represents the speller module from the main menu. Following this selection, row-column scanning method begins in the speller module. First, the rows are highlighted one by one from the top to the bottom until the user selects the desired row. Then, the letters are highlighted one-by-one. If the highlighted letter is the targeted one, the user hits the switch to type that letter into the speech-generator module or anywhere receiving text input on the screen such as the address bar of a web browser. The user can also pass to the numbers module from the speller menu to type the numbers.

5.4 Evaluation

In this section, we evaluate the GLOSTER 1.0 and the proposed mouse pointing technique CoorP in accordance with the results of a usability study conducted by us. Firstly, the participants are introduced. Then, we present the apparatus used in the usability study. Afterward, the tests and the procedure applied during the evaluation are explained. Lastly, we conclude the section by sharing the objective and the subjective evaluation data.

5.4.1 Participants

The usability study was conducted at Medical Faculty of the Izmir Katip Celebi University (Turkey) after the approval of the Ethics Committee of the University on 10.10.2018 (with a decision number: 334). Informed consents of all participants were received prior to the experiments. A total of 20 participants (11 males and 9 females) whose ages ranged between 21 and 65, took part in the evaluation of the proposed systems.

Table 5.2 summarizes the age statistics of all participants according to participants' gender. All voluntary participants have several difficulties in controlling their hands and thus couldn't operate a computer with conventional ways. Besides, they were all under medical treatment for several motor disabilities, while the experiments were conducted. All participants can (1) move their head; (2) find a target on the screen; (3) follow a moving target; (4) maintain gaze on a stable target; (5) stay focused on tests during experiments. We also applied the MMSE prior to experiments to validate that the participants have sufficient cognitive ability to complete our tests.

Table 5.2: Age statistics of the participants according to the genders.

Gender	Mean Age	Number of Participants
Mix	44.8 (sd = 13.2)	20
Female	42.7 (sd = 12.0)	9
Male	46.6 (sd = 14.4)	11

5.4.2 Apparatus

A laptop (Lenovo G505S; CPU: AMD A8-4500M 1.9 GHz; RAM: 6 GB DDR3; screen: LCD 15.6; OS: Windows 10 64 bits; resolution: 1600 x 900) and a smartphone with gyroscope sensor (Sony Xperia XZ1 Compact; CPU: Qualcomm Snapdragon 835; RAM: 4GB; OS: Android Oreo 8.0) were employed for the experiments.

5.4.3 Tests

We developed a new test tool called PointingChallenge for the experiments to collect the objective evaluation data. By means of this tool, we compare the CrossHair and the proposed CoorP techniques through a standardized test.

The user interface of the PointingChallenge can be seen in Figure 5.4. There are 3 different square boxes in a white panel whose size is 1000 pixels in width and 600 pixels in height. Sizes of the red, green and blue square boxes are 30x30, 25x25 and 20x20 pixels, respectively. The test tool has three input parameters to be configured previous to start the test. The first one is the mouse pointing technique to be tested (i.e., the Crosshair or the CoorP), while the second one is the scan-line sensitivity which is described in Section 5.3.1. The last input parameter is the scanning time which defines how fast the numbers are highlighted if the CoorP pointing technique is tested. After the test is completed, it provides a performance evaluation metric as an output called the mispoint count. In principle, the mispoint count is the sum of the wrong mouse cursor pointings during the test.

Figure 5.4: The user interface of the PointingChallenge with three boxes in different sizes and colors.

In the test, the user aims to navigate the mouse cursor onto the target square boxes in an order. The test is performed as follows:

(1) The test begins once the start button is clicked.

(2) Firstly, the red box is targeted. To do this, the test tool scans the white panel according to the employed mouse pointing technique. If the user succeeds to point the mouse cursor onto the red box, the step 3 begins. But, if the user makes a wrong mouse cursor pointing, the step 2 is repeated until the user succeeds.

During the test, the test tool counts all wrong mouse pointings as the mispoint count.

(3) Next, the green box is targeted. If the user succeeds, the step 4 begins. Otherwise, the step 3 is repeated.

(4) Lastly, the blue box is targeted. If the user succeeds, the test stops. Otherwise, this step is repeated.

At the end of the test, the mispoint count can be seen on the interface.

In the test, the CoorP technique is employed as it is described in Section 5.3.2, while the CrossHair technique is employed as below:

(1) A horizontal scan-line begins to scan the white panel from the bottom to the top according to the scan-line sensitivity.

(2) The user presses the switch if the horizontal scan-line intersects with the target box. The first switch press defines the Y-coordinate of the target box on the screen.

(3) A vertical scan-line scans the white panel from the left side to the right side.

(4) The user presses the switch to stop the scanning when the vertical scan-line intersects with the target box. This switch press defines the X- coordinate.

For subjective evaluation, we employ a SUS questionnaire which consists of ten statements with a five-point Likert scale as can be seen in Table 5.3.

Table 5.3: The statements of the SUS questionnaire according to the GLOSTER 1.0 with average scale values of all participants.

Statements	Scale
1. I would like to use the GLOSTER 1.0 frequently.	4.01
2. I found the GLOSTER 1.0 unnecessarily complex.	1.11
3. I found the GLOSTER 1.0 easy to use.	4.23
4. I would need the support of a technical person to be able to use the GLOSTER 1.0.	1.64

5. I found the various functions in the GLOSTER 1.0 were well integrated.	4.10
6. I thought there was too much inconsistency in the GLOSTER 1.0.	1.34
7. I would imagine that most people would learn to use the GLOSTER 1.0 very quickly.	4.29
8. I found the GLOSTER 1.0 very cumbersome/awkward to use.	1.12
9. I felt very confident using the GLOSTER 1.0.	4.22
10. I need to learn a lot of things before I could get going with the GLOSTER 1.0.	1.20

5.4.4 Procedure

First, the participants were informed about the test. Then, they practiced the tests in a counterbalanced order under our guidance. For all tests, the HeadGyro software switch which is described in Section 4.3.3 was employed. When the participants felt confident enough to start the tests, we applied the tests as follows:

(1) The CoorP and the CrossHair techniques were tested by each participant (n=20) twice for two different scan-line sensitivity values as 10 and 20, where the scanning time is 1000 milliseconds via the PointingChallenge tool. The tests were applied in the counterbalanced order to avoid learning and repetition effects. The participants were also allowed to get rest up to 5 minutes between the tests.

(2) Then, we asked the participants to complete a computer access task. In this task, the participants open the web-browser and search for the daily weather cast by using the GLOSTER 1.0 under our guidance. We didn’t collect any quantitative data such as mispoint count in this task. After the participants complete this task, we applied the SUS questionnaire to the participants for quantitative subjective evaluation about the GLOSTER 1.0. Besides, we also collected the qualitative

subjective data via our observations and participants' responses of open-ended questions.

5.4.5 Objective Data based Results

According to the results of the PointingChallenge test, the CoorP performed considerably better than the CrossHair for both scan-line sensitivity values in terms of the mispoint count. The difference among the means between two mouse pointing techniques was found statistically significant for both scan-line sensitivity values according to Student's t-test. As can be seen in Figure 5.5, once the scan-line sensitivity is doubled from 10 to 20, the mispoint count shows a noteworthy decrease both for the CoorP and the CrossHair.

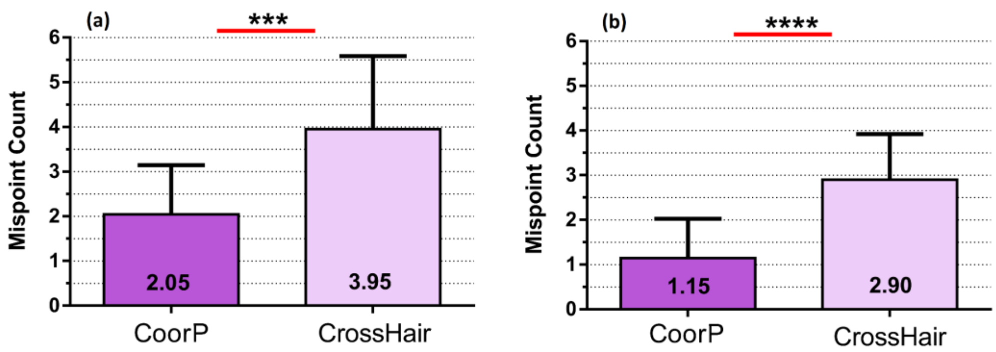


Figure 5.5: Mean values of the two mouse pointing techniques in terms of mispoint count according to the two different scan-line sensitivity level where in (a) the scan-line sensitivity is 10, (b) the scan-line sensitivity is 20. (***$p < 0.001$; ****$p < 0.0001$).

5.4.6 Subjective Data based Results

Table 5.3 shows the results of the SUS questionnaire as quantitative subjective data. It represents the average scale values acquired from all participants after they completed the computer access task by using the GLOSTER 1.0 and the HeadGyro. The average SUS score was calculated as explained in Section 4.3.3. The score was calculated as 86.1 which is considered as quite impressive.

All participants declared that they would prefer the CoorP technique for mouse cursor pointing instead of the CrossHair. 12 participants also stated that the CrossHair was overwhelming, since it scans the whole screen which leads to a long waiting time. They declared that the long waiting times causes them to lose their attention.

5.5 Conclusion and Discussion

In this chapter, we proposed a new single-switch accessible CAT called the GLOSTER 1.0 with a novel mouse pointing technique. The GLOSTER 1.0 is an all-in-one solution which allows the user to perform the three main functions for computer access as pointing, clicking, and typing. In other words, it provides all three functions which can be performed by a mouse and a keyboard in conventional way. By means of the GLOSTER 1.0, trigger signals coming from the switches can be translated into meaningful commands to control a computer.

We also proposed a novel single-switch based mouse pointing technique called the CoorP within this chapter. To evaluate the proposed technique and compare it with the CrossHair technique —which is the most preferred technique by the existing CATs— we developed a test tool called the PointingChallenge. A usability study with 20 participants was conducted by employing the PointingChallenge for the CoorP and a SUS questionnaire for the GLOSTER 1.0.

According to the usability study results, the proposed mouse pointing technique CoorP provided a better solution than the CrossHair technique to the single-switch based mouse pointing problem. Moreover, the overall SUS score as 86,1 also demonstrated the high usability of the GLOSTER 1.0 in a real-life scenario.

The mispoint count would be decreased further if the scan-line sensitivity was increased more in the experiments. But, in this case, the scan-line would scan the screen more slowly, which leads to a decrease in the attention level of the user's as it is already declared by the participants during the experiments. As a future work, more experiments should be performed to define the optimum scan-line sensitivity.

We need to note in here that the current typing method of the GLOSTER 1.0 was designed very simple. To improve the system, typing rate might be accelerated by designing a better alphabet in the speller menu according to the frequency of the letters. Furthermore, a word prediction algorithm might be implemented where a list of possible words is offered to the user as a new letter typed by the user.

A further future work would be devising a mark-up language which allows to perform pre-defined tasks in GLOSTER 1.0. For example, the user can check his/her e-mail once the user types the command <e-mail>.

We strongly believe in the GLOSTER 1.0 and the proposed software switches to have a marked impact on people's lives, since they are together capable of enabling the single-gesture based hands-free computer access without purchasing any software or dedicated devices.

6

The SITbench 1.0:

A Novel Switch-Based Interaction Technique Benchmark

Within this chapter, we focus on how to devise a better tool that enables objective evaluation of a SIT. Evaluation process of a SIT requires an interdisciplinary team effort and takes a considerable amount of time. Collecting subjective evaluation data from the users is a very common approach, but the subjective evaluation data alone might be manipulated and unreliable for comparing performances in many cases. Thus, therapists generally cannot succeed in determining the optimum SIT setup (i.e., determining the most appropriate combination of setup variables such as the switch type or switch site) at first attempts since it is hard to evaluate the measurable performance by collecting subjective data instead of objective data. On the other hand, the existing objective evaluation methods in literature are far from being a benchmark. To make performance evaluation of SITs by using a number of standard tests and empirical attributes, a benchmark application is also required. Therefore, we propose a novel benchmark for performance evaluation called the SITbench 1.0 that provides a quicker and more accurate switch evaluation process by collecting and saving the objective data automatically. We conduct a usability study with eight participants and demonstrate that the objective data collected via the SITbench 1.0 helps to determine the optimum SIT setup accurately.

6.1 Introduction

As we stated in Section 1.1, there have been many people with motor-disabilities worldwide who depend on SITs for computer access, communication or any other reasons. Therefore, plenty of different SITs have been developed varying from traditional hardware switches to software switches as proposed within this thesis. In principle, they assist the users to interact with their environment. For example, a user can select a target on computer screen by hitting a single switch [23], or an electric wheelchair can be operated via multiple switches [110]. It is obvious that an efficient evaluation process of a SIT plays a vital role to develop better SITs and determine the optimum SIT setup for motor-impaired people.

There are many variables in a SIT setup such as switch type, scanning time, switch site, the users' posture, activation method, etc. For example, even a simple button switch can be used in several ways: it can be activated by hand or any other body part. Likewise, the users can be positioned in different postures during switch usage, which might affect the performance dramatically. The main aim of a SIT evaluation is to determine the optimum SIT setup which is the most suitable combination of these variables for the users to interact with their environment. To this end, a considerable time and effort is needed by an interdisciplinary team that includes many trials with different variables of SIT setup. On the other hand, assistive technology professionals require a benchmark application [111], which is compatible with most SITs, to make a better comparison and evaluation automatically with standardized tests under the same conditions. Considering the increasing number of the SIT users, any tool that allows a more accurate and quicker SIT evaluation process becomes an important requirement day by day.

Currently, SIT evaluation is performed in three ways: (a) collecting subjective data [112-116] via questionnaires, observations and interviews by an interdisciplinary team; (b) collecting objective data [12, 25, 28, 40, 46, 117-120] via performance tests; (c) collecting both subjective and objective data [121, 122].

Because the subjective data alone might be unreliable and manipulated easily for performance evaluation, it might be hard to succeed in determining the optimum switch setup on the first attempts in many cases by therapists. They might need several attempts by re-applying questionnaires or making new observations. For each unsuccessful attempt, serious time and effort are required to collect a new subjective data. Thus, collecting subjective data is not a proper way to evaluate the measurable performance of a SIT. Without a performance evaluation, it might be very challenging to achieve the optimum SIT setup with subjective evaluation alone. On the other hand, although collecting objective data is the most appropriate method for performance evaluation, current objective evaluation methods in literature are far from being a benchmark. These methods are mostly designed to evaluate just a specific SIT, which makes them ineligible to be a benchmark where the other SITs could be evaluated via standardized test. To the best of our knowledge, there are only two evaluation applications in literature [118, 119] which are close to be a benchmark for SIT evaluation. They can provide quantitative data to evaluate computer access skills and help therapists to choose the switch type and position. But both applications have some common limitations that we aim to overcome with our novel tool the SITbench 1.0:

- **Incompatibility:** Switch-accessible applications might require different keyboard characters or mouse clicks from switches to work. Furthermore, each switch might emulate and send different keyboard characters or mouse clicks depending on its manufacturer. Unfortunately, commonly agreed standard is not available. For example, while some switch-accessible applications might expect to receive a keyboard space character, other applications might expect to receive a mouse right-click. Both applications expect to receive a mouse left-click to work. In other words, they are only compatible with switches which are able to emulate mouse left-click. The remaining switches are excluded, which means that just a minority of SITs are compatible and could be evaluated with these applications. Therefore, we consider that they are far from being a proper benchmark for SIT evaluation. Our novel tool the SITbench 1.0 is compatible with all switches, which can emulate any mouse-clicks or keyboard characters, since it allows therapists to assign the expected characters from any switch.

- **Limited number of switches:** They are only capable of evaluating single switch systems. Double switch support is also required, since double switch usage is widely used as an alternative interaction method. The SITbench 1.0 is capable of allowing both single and double switch evaluation.
- **Limited number of tests:** Both applications employ only one test that measures press time (i.e., the time from the prompt to the when switch is pressed) and release time (i.e., the time from when the switch is pressed until it is released) of a switch. The SITbench 1.0 includes two more additional tests to evaluate SITs with single and double switch.
- **Database requirement:** They have some reporting functions for the test results. However, we considered that a well-structured database would be useful to share the results and apply some queries or statistical tests. In addition to reporting function, the SITbench 1.0 also allows to save the test results automatically into a Microsoft Access database.
- **Sufficient attention span requirement:** Sufficient attention span via both applications might not be achieved especially by infants, since they can become distracted and lose their attention easily during long and boring sessions. We applied gamification techniques while designing the SITbench 1.0 tests with the intent to make evaluations more engaging and fun.

Therefore, we propose a novel SIT evaluation tool namely the SITbench 1.0 as a benchmark application which helps to determine the optimum SIT setup with the aim of providing a quicker and more accurate SIT evaluation process. To collect the objective data, the SITbench 1.0 includes three different games which can be played via single or double switch. It measures and saves the performance metrics (accuracy, precision, recall, false positive rate) automatically at the end of each trial.

A usability study with eight participants was conducted as a part of this work in order to test and demonstrate the proposed benchmark application. We identified two different switch sites to be tested by the users under the same conditions in order to determine the most suitable switch site. To this end, we collected the objective data via the SITbench 1.0. Results revealed that the SITbench 1.0 could help to determine the optimum switch setup accurately. We also applied a SUS [58] questionnaire to evaluate the SITbench 1.0 itself, and the results were quite satisfactory.

More potential SIT users can be served at the same time period with the same workforce since a quicker and more accurate SIT evaluation process is provided by the SITbench 1.0, which might prevent governments to spend high amounts of money as a result of better cost and schedule management. As a benchmark application, it allows to make objective comparisons with standardized tests under the same conditions by collecting the performance data of SITs for assistive technology community automatically. Thus, it provides extra time for therapists to observe more subjective aspects of client needs. On the other hand, it might be used to evaluate fine-motor skills of clients as a clinical tool. Occupational therapists can track the patients' progress by the SITbench 1.0 that allows to measure and record clients' fine motor performance and reflexes automatically in the form of quantitative objective data. The SITbench 1.0 might also help to improve the contingency awareness of the ones with profound and multiple learning disabilities, or it might be useful for pupils with severe learning difficulties to assess their auditory and visual attention.

This chapter proceeds with Section 6.2 that presents the design and implementation of our novel switch evaluation tool. Then, in the Section 6.3, we share the objective results of our usability study and the questionnaire results of the SITbench 1.0. Finally, we conclude our study and discuss our future work in Section 6.4.

6.2 Design of the SITbench 1.0

The SITbench 1.0 is designed as a novel benchmark application for assistive technology and healthcare professionals to determine the most appropriate SIT setup. It helps to collect and save the objective data automatically with the aim of optimum SIT setup. To this end, three different switch-accessible games, depending on single or double switch, were designed within the SITbench 1.0 namely *Tie-Smiley Matching Game, Non-stop Driver Game* and *Hungry Frog Game* respectively.

The SITbench 1.0 welcomes the users with a very simple interface (Figure 6.1) when it is initialized. In welcoming screen, the users can select the games (i.e., tests) and open the key assignment module to assign the expected keys from switches.

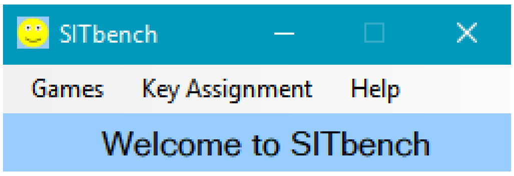


Figure 6.1: Welcoming screen of the SITbench 1.0.

As can be seen in Figure 6.2, single and double switch settings can be configured according to expected key (i.e., a keyboard character or a mouse-click) from any SITs to be tested via the SITbench 1.0. In this way, the SITbench 1.0 becomes compatible with the majority of assistive switches, since almost all switches on the market can emulate a keyboard character or a mouse-click.

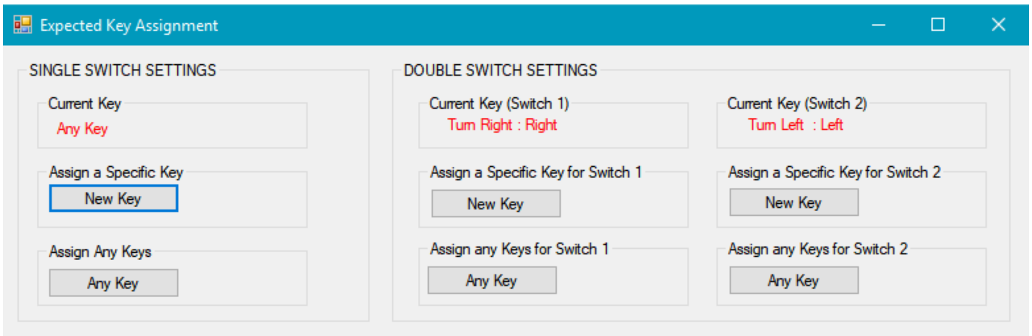


Figure 6.2: Expected key assignment module.

6.2.1 Tie-Smiley Matching Game (TSMG)

TSMG is a single-switch accessible game based on indirect selection with automatic linear scanning method. As it is exemplified in Figure 6.3, an indirect selection with automatic linear scanning method can be summarized in three steps: (1) letters in a scanning array (English alphabet as a selection set) are highlighted one-by-one on the screen for an equal duration t units of time where t represents scanning time, i.e., time interval between two successive states; (2) until the end of each state, the user is allowed to make a selection by hitting a switch or sending any kind of signal detected by a sensor (e.g., a blink); (3) if the highlighted letter is the target (i.e., what the user intends to select), the user sends a selection signal such as blinking.

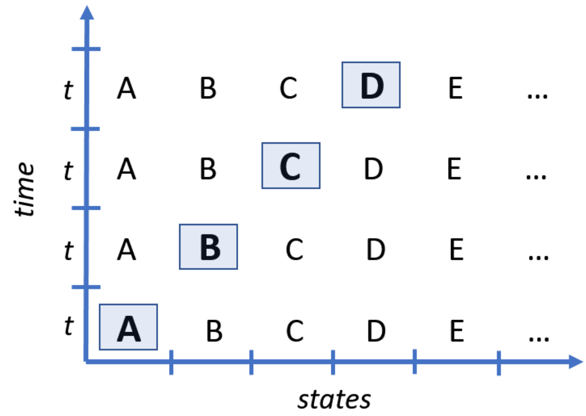


Figure 6.3: Time-state model of an automatic linear scanning sample.

There are five different templates which could be tested via TSMG. Figure 6.4 shows an initial form of template 1. The scanning array of each template consists of yellow and red smileys (26 smileys in total). Red smileys are targets to be selected, and they are set in a different order for each template in order to avoid repetition. Targets are seen by the user before starting and during the test.

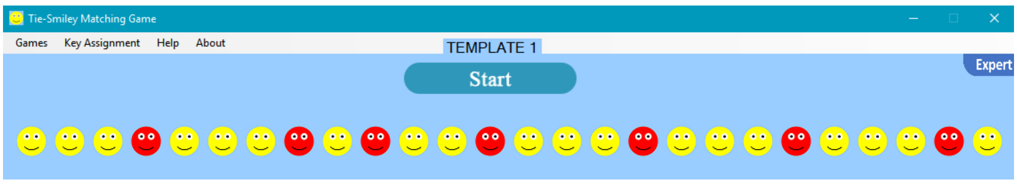


Figure 6.4: Initial form of TSMG in template 1.

TSMG is based on automatic linear scanning where each smiley is highlighted for a given time period (i.e., scanning time) one-by-one. The user should activate the switch once the highlighted smiley is red one. The user also hears a click sound as an auditory prompt, as soon as the target is highlighted. When the switch is activated, it sends the expected key to the SITbench 1.0 as a selection signal. Once the expected key is received, the SITbench 1.0 gives a sensory feedback by swapping the background colour of the interface like a blink. In expert mode (Figure 6.5), therapists can enter some details about the user. They can also select the template and set the scanning time (in milliseconds).

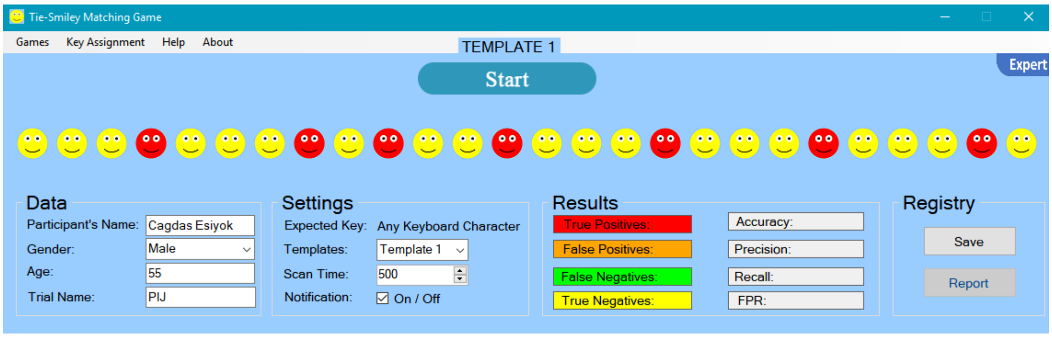


Figure 6.5: Initial form of TSMG in expert mode.

The user aims to match each smiley with a tie in a way that smiley and its tie are in the same colour (e.g., red smileys with red ties). To this end, the user should select all red smileys but yellow ones via a switch. A sample view of results after the user completed a trial without any mistake can be seen in Figure 6.6.

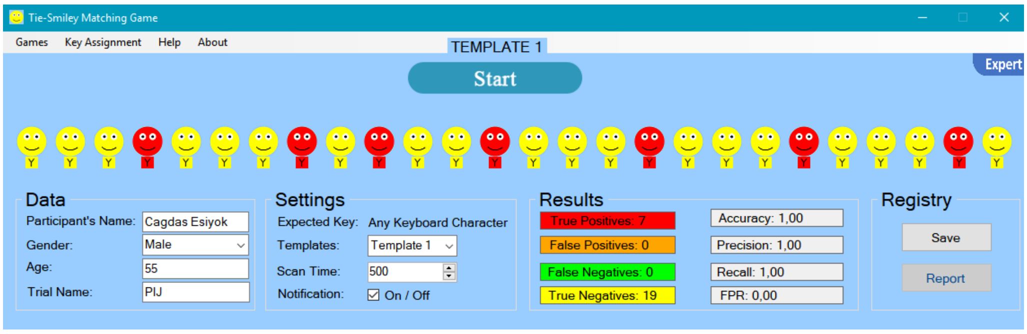


Figure 6.6: A view of TSMG in the end of a trial following a user performance without any mistake.

At the end of each trial, confusion matrix variables (true positives (TP), false positives (FP), false negatives (FN), true negatives (TN)) are calculated and assigned automatically as can be seen in Figure 6.7 according to count and colour of ties in a way that: TP represents count of red ties; FP represents count of orange ties; FN represents count of green ties; TN represents count of yellow ties. All performance evaluation metrics (accuracy, precision, recall and false positive rate) are measured by SITbench automatically at the end of trial by using the following formulas:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$false\ positive\ rate = \frac{FP}{TN + FP}$$

The SITbench 1.0 allows therapists to save the results and all data into a structured database, and it has also a reporting function to print or save the results as a document.

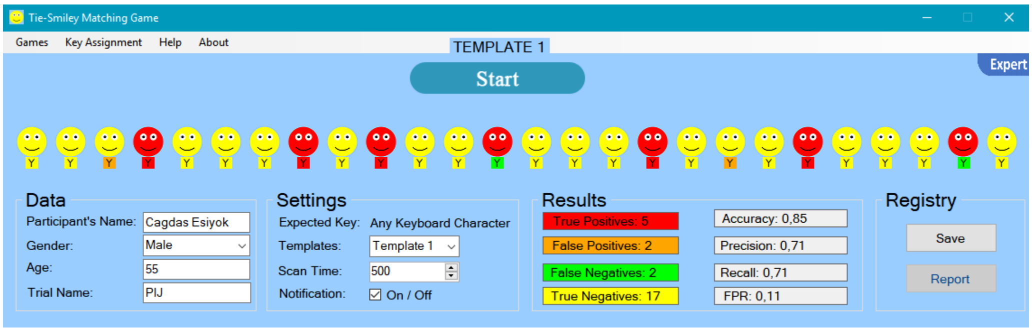


Figure 6.7: A general view from TSMG in the end of a trial following a user performance with several mistakes (i.e., with false negatives and false positives).

6.2.2 Non-stop Driver Game (NDG)

NDG can be played with single or double switch. While the single-switch accessible version is called SS-NDG, double-switch accessible version is called DS-NDG. Figure 6.8 shows the initial form of SS-NDG. Game objects are labelled with blue numbers in

Figure 6.8 to introduce them: label *1* shows left signal (i.e., orange square box); *2* is right signal; *3* represents green car; *4* is finish line.

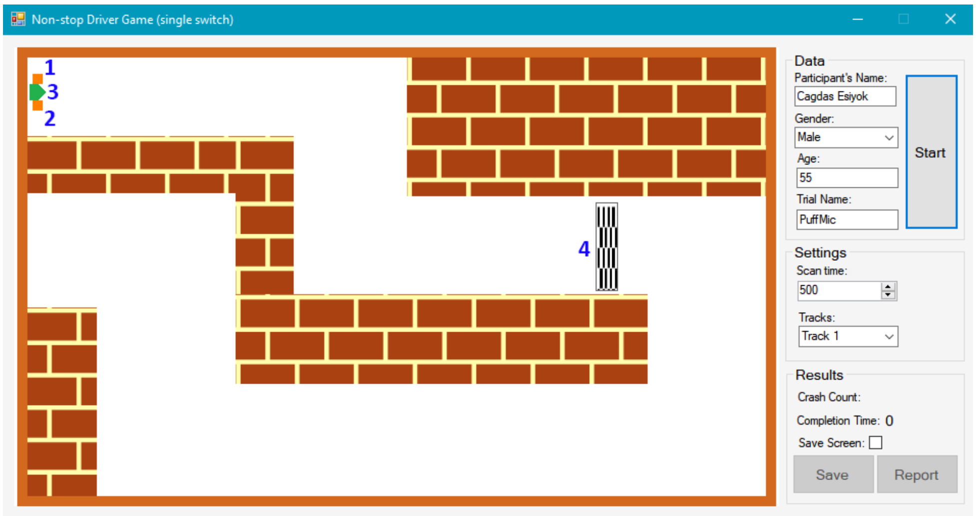


Figure 6.8: The initial form of SS-NDG where *1* shows left signal (i.e., orange square box); *2* is right signal; *3* represents green car, *4* is finish line.

Before starting the game, therapists can select the track and adjust the scanning time. The aim of the user is to reach the finish line as soon as possible with minimum crash into the walls. SS-NDG depends on automatic scanning method with a single switch where signals (i.e., car's left and right signals which are illustrated as orange boxes) are flashed one by one for a given time period (i.e., scanning time in milliseconds). After the game starts, the car begins to move and never stops until reaching the finish line. To turn the car left, the user hits the switch once car's left signal is flashed; and hits the switch once right signal is flashed to turn the car right. The only difference between SS-NDG and DS-NDG is that the car in DS-NDG doesn't have left and right signals (Figure 6.9) since it is controlled with double switch. The user activates the first switch to turn it left and the second switch to turn it right.

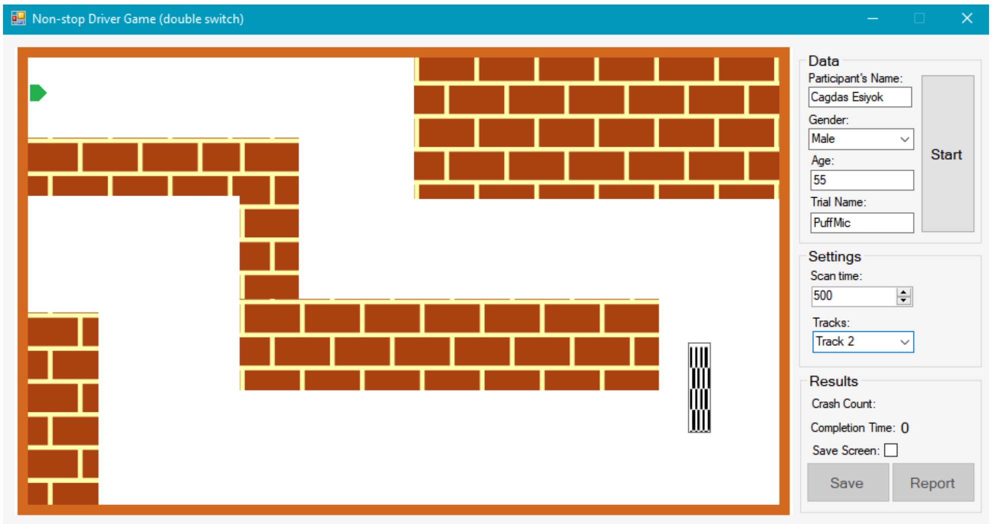


Figure 6.9: The initial form of DS-NDG in track 2.

We also assigned two different sounds to the SITbench 1.0 as auditory prompts according to left and right signals. In other words, the user hears two different sounds when signals are flashed during game. Once the user hits the switch, expected key is received and the SITbench 1.0 provides a sensory feedback visually by swapping the background colour of the interface. There have been five different tracks where each track has a different finish line location from each other to avoid learning effect. A sample view in the end of a trial is shown in Figure 6.10 where the user completed game via double switch in track 3 without any crash. Completion time (in seconds) and crash count are measured automatically as performance metrics at the end of each trial. The SITbench 1.0 enables therapists to save all results and data into a database and to report them as a document. It also depicts a black tracking line of the car (Figure 6.10) and allows therapists to save the screenshot of the interface as a separate image file.

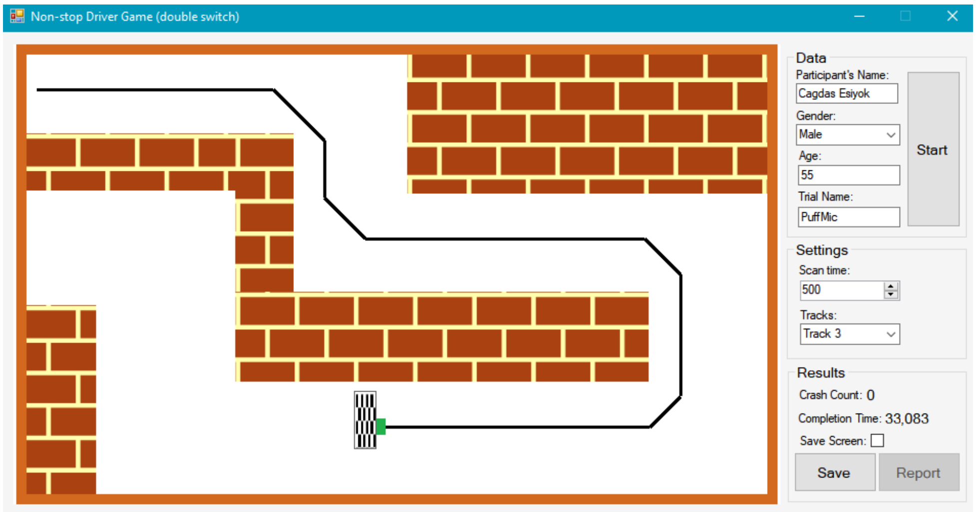


Figure 6.10: A view of DS-NDG in the end of a trial where the user reached to the finish line in track 3 without any crash.

6.2.3 Hungry Frog Game (HFG)

HFG is a single-switch accessible application to measure the user's switch performance. At the beginning of each trial, therapists can select the scenario and enter the user's details. Each trial of the game consists of ten tasks. As it is illustrated in Figure 6.11, each task in a trial is achieved in a way that: (a) the user waits until a fly is appeared; (b) the user activates the switch as soon as a fly is seen; (c) the frog eats the fly once the user activates the switch.

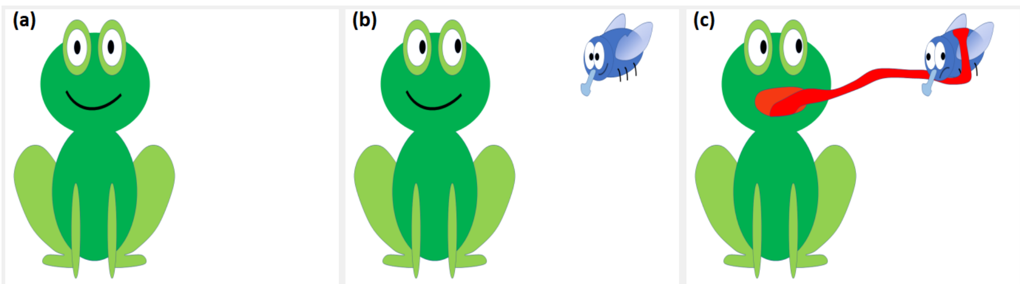


Figure 6.11: All three frames shown to the user during a task: (a) the frame shown until a fly is appeared; (b) the frame shown until the user activates the switch; (c) the frame shown once the user activates the switch.

As soon as the fly appears, the user hears a click sound as an auditory prompt. When the expected key is received from the switch, background colour of the SITbench 1.0 is swapped like a blink to give a sensory feedback. After the user completes ten tasks, the SITbench 1.0 measures average press time (i.e., the average time from when the fly appears to the when switch is pressed) and average release time (i.e., the average time from when the switch is pressed until it is released) automatically. The fastest and the slowest press time and release time among ten tasks are also detected. HFG has five different scenarios to avoid repetition and learning affect. For each scenario, waiting times (i.e., the time from when the user starts to wait to the when the fly appears, and not more than 6 seconds) of each task in a trial are set different from each other. Figure 6.12 shows the view of interface in the end of each trial. Six performance metrics (measured in seconds) can be saved into a database and reported via the SITbench 1.0: (1) average press time; (2) average release time; (3) the fastest press time; (4) the slowest press time; (5) the fastest release time; (6) the slowest release time.

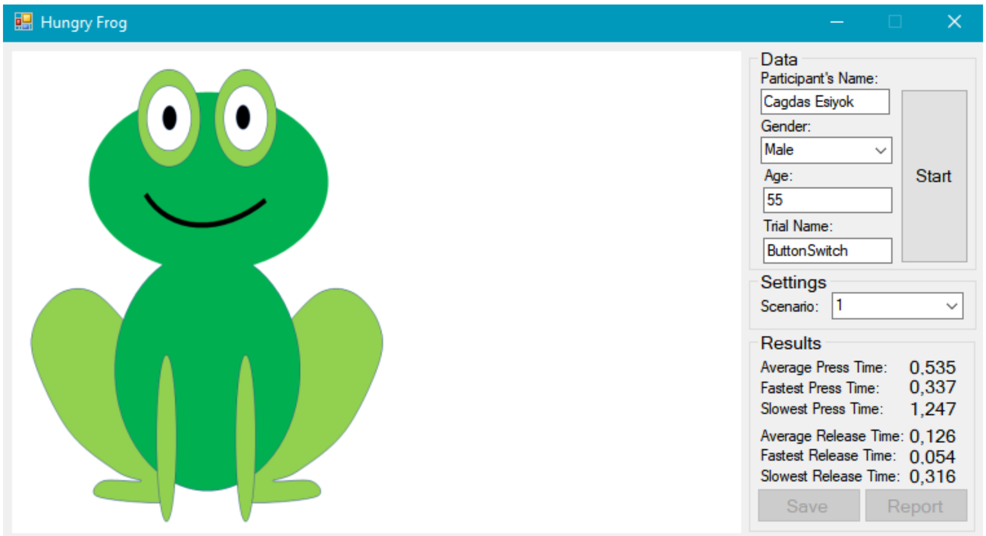


Figure 6.12: A view of HFG in the end of a trial.

6.3 Evaluation

We conducted a usability study as a demonstration of the SITbench 1.0. We identified two different switch sites (Figure 6.13 (c)) to be tested: forefinger distal pulp (hereafter:

FDP) and forefinger proximal interphalangeal joint (hereafter: FPIJ). FPD was considered as a proper switch site to activate a switch easily in contrast to FPIJ. We aimed to demonstrate that the SITbench 1.0 can determine the most proper switch site. To this end, the users performed tests by using two different switch sites. A questionnaire was also applied to evaluate the SITbench 1.0 itself.

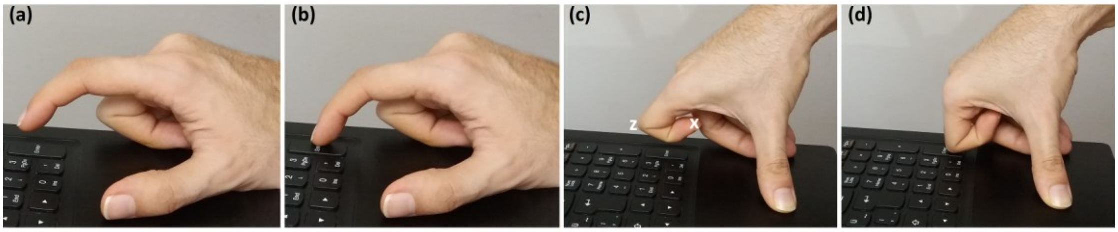


Figure 6.13: Positions of forefinger during experiments according to two switch sites FDP (represented by x) and FPIJ (represented by z): (a) switch press with FDP; (b) switch release with FDP; (c) switch press with FPIJ; (d) switch release with FPIJ.

In this section, firstly we introduce the participants. Then, we present the apparatus used and the procedure applied. At last, we share the experimental findings.

6.3.1 Participants

Eight able-bodied participants (mean age = 30.2, standard deviation = 3.1), including four females and four males, took part in this study. Just two of the participants were familiar with switch-accessible applications before experiments.

6.3.2 Apparatus

A laptop (Model: Lenovo G505S; CPU: AMD A8-4500M 1.9 GHz; RAM: 6 GB DDR3; Screen: LCD 15.6; OS: Windows 10 64 bits; Resolution: 1600 x 900) was employed within this study for experiments.

6.3.3 Procedure

At the beginning, the participant is positioned in front of a laptop in a way that the participant is able to access laptop's keyboard easily. *Enter* key on keyboard was considered as a switch.

Participants were informed about the SITbench 1.0 and tests, and then they practised the SITbench 1.0 in counterbalanced order until they become ready for tests. This practicing step took 20 minutes approximately for each participant. Following positioning and practicing steps, three tests were applied to participants to collect objective performance data:

- TSMG: Each switch site (FDP and FPIJ) was tested by each participant (n=8) for each template (n=5) two times where scanning time is 500 milliseconds (i.e., each participant performed 20 trials in total with TSMG).
- SS-NDG: Each switch site (FDP and FPIJ) was tested by each participant (n=8) for each track (n=5) where scanning time is 500 milliseconds (i.e., each participant performed 10 trials in total with SS-NDG).
- HFG: Each switch site (FDP and FPIJ) was tested by each participant (n=8) for each scenario (n=5) (i.e., each participant performed 10 trials in total with HFG).

All tests were applied in counterbalanced order to avoid learning and repetition effects. The participants were also allowed to rest (1 to 5 minutes) during experiments to prevent excessive mental or physical fatigue.

At the end of experiments, a SUS questionnaire [58], which is an industry standard, was applied to the participants to evaluate the usability of the SITbench 1.0 application. The SUS includes ten statements (Table 6.1) with a five-point Likert scale. Scale value of statements is ranging from 1 (strongly disagree) to 5 (strongly agree). We modified SUS statements according to the SITbench 1.0 to clearly describe it. On the other hand, qualitative subjective data was collected via our observations and participants' responses of open-ended questions about the SITbench 1.0.

Table 6.1: Modified statements of a SUS questionnaire with average scale values of all participants.

Statements	Scale
1. I would use the SITbench 1.0 for SIT evaluation tasks frequently.	4.00
2. I found the SITbench 1.0 unnecessarily complex.	1.37
3. I found the SITbench 1.0 easy to use.	4.12
4. I would need the support of a technical person to be able to use the SITbench 1.0.	1.75
5. I found the various functions in the SITbench 1.0 were well integrated.	4.37
6. I thought there was too much inconsistency in the SITbench 1.0.	1.37
7. I would imagine that most people would learn to use the SITbench 1.0 very quickly.	4.25
8. I found the SITbench 1.0 very cumbersome/awkward to use.	1.50
9. I felt very confident using the SITbench 1.0.	4.12
10. I need to learn a lot before I can use the SITbench 1.0.	1.25

6.3.4 Objective Data based Results

FDP as a switch site showed quite impressive performance in comparison with FPIJ in all three tests (TMSG, SS-NDG, HFG) as it is expected at the beginning. It is demonstrated that the SITbench 1.0 succeeded to determine the most appropriate switch site as FDP.

According to results of TSMG (Figure 6.14), FDP was better than FPIJ in all performance evaluation metrics (accuracy, precision, recall and false positive rate).

SS-NDG results (Figure 6.15) also suggested that FDP performed better than FPIJ in terms of the completion time and the crash count.

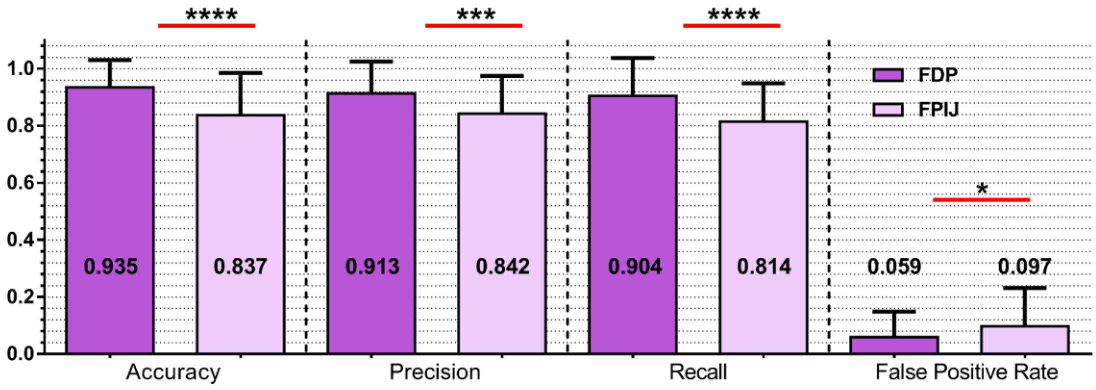


Figure 6.14: Mean values of two switch sites for all participants through evaluation metrics of TSMG (accuracy, precision, recall, false positive rate). (*$p < 0.05$; ***$p < 0.001$; ****$p < 0.0001$).

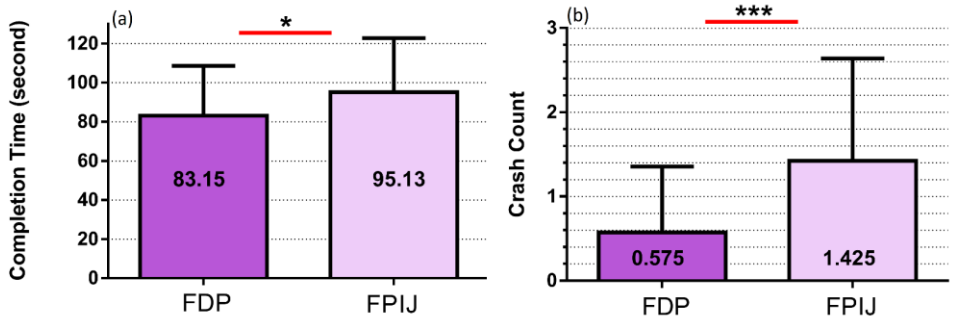


Figure 6.15: Mean values of two switch sites for all participants according to evaluation metrics of SS-NGD as (a) the completion time and (b) the crash count. (*$p < 0.05$; ***$p < 0.001$).

Lastly, HFG results (Figure 6.16) proved that FDP is by far the best switch site in all evaluation metrics (average press time, average release time, the fastest press time, the slowest press time, the fastest release time, the slowest release time). We also applied Student's t-test for both switch sites through all evaluation metrics in all three tests. In consequence of t-tests, it is proved that there is a significant difference between the performance of FDP and FPIJ for all evaluation metrics.

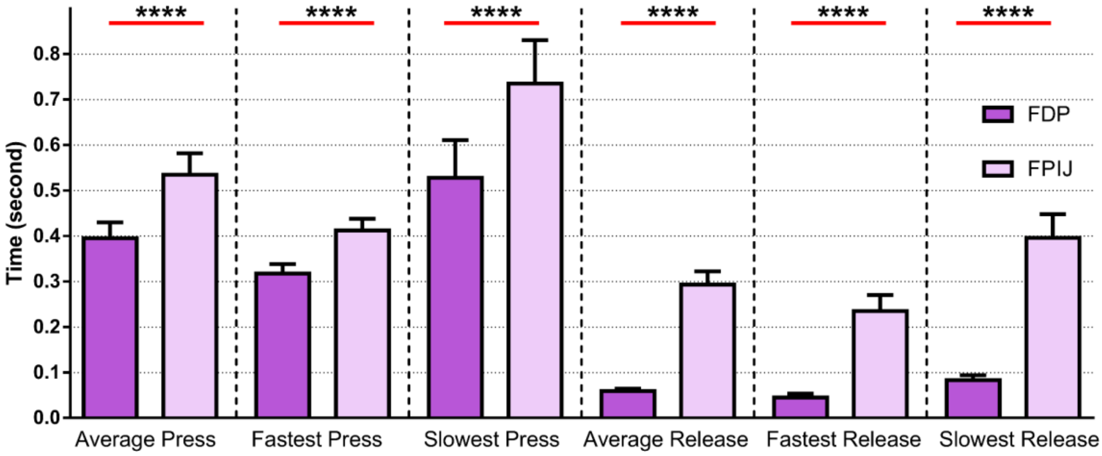


Figure 6.16: Mean values of two switch sites for all participants through evaluation metrics of HFG (average press time, average release time, the fastest press time, the slowest press time, the fastest release time, the slowest release time) (****$p < 0.0001$).

6.3.5 Subjective Data based Results

Results of a SUS questionnaire are listed in Table 6.1. Scale column holds the average scale values (1 to 5) of each statement for all participants. Average SUS score for all participants was calculated as 84. According to adjective rating scale [94], overall SUS score (84) of the SITbench 1.0 was rated as excellent, and SUS scores of each participant ranged from good to excellent. Prior to experiments, all participants were excited for experiments. Just two of them had a previous experience with SITs. They all declared that the SITbench 1.0 would be a very useful tool for assistive technology community. One participant stated that he could have performed better if scanning time was slower. Two of the participants suggested to increase the size of smileys in TSMG. All participants declared that FDP is definitely more proper than FPIJ as a switch site. None of the participants experienced fatigue during tests.

6.4 Conclusion and Discussion

Evaluation process is one of the most important tasks in order to reach the optimum SIT setup. Because the optimum SIT setup plays a vital role for people with motor disabilities to interact with their environment, any tool to achieve the optimum SIT setup for having a better cost and schedule management becomes a very important requirement considering the increasing number of the SIT users.

Determining the optimum switch setup by collecting the subjective data might be challenging, since the subjective data alone might be unreliable and manipulated easily for performance evaluation. Therapists might have to re-apply questionnaires and make new observations several times. A serious time and effort are needed for these repeated trials to collect subjective data. Therefore, subjective data collection instead of objective data doesn't seem a proper method for performance evaluation of a SIT.

On the other hand, current evaluation methods based on collecting objective data in literature are far from being a benchmark. These methods are generally employed to evaluate just a specific SIT. In other words, they are not designed to evaluate the other SITs, which makes them ineligible to be a benchmark. To the best of our knowledge, there have been just two applications [118, 119] in literature which are close to be a benchmark. The main limitations of these applications and solutions we proposed with the SITbench 1.0 are as follows: (a) They only work with the SITs that can emulate mouse left-click, which makes them compatible with just a minority of SITs for evaluation. The SITbench 1.0 as a benchmark allows to assign any expected characters or mouse-clicks from any SIT. By this way, all SITs which can emulate keyboard characters or mouse-clicks could be evaluated and compared via the SITbench 1.0 with standardized tests; (b) They only support single switch based systems. Because double switch usage is a widely preferred interaction technique, the SITbench 1.0 supports double switch evaluation as well; (c) They have only one test to measure press time and release time of a switch. The SITbench 1.0 has two more performance tests to evaluate SITs; (d) They don't allow to save the results into an external database, although they have some reporting functions. The SITbench 1.0 supports to save the result automatically into a database to share or analyse it for further studies. Therefore, we propose the SITbench 1.0 as a benchmark application that helps to determine the optimum SIT setup to provide a quicker and more accurate SIT evaluation process by collecting and saving the objective data automatically.

We conducted a usability study as a demonstration with eight participants to evaluate the usage of different switch sites. To this end, objective data was collected via the SITbench 1.0. FDP performed better performance than FPIJ in all tests as it is expected. Findings demonstrated that the SITbench 1.0 is capable of determining the most proper switch site with the aim of optimum SIT setup. Result of a SUS questionnaire to evaluate the SITbench 1.0 itself was also quite satisfactory.

A quicker and more accurate SIT evaluation via the SITbench 1.0 helps to serve more potential SIT users at the same time period with the same workforce. As a result of a better cost and schedule management, the SITbench 1.0 might prevent governments from unnecessary expenses and human-resource allocations. But, future studies with the SITbench 1.0 are required to verify that the SITbench 1.0 is capable of doing this. On the other hand, it might be also employed by therapists and assistive technology professionals to measure the fine-motor skills and reflexes of the users as a clinical tool. They can track the progress of the user's skill via the SITbench 1.0, since it is able to measure and save the performances automatically as a quantitative objective data. The SITbench 1.0 can also be utilized to improve the contingency awareness of the ones with profound and multiple learning disabilities. Besides, it might be employed as a tool to assess auditory and visual attention of people with severe learning difficulties.

In order to improve the SITbench 1.0 and overcome some of its limitations, some future studies would be quite useful. We intend to include new tests depending on several scanning methods. So as to test the efficiency of the SITbench 1.0 better, we aim to extend the participant group with motor-impaired people. Since the SITbench 1.0 is currently compatible with only desktop computers, it might be modified to be compatible with mobile systems such as smartphones and tablets to extend the target group. We also aim to include some tests such as a speller to evaluate the users' computer access activities. Employing a group of therapists and assistive technology professionals to evaluate and demonstrate the SITbench 1.0 would be also quite useful.

7

Conclusion and Outlook

Through this thesis, we addressed the single-gesture based hands-free computer access problem of the people with severe motor-neuron impairments to improve their self-sufficiency. In this chapter, we summarize and discuss the main contributions in Section 7.1 under three domains through chapters by revisiting the research questions. Subsequently, an overview of the possible future expansions is given in Section 7.2. Finally, we conclude the thesis with remarks about the work presented within this thesis.

7.1 Summary of the Thesis and Contributions

Software Switches: Single-gesture based Hands-free Interaction Techniques

Chapter 2:

Our efforts to provide efficient solutions to single-gesture based hands-free computer access problem led us to focus on the following research question:

Q1: How to devise an efficient approach enabling single-gesture based hands-free HCI?

We started with a literature review to handle this question. To have a broader point of view, we reviewed the current hands-free interaction techniques —instead of reviewing just the single-gesture based solutions— in terms of the physical gestures used. This review helped us to identify two major problems of the current single-gesture based hands-free interaction techniques. The first problem is that the majority of current single-gesture based hands-free HCI techniques depend on expensive dedicated devices beyond standard computer peripherals. Unfortunately, high-cost of dedicated devices became a new barrier for the people with motor-disabilities. Many people living in poor and middle income countries have difficulties to afford such devices [7-9]. Thus, as the only reasonable exception, we decided to exclude smartphones from dedicated devices list because the total number of smartphones —3.2 billion in 2019 [48]— got ahead of the total number of computers in recent years worldwide [49]. Smartphones became easy to access devices like computer peripherals today even for the ones live in low-income countries. Furthermore, unlike the dedicated devices, smartphones can also be used to achieve several services beyond assistive purposes.

The second problem is that the majority of current single-gesture based hands-free solutions for computer access in literature are only compatible with specific switch-accessible interfaces. While some switch-accessible interfaces expect to receive a specific keyboard character like space, the others expect to receive a mouse click. Unfortunately, a commonly agreed standard on this is lacking. To overcome these two major problems, we proposed our novel software switch approach. In brief, the software switch approach has two principles: an interaction technique based on our software switch approach (1) should not require any dedicated devices, and (2) should be configurable to be compatible with the other switch-accessible interfaces.

After we proposed our software switch approach, in Chapter 2, we also aimed to identify the most suitable types of hands-free gestures which will be used with the proposed software switches. To do this, we began with the evaluation of gesture types of current hands-free interaction techniques given in Section 2.1. We omitted the HCI techniques which require dedicated devices in accordance with the first principle of our software switch approach, and we focused on the techniques which don't require dedicated devices.

Afterwards, we excluded eye-gaze and facial gestures (e.g., eye blink gesture), since they might be highly affected by Midas Touch problem [50]. For example, if the eye blink is used to interact with a computer, it is hard to distinguish whether the user blinked voluntary as a trigger signal or it was just a regular eye blink performed unconsciously. Although the Midas Touch problem could be resolved by employing multi-modal inputs such as performing an eye blink with an eye brow raising simultaneously as a trigger signal, employing a multi-modal input was not an option in the scope of this thesis since we focused on the single-gesture based hands-free computer access problem.

As another gesture types in literature, the speech and non-speech based solutions were also omitted by us, because the speech recognition algorithms are mostly preferred for more complicated tasks than just a single-gesture. Unlike these gestures, we revealed that the respiration (i.e., sip and puff) and head movement (e.g., a head tilt) gestures are not affected by Midas Touch problem, and they can be easily recognized via simple algorithms. Therefore, we considered them as the most suitable hands-free gesture types among existing solutions to interact with a computer according to the software switch approach.

Chapters 3 and 4:

Following the detection of the most suitable gestures, next step was to devise efficient interaction methods by following the principles of our software switch approach to interact with a computer by a single puff or head gesture. To do this, we began with addressing the following research questions:

Q2: How to devise a better technique that enables a person to interact with a computer by a single puff-gesture?

Q3: How to devise a better technique that enables a person to interact with a computer by a single head-gesture?

To answer Q2, we designed two novel interaction techniques called the PuffMic and the PuffCam software switches in Chapter 3. Both software switches are based on a puff-gesture where a strong puff, detected by a microphone or a modified camera, is considered as a pressed switch. In Chapter 4, in accordance with our efforts to answer Q3, we designed two more novel interaction techniques called the HeadCam and the HeadGyro software switches. Both software switches are capable of recognizing head gestures (e.g., a head tilt) via a standard camera or a gyroscope sensor of a smartphone to consider them as pressed switches.

To sum up, all software switches proposed within this thesis can allow to interact with a computer by a single-gesture in a way that they receive the user's gesture as an input signal to translate them into emulated switch presses. Furthermore, they don't require any dedicated device, and they are compatible with the other switch-accessible interfaces in accordance with our software switch approach. The high usability of the proposed switches was demonstrated by two different studies conducted with 82 participants with/out disabilities in total. As a result of these usability studies via the TSMG test of the SITbench 1.0, we demonstrated that the performance of the proposed software switches through evaluation metrics —accuracy, precision, recall and false positive rate— were quite impressive. For each evaluation metrics, the PuffMic exhibited the worst performance, whereas the HeadGyro was the best. According to our observations, the reasons why the PuffMic performed the worst is considered as follows:

- The position of the headset microphone should be stable during experiments since the threshold value is assigned in the calibration step and highly depends on the initial position of the microphone. Any possible change in the position (angle or distance between microphone and the user's mouth or nose) during the experiments might lead to a minor or a major change in the air pressure (originated from the exhalation pressure) applied on the microphone, which is considered as the main reason of high false positive rate.

Besides, each software switch exhibited different reactions to the external factors. While the HeadGyro and the PuffMic are not affected by the light level (i.e., they can even work in the darkness), the PuffCam and the HeadCam are affected. As a microphone based solution, the PuffMic can be affected by high external voices and the user's speech unlike the other software switches. Additionally, the HeadGyro is more sensitive than the HeadCam since it is capable of recognizing tiny head movements.

The GLOSTER 1.0: A Single-gesture Accessible Hands-free CAT

Chapter 5:

On the other hand, the proposed software switches were not functional alone to control a computer, since they can only serve like traditional switches by recognizing a single head or puff gesture to translate them into emulated switch presses. In other words, they can only allow to interact with a computer.

As a general approach, computer access depends on three functions: pointing, clicking, and typing. In conventional use, pointing and clicking functions are performed by interacting with a computer via a standard mouse, while typing is achieved by a keyboard. But, if the user has an only single-gesture unimpaired, all three functions have to be performed with that single-gesture/single-switch. To do this, the proposed software switches require a CAT with a scanning interface which converts the emulated switch presses into commands (e.g., mouse clicks) in a way that a computer can understand. In accordance with this requirement, we focused on the following research question in Chapter 5:

Q4: How to devise a better CAT that enables a person to control a computer with a single-gesture?

This question led us to a new single-switch accessible CAT called the GLOSTER 1.0 with a novel mouse pointing technique. It allows the users to easily control a computer by employing the proposed software switches or any other traditional switches. The proposed mouse pointing technique called the CoorP also provided a more accurate solution to the single-switch based mouse pointing problem than the most popular mouse pointing technique in literature.

The SITbench 1.0: An Evaluation Tool

<u>Chapter 6:</u>

In the evaluation stage of the proposed software switches, we realized the lack of a reliable evaluation tool. Evaluation process of a SIT —like the proposed software switches— requires an interdisciplinary team effort and takes a considerable amount of time, since a SIT setup depends on many variables such as switch type, switch position or switch site. Optimum SIT setup (i.e., the most appropriate combination of setup variables) could not be achieved at first attempts.

Although subjective evaluation is a very common approach, we considered that the subjective evaluation data alone might be manipulated and unreliable for comparing performances in many cases. It is hard to evaluate the measurable performance by collecting subjective data instead of objective data. Unfortunately, the existing objective evaluation methods are far from being a benchmark where the other SITs could be evaluated via standardized test. They are mostly designed to evaluate just a specific SIT. Requirement of a useful tool to evaluate our software switches objectively led us to focus on the following research question in Chapter 6:

Q5: How to devise a better tool that enables objective evaluation of switch-based interaction techniques?

In an effort to address Q5, we proposed a novel benchmark tool for objective evaluation called the SITbench 1.0. It was demonstrated by a usability study with 8 participants that the SITbench 1.0 provides a quicker and more accurate switch evaluation process by collecting the objective data automatically.

7.2 Outlook

Within the scope of this thesis, novel hands-free interaction techniques and alternative computer access solutions were presented in accordance with our efforts to answer the research questions in Section 1.3. However, there are still possible future expansions for the proposed studies. We identify potential research directions below that can be followed in the future.

The proposed software switches are mainly based on simple audio and video processing algorithms to recognize the user's gesture. As a future research direction, machine learning techniques can be employed to recognize these gestures. Such a direction would allow to achieve better recognition performance or the opposite.

While the PuffMic and the PuffCam serve as alternatives to traditional puff switches, the HeadCam and the HeadGyro are alternatives to traditional head switches. In another future work, a usability study to compare the proposed software switches with the traditional alternatives would be useful to see how usable the proposed software switches.

Employing a unidirectional-cardioid microphone was out of option within this thesis because they are expensive and beyond the standard computer peripherals. In a future study, they would be employed to reject sounds from other directions if high-cost of unidirectional-cardioid microphones can be tolerated.

We compared the performance of CG with DG to see whether there is any significant difference between these groups, because we aimed that our approach will be utilized by able-bodied people as well. The application area of the proposed interaction techniques is quite flexible and should not be considered just for assistive technology area, although we aimed to provide comprehensive solutions for motor-impaired people within this thesis. For example, in a multi-modal computer video game, one can try to employ our software switches in a way that players use their puff/head gesture as a new interaction method during the game-play.

Within this thesis, we employed the proposed software switches in the SITbench 1.0 and the GLOSTER 1.0. As another future work, the proposed software switches would be employed in a different single-switch accessible interface to provide a further validation in terms of their compatibility in real-life scenarios. Another future work would be making the proposed software switches and interfaces compatible with the other operating systems like Macintosh to improve the accessibility. Currently, they are only compatible with Microsoft-based operating systems.

It is also worth noting that the HeadCam and the HeadGyro are also capable of recognizing any motion of the other body parts such as the user's shoulder or leg beyond head movements. In a future work, different physical movements can be targeted depending on the user's unimpaired physical ability to evaluate the flexibility of the proposed software switches.

The GLOSTER 1.0 is based on an automatic row-column scanning method to type text in a way that the user has to type all letters of a word. Current text entry method can be accelerated by a prediction technique where a list of possible words is offered to the user as a new letter typed. Moreover, the alphabet in the speller menu can be designed more wisely according to the frequency of the letters. By this way, the user would achieve a better text input rate.

A further future expansion would be devising a mark-up language which allows to perform pre-defined tasks in GLOSTER 1.0, for example the command "<browser>"

typed by the user will open the web browser and etc. Such commands would increase the usability of the CAT.

The SITbench 1.0, in its current form, is capable of providing standardized tests with single and double switch access. Development of new standardized tests with multi-switch access would be another research direction to improve the proposed evaluation benchmark.

Besides, it would be interesting to employ the SITbench 1.0 for clinical purposes to evaluate the usability and the efficiency. Occupational therapists can evaluate fine-motor skills of clients as a clinical tool. They can track the patients' progress by the SITbench 1.0 that allows to measure and record clients' fine motor performance and reflexes automatically in the form of quantitative objective data. Additionally, the SITbench 1.0 might also help to improve the contingency awareness of the ones with profound and multiple learning disabilities, or it might be useful for pupils with severe learning difficulties to assess their auditory and visual attention.

7.3 Final Remarks

Within this thesis, we aimed to provide comprehensive solutions to single-gesture based hands-free computer access problem. Considering the diversity of computer applications and their various useful services, the ability of operating a computer gives people with motor-impairments crucial capabilities for a more independent life; which highly motivated us to investigate new interaction techniques and interfaces in order to be able to enhance these people's quality of life.

We strongly believe in the GLOSTER 1.0 and the proposed software switches to have a marked impact on people's lives, since they are together capable of enabling the single-gesture based hands-free computer access without purchasing any dedicated device or software. As low-cost alternatives, the proposed software switches can be preferred instead of expensive puff and head switches. Currently, they are the only switch options for the ones who can control their puff or head gestures but cannot afford any dedicated devices. Besides, since a quicker and more accurate evaluation process is provided by the SITbench 1.0, more switch users can be evaluated at the same time period with the same workforce; which leads to a better cost and schedule management.

8 References

1. Drainoni, M.L., et al., *Patterns of internet use by persons with spinal cord injuries and relationship to health-related quality of life.* Archives of Physical Medicine and Rehabilitation, 2004. **85**(11): p. 1872-1879.
2. ScanBuddy. Available from: https://web.archive.org/web/20190429175834/http://www.ahf-net.com/scanbuddy/ (Accessed on 24.11.2019).
3. SwitchXS. Available from: https://web.archive.org/web/20190518141658/http://www.orin.com/access/switchxs/ (Accessed on 24.11.2019).
4. World Report on Disability. Available from: https://web.archive.org/web/20191110112641/https://www.who.int/disabilities/world_report/2011/report.pdf (Accessed on 24.11.2019).
5. Armour, B.S., et al., *Prevalence and causes of paralysis—United States, 2013.* American Journal of Public Health, 2016. **106**(10): p. 1855-1857.
6. International Labour Organization. Available from: https://web.archive.org/web/20180425014312/http://www.ilo.org/wcmsp5/groups/public/---dgreports/---dcomm/documents/publication/wcms_087707.pdf (Accessed on 24.11.2019).
7. Borg, J. and Östergren, P.O., *Users' perspectives on the provision of assistive technologies in Bangladesh: awareness, providers, costs and barriers.* Disability and Rehabilitation: Assistive Technology, 2015. **10**(4): p. 301-308.
8. de Witte, L., et al., *Assistive technology provision: towards an international framework for assuring availability and accessibility of affordable high-quality assistive technology.* Disability and Rehabilitation: Assistive Technology, 2018. **13**(5): p. 467-472.
9. Li, W. and Sellers, C., *Improving assistive technology economics for people with disabilities: harnessing the voluntary and education sectors*. in *2009 IEEE*

Toronto International Conference Science and Technology for Humanity (TIC-STH). 2009. IEEE.

10. Cecotti, H., *Spelling with non-invasive brain–computer interfaces–current and future trends.* Journal of Physiology, 2011. **105**(1-3): p. 106-114.

11. Guan, C., Thulasidas, M., and Wu, J., *High performance P300 speller for brain-computer interface*. in *2004 IEEE International Workshop on Biomedical Circuits and Systems.* 2004. IEEE.

12. Borghetti, D., et al., *A low-cost interface for control of computer functions by means of eye movements.* Computers in Biology and Medicine, 2007. **37**(12): p. 1765-1770.

13. Lv, Z., et al., *Implementation of the EOG-based human computer interface system*. in *2008 2nd International Conference on Bioinformatics and Biomedical Engineering*. 2008. IEEE.

14. Smith, J.D. and Graham, T., *Use of eye movements for video game control*. in *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*. 2006. ACM.

15. Zeng, Z. and Roetting, M., *A text entry interface using smooth pursuit movements and language model*. in *Proceedings of the 2018 ACM Symposium on Eye Tracking Research and Applications*. 2018. ACM.

16. Easy Flex Dual Ultimate Switch. Available from: http://web.archive.org/web/20191114040944/https://enablingdevices.com/product/easy-flex-dual-ultimate-switch/ (Accessed on 24.11.2019).

17. Gerdtman, C., Bäcklund, Y. and Lindén, M., *A gyro sensor based computer mouse with a usb interface: a technical aid for motor-disabled people.* Technology and Disability, 2012. **24**(2): p. 117-127.

18. Zhang, T., Li, L. and Yan. H., *The HCI method for upper limb disabilities based on EMG and gyros*. in *2014 IEEE 13th International Workshop on Advanced Motion Control (AMC)*. 2014. IEEE.

19. SmartNav. Available from: https://web.archive.org/web/20190607100209/http://www.naturalpoint.com/smartnav/ (Accessed on 24.11.2019).

20. Honye, S. and Thinyane, S., *WiiMS: Simulating mouse and keyboard for motor-impaired users*. in *2012 Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*. 2012. ACM.

21. Quha Pufo. Available from: http://web.archive.org/web/20191205065344/https://www.quha.com/products-2/quha-pufo/ (Accessed on 05.12.2019).

22. Imperium Sip and Puff Switch. Available from: https://web.archive.org/web/20191226213339/https://www.ablenetinc.com/technology/switches/imperium-sip-and-puff-switch (Accessed on 26.12.2019).

23. Gerdtman, C. and Lindén, M., *Six-button click interface for a disabled user by an adjustable multi-level sip-and-puff switch*. in *Proceedings of SIGRAD 2010 Content Aggregation and Visualization.* 2010. Linköping University Electronic Press.

24. Felzer, T., et al., *Qanti: a software tool for quick ambiguous non-standard text input*. in 2010 *International Conference on Computers for Handicapped Persons*. 2010. Springer.

25. Huang, C.N., Chen, C.H. and Chung, H.Y., *Application of facial electromyography in computer mouse access for people with disabilities.* Disability and Rehabilitation, 2006. **28**(4): p. 231-237.

26. Kim, J., et al., *The tongue enables computer and wheelchair control for people with spinal cord injury.* Science Translational Medicine, 2013. **5**(213): p. 213ra166

27. Struijk, L.N.A., *An inductive tongue computer interface for control of computers and assistive devices.* IEEE Transactions on Biomedical Engineering, 2006. **53**(12): p. 2594-2597.

28. Simpson, T., et al., *Tooth-click control of a hands-free computer interface.* IEEE Transactions on Biomedical Engineering, 2008. **55**(8): p. 2050-2056.

29. LipSync. Available from: http://web.archive.org/web/20191205065647/https://www.makersmakingchange.com/project/lipsync/ (Accessed on 05.12.2019).

30. Jouse2. Available from: https://web.archive.org/web/20191112144721/http://teknyka.com/Jouse2/ (Accessed on 24.11.2019).

31. Sturtz, C.R., *Mouth-operated computer input device and associated methods*. 2010. Google Patents.

32. TetraMouse. Available from: http://web.archive.org/web/20191205070138/https://tetramouse.com/ (Accessed on 05.12.2019).

33. Ultra Joystick. Available from: http://web.archive.org/web/20191205054750/http://www.zyteq.com.au/products/computer_accessing/ultra-joystick (Accessed on 05.12.2019).

34. Chin Switch. Available from: https://web.archive.org/web/20191226213635/https://www.talktometechnologies.com/collections/switches/products/chin-switch (Accessed on 26.12.2019).

35. Kim, E.Y., et al., *Eye mouse: mouse implementation using eye tracking*. in *2005 ICCE Digest of Technical Papers. International Conference on Consumer Electronics.* 2005. IEEE.

36. Betke, M., Gips, J. and Fleming, P., *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, 2002. **10**(1): p. 1-10.

37. Kim, H. and Ryu, D., *Computer control by tracking head movements for the disabled*. in *2006 International Conference on Computers for Handicapped Persons*. 2006. Springer.

38. Varona, J., Manresa-Yee, C. and Perales, F.J., *Hands-free vision-based interface for computer accessibility.* Journal of Network and Computer Applications, 2008. **31**(4): p. 357-374.

39. ACAT Vision. Available from: https://web.archive.org/web/20200102215722/https://01.org/acat/switches (Accessed on 02.01.2020).

40. Grauman, K., et al., *Communication via eye blinks and eyebrow raises: video-based human-computer interfaces.* Universal Access in the Information Society, 2003. **2**(4): p. 359-373.

41. Dai, L., et al., *Speech-based cursor control: a study of grid-based solutions*. in *2004 ACM SIGACCESS. Accessibility and Computing*. 2004. ACM.

42. Karimullah, A.S. and Sears, A., *Speech-based cursor control*. in *2002 Proceedings of the 5th International ACM Conference on Assistive Technologies*. 2002. ACM.

43. Manaris, B., McCauley, R. and MacGyvers, V., *An intelligent interface for keyboard and mouse control*. in *2001 Proc. 14th Int'l Florida AI Research Symposium (FLAIRS-01)*. 2001. Citeseer.

44. Harada, S., et al., *The vocal joystick: evaluation of voice-based cursor control techniques*. in *2006 Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*. 2006. ACM.

45. Sporka, A.J., Kurniawan, S.H. and Slavík, P., *Acoustic control of mouse pointer.* Universal Access in the Information Society, 2006. **4**(3): p. 237-245.

46. Sporka, A., Kurniawan, S.H. and Slavík, P., *Non-speech operated emulation of keyboard*, in *2006 Designing Accessible Technology.* 2006. Springer.

47. Zielasko, D., et al., *Blowclick: A non-verbal vocal input metaphor for clicking*. in *2015 Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. 2015. ACM.

48. Statista: Smartphone Users Worldwide 2016-2021. Available from: https://web.archive.org/web/20191120021523/https://www.statista.com/statisti

cs/330695/number-of-smartphone-users-worldwide/ (Accessed on 24.11.2019).

49. Bröhl, C., et al., *Desktop PC, tablet PC, or smartphone? An analysis of use preferences in daily activities for different technology generations of a worldwide sample*. in *2018 International Conference on Human Aspects of IT for the Aged Population*. 2018. Springer.

50. Jacob, R.J., *What you look at is what you get: eye movement-based interaction techniques*. in *1990 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1990. ACM.

51. Sip-and-puff Devices by Orin. Available from: https://web.archive.org/web/20190609132645/http://www.orin.com/access/sip_puff/ (Accessed on 05.12.2019).

52. Sip-and-puff Switch by Enabling Devices. Available from: https://web.archive.org/web/20191206045010/https://enablingdevices.com/product/sip-and-puff-switches/ (Accessed on 05.12.2019).

53. Mace, M., et al., *A heterogeneous framework for real-time decoding of bioacoustic signals: applications to assistive interfaces and prosthesis control.* Expert Systems with Applications, 2013. **40**(13): p. 5049-5060.

54. Sullivan, A., Bailey, J. and Stokic, D., *Predictors of oral health after spinal cord injury.* Spinal Cord, 2013. **51**(4): p. 300-305.

55. Esiyok, C. and Albayrak, S., *SITbench 1.0: a novel switch-based interaction technique benchmark.* Journal of Healthcare Engineering, 2019. https://doi.org/10.1155/2019/5075163

56. Fu, Y. and Huang, T.S., *hMouse: head tracking driven virtual computer mouse*. in *2007 IEEE Workshop on Applications of Computer Vision (WACV'07)*. 2007. IEEE.

57. Gorodnichy, D., Dubrofsky, E. and Mohammad, A., *Working with a computer hands-free using the nouse perceptual vision interface.* in *2007. International Workshop on Video Processing and Recognition (VideoRec’07)*. 2007. Canada.

58. Brooke, J., *SUS-a quick and dirty usability scale.* Usability Evaluation in Industry, 1996. **189**(194): p. 4-7.

59. Quha Zono. Available from: https://web.archive.org/web/20200707154711/https://www.quha.com/products-2/zono/ (Accessed on 07.07.2020).

60. Antunes, R.A., et al., *Intelligent HCI device for assistive technology*. in *2019 Doctoral Conference on Computing, Electrical and Industrial Systems*. 2019. Springer.

61. Blackmon, F.R. and Weeks, M., *Target acquisition by a hands-free wireless tilt mouse*. in *2009 IEEE International Conference on Systems, Man and Cybernetics*. 2009. IEEE.

62. Castillo, A., et al., *Hands free mouse*. in *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. 2016. IEEE.

63. Raya, R., et al., *Wearable inertial mouse for children with physical and cognitive impairments.* Sensors and Actuators A: Physical, 2010. **162**(2): p. 248-259.

64. Rosas-Cholula, G., et al., *Gyroscope-driven mouse pointer with an Enotiv® EEG headset and data analysis based on empirical mode decomposition.* Sensors, 2013. **13**(8): p. 10561-10583.

65. Sancha-Ros, S. and García-Garaluz, E., *Computer access and alternative and augmentative communication (AAC) for people with disabilities: a multi-modal hardware and software solution*. in *2015 International Work-Conference on Artificial Neural Networks*. 2015. Springer.

66. Sim, N., et al., *The head mouse—head gaze estimation" in-the-wild" with low-cost inertial sensors for BMI use*. in *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*. 2013. IEEE.

67. wook Kim, Y. and hyun Cho, J., *A novel development of head-set type computer mouse using gyro sensors for the handicapped*. in *2nd Annual International*

IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology Proceedings (Cat. No. 02EX578). 2002. IEEE.

68. Headmaster Plus. Available from: https://web.archive.org/web/20191221235644/https://abledata.acl.gov/product/headmaster-plus-model-hm-2p/ (Accessed on 24.11.2019).

69. LoPresti, E.F., Brienza, D.M. and Angelo, J., *Head-operated computer controls: effect of control method on performance for subjects with and without disability.* Interacting with Computers, 2002. **14**(4): p. 359-377.

70. Andres, R.O. and Hartung, K.J., *Prediction of head movement time using fitts' law.* Human Factors, 1989. **31**(6): p. 703-713.

71. Alhamzawi, H.A., *Control mouse cursor by head movement: development and implementation.* Applied Medical Informatics, 2018. **40**(3-4): p. 39-44.

72. El-Afifi, L., et al., *'Hands-free interface'-a fast and accurate tracking procedure for real time human computer interaction*. in *2004 Proceedings of the Fourth IEEE International Symposium on Signal Processing and Information Technology.* 2004. IEEE.

73. Gorodnichy, D.O., Malik, S. and Roth, G., *Nouse 'use your nose as a mouse'-a new technology for hands-free games and interfaces*. in *2002 Proc. Intern. Conf. on Vision Interface (VI'2002)*. 2002.

74. Javanovic, R. and MacKenzie, I.S., *Markermouse: mouse cursor control using a head-mounted marker*. in *2010 International Conference on Computers for Handicapped Persons*. 2010. Springer.

75. Kim, S., et al., *Head mouse system based on gyro-and opto-sensors*. in *2010 3rd International Conference on Biomedical Engineering and Informatics*. 2010. IEEE.

76. Kjeldsen, R., *Improvements in vision-based pointer control*. in *2006 Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*. 2006. ACM.

77. Kumar, R. and Kumar, A., *Black pearl: an alternative for mouse and keyboard.* Journal on Graphics, Vision and Image Processing, 2008. **8**: p. 1-6.

78. Lin, Y.P., et al., *Webcam mouse using face and eye tracking in various illumination environments*. in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*. 2006. IEEE.

79. Loewenich, F. and Maire, F., *Hands-free mouse-pointer manipulation using motion-tracking and speech recognition*. in *2007 Proceedings of the 19th Australasian Conference on Computer-Human Interaction: Entertaining User Interfaces*. 2007. ACM.

80. Manresa-Yee, C., Varona, J. and Perales, F.J., *Towards hands-free interfaces based on real-time robust facial gesture recognition*. in *2006 International Conference on Articulated Motion and Deformable Objects*. 2006. Springer.

81. Morris, T. and Chauhan, V., *Facial feature tracking for cursor control.* Journal of Network and Computer Applications, 2006. **29**(1): p. 62-80.

82. Pallejà, T., et al., *Using the optical flow to implement a relative virtual mouse controlled by head movements.* J. UCS, 2008. **14**(19): p. 3127-3141.

83. Perini, E., et al., *FaceMouse: a human-computer interface for tetraplegic people*. in *2006 European Conference on Computer Vision*. 2006. Springer.

84. Song, Y., Luo, Y. and Lin, J., *Detection of movements of head and mouth to provide computer access for disabled*. in *2011 International Conference on Technologies and Applications of Artificial Intelligence*. 2011. IEEE.

85. Strumiłło, P. and Pajor, T., *A vision-based head movement tracking system for human-computer interfacing*. in *2012 Joint Conference New Trends In Audio & Video And Signal Processing: Algorithms, Architectures, Arrangements And Applications (NTAV/SPA)*. 2012. IEEE.

86. Su, M.C., Su, S.Y. and Chen, G.D., *A low-cost vision-based human-computer interface for people with severe disabilities.* Biomedical Engineering: Applications, Basis and Communications, 2005. **17**(06): p. 284-292.

87. Toyama, K., *Look, ma-no hands! hands-free cursor control with real-time 3D face tracking.* PUI98, 1998.

88. Tu, J., Huang, T. and Tao, H., *Face as mouse through visual face tracking*. in *2005 The 2nd Canadian Conference on Computer and Robot Vision (CRV'05)*. 2005. IEEE.

89. HeadMouse Nano. Available from: https://web.archive.org/web/20191006105058/https://www.orin.com/access/headmouse/ (Accessed on 24.11.2019).

90. TrackerPro. Available from: https://web.archive.org/web/20160731023751/https://www.ablenetinc.com/trackerpro (Accessed on 24.11.2019).

91. Walsh, E., Daems, W. and Steckel, J., *An optical head-pose tracking sensor for pointing devices using IR-LED based markers and a low-cost camera*. in *2015 IEEE SENSORS*. 2015. IEEE.

92. Li, S., Ngan, K.N. and Sheng, L., *A head pose tracking system using RGB-D camera*. in *2013 International Conference on Computer Vision Systems*. 2013. Springer.

93. Head Switch. Available from: http://web.archive.org/web/20191125002431/https://enablingdevices.com/product/head-switch/ (Accessed on 11.12.2019).

94. Bangor, A., Kortum, P. and Miller, J., *Determining what individual SUS scores mean: Adding an adjective rating scale.* Journal of Usability Studies, 2009. **4**(3): p. 114-123.

95. Assistive Context-Aware Toolkit (ACAT). Available from: http://web.archive.org/web/20191226190205/https://01.org/acat/ (Accessed on 26.12.2019).

96. GRID 3. Available from: https://web.archive.org/web/20200102220138/https://thinksmartbox.com/product/grid-3/computer-control/ (Accessed on 02.01.2020).

97. EZ Keys. Available from: http://web.archive.org/web/20200102221025/http://www.techaccess4you.com/ez-keys-software/ (Accessed on 02.01.2020).

98. WinSCAN 3.0 CS. Available from: https://web.archive.org/web/20170128015156/http://www.acsw.com/ws1.html (Accessed on 02.01.2020).

99. Qualiworld. Available from: http://web.archive.org/web/20200102222223/https://abledata.acl.gov/product/qualiworld-basic (Accessed on 02.01.2020).

100. Zhao, X.A., et al., *Typing with eye-gaze and tooth-clicks*. in *2012 Proceedings of the Symposium on Eye Tracking Research and Applications*. 2012. ACM.

101. Lafi, S.M. and Hock, S.K.B., *An adaptive text entry method based on single-key minimal scan matrix for people with severe motor disabilities.* International Journal of Scientific & Engineering Research, 2012. **3**(8): p. 1-5.

102. Prabhu, V. and Prasad, G., *Designing a virtual keyboard with multi-modal access for people with disabilities*. in *2011 World Congress on Information and Communication Technologies*. 2011. IEEE.

103. MacKenzie, I.S. and Ashtiani, B., *BlinkWrite: efficient text entry using eye blinks.* Universal Access in the Information Society, 2011. **10**(1): p. 69-80.

104. Song, Y.C., *Joystick text entry with word prediction for people with motor impairments*. in *2010 Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility*. 2010. ACM.

105. SofType 5. Available from: https://web.archive.org/web/20200102222909/https://www.orin.com/access/softype/ (Accessed on 02.01.2020).

106. KeyStrokes. Available from: https://web.archive.org/web/20200102223101/https://www.orin.com/access/keystrokes/ (Accessed on 02.01.2020).

107. Clicker 7. Available from: https://web.archive.org/web/20200102223344/https://www.cricksoft.com/uk/clicker/learn-more/accessibility (Accessed on 02.01.2020).

108. CrossScanner. Available from: https://web.archive.org/web/20200103164255/http://rjcooper.com/cross-scanner/ (Accessed on 03.01.2020).

109. Gus Scanning Cursor. Available from: https://web.archive.org/web/20200103170003/https://www.turningpointtechnology.com/Software/GS/SC.asp (Accessed on 03.01.2020).

110. Tsui, C.S.L., et al., *EMG-based hands-free wheelchair control with EOG attention shift detection*. in *2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. 2007. IEEE.

111. Triccas, L.T., et al., *Is there a standard procedure for assessing and providing assistive devices for people with neuro-disabling conditions in United Kingdom? A nation-wide survey.* Disability and Health Journal, 2019. **12**(1): p. 93-97.

112. Glickman, L., et al., *The effect of switch control site on computer skills of infants and toddlers.* American Journal of Occupational Therapy, 1996. **50**(7): p. 545-553.

113. Hoppenbrouwers, G., Stewart, H. and Kernot, J., *Determining the content validity, inter-rater reliability and usability of the novita switch record form.* Technology and Disability, 2016. **28**(1, 2): p. 1-11.

114. Kubota, M., et al., *New ocular movement detector system as a communication tool in ventilator-assisted werdnig-hoffmann disease.* Developmental Medicine & Child Neurology, 2000. **42**(1): p. 61-64.

115. Sax, C., Fisher, D. and Pumpian, I., *Outcomes for students with severe disabilities: case studies on the use of assistive technology in inclusive classrooms.* Technology and Disability, 1996. **5**(3-4): p. 327-334.

116. Shull, J., et al., *Assistive technology programming for a young child with profound disabilities: a single-subject study.* Physical & Occupational Therapy in Pediatrics, 2004. **24**(4): p. 47-62.

117. Dumont, C., Vincent, C. and Mazer, B., *Development of a standardized instrument to assess computer task performance.* American Journal of Occupational Therapy, 2002. **56**(1): p. 60-68.

118. Koester, H.H., et al., *Compass: software for computer skills assessment*. in *2003 CSUN International Conference on Technology and Persons with Disabilities.* 2003. Los Angeles, CA.

119. Liffick, B., Romich, B. and Hill, K., *A single-switch performance evaluation tool*. in *2005 Proceedings of the 11th International Conference on Human-Computer Interaction. 2*005. Las Vegas, NV. 2

120. Standen, P.J., et al., *An evaluation of the wii nunchuk as an alternative assistive device for people with intellectual and physical disabilities using switch controlled software.* Computers & Education, 2011. **56**(1): p. 2-10.

121. Chan, J., et al., *Evaluation of a non-invasive vocal cord vibration switch as an alternative access pathway for an individual with hypotonic cerebral palsy–a case study.* Disability and Rehabilitation: Assistive Technology, 2010. **5**(1): p. 69-78.

122. Esiyok, C., et al., *Software switches: novel hands-free interaction techniques for quadriplegics based on respiration–machine interaction.* Universal Access in the Information Society, 2019. p. 1-13. DOI: 10.1007/s10209-019-00645-2.