



Article

A Human–Computer Interface Replacing Mouse and Keyboard for Individuals with Limited Upper Limb Mobility

Diyar Gür ¹, Niklas Schäfer ², Mario Kupnik ² and Philipp Beckerle ^{3,4,*}

¹ Department of Electrical Engineering and Information Technology, Technical University of Darmstadt, 64283 Darmstadt, Germany; diyar.guer@stud.tu-darmstadt.de

² Measurement and Sensor Technology, Department of Electrical Engineering and Information Technology, Technical University of Darmstadt, 64283 Darmstadt, Germany; niklas.schaefer1@tu-darmstadt.de (N.S.); mario.kupnik@tu-darmstadt.de (M.K.)

³ Elastic Lightweight Robotics Group, Robotics Research Institute, Department of Electrical Engineering and Information Technology, Technische Universität Dortmund, 44227 Dortmund, Germany

⁴ Institute for Mechatronic Systems in Mechanical Engineering, Technical University of Darmstadt, 64287 Darmstadt, Germany

* Correspondence: philipp.beckerle@tu-dortmund.de

Received: 30 September 2020; Accepted: 24 November 2020; Published: 27 November 2020



Abstract: People with physical disabilities in their upper extremities face serious issues in using classical input devices due to lacking movement possibilities and precision. This article suggests an alternative input concept and presents corresponding input devices. The proposed interface combines an inertial measurement unit and force sensing resistors, which can replace mouse and keyboard. Head motions are mapped to mouse pointer positions, while mouse button actions are triggered by contracting mastication muscles. The contact pressures of each fingertip are acquired to replace the conventional keyboard. To allow for complex text entry, the sensory concept is complemented by an ambiguous keyboard layout with ten keys. The related word prediction function provides disambiguation at word level. Haptic feedback is provided to users corresponding to their virtual keystrokes for enhanced closed-loop interactions. This alternative input system enables text input as well as the emulation of a two-button mouse.

Keywords: assistive technology; alternative input device; keyboard replacement; hands-free mouse; vibrotactile feedback

1. Introduction

People with muscular dystrophy (MD) and some other muscle and nervous system disorders lose gross motor control while retaining fine motor control in the early stages. Typical neurological symptoms of MD are muscle weakness and ataxia, as well as loss of balance and coordination [1]. Friedreich's Ataxia (FA), which is the most common hereditary form of ataxia, affects roughly about 1 in 50,000 individuals of European descent [2]. FA is a slowly progressing disease with symptoms that first appear around puberty. Another related disease is spinal muscular atrophy (SMA), a hereditary neuromuscular disorder that leads to weakness in the body, arms and legs. It affects both boys and girls between six months and three years of age and progresses rapidly [3].

As with the rest of the population, people with these and related diseases are increasingly using computers. However, because the limbs gradually atrophy, the arms, wrists, and fingers become more challenging to control as the disease progresses [1]. As a result, human–computer interface devices

such as keyboard and mouse become difficult or impossible to use. Thus, an alternative interface method for mouse and keyboard operation is required [4].

Many alternative text entry systems have been developed by academic research as well as commercial enterprises, including the scanning ambiguous keyboards (SAK) [5–7]. The concept of ambiguous keyboards is defined by distributing the letters to a small number of keys with several letters per key using a built-in dictionary to disambiguate keyboard input. The strategy of SAK is to combine a visual keyboard with specific muscle contractions by a single key, button, or sensor. The keyboard is divided into selectable areas highlighted (= “scanned”) in sequence, allowing muscle contractions to be used as input for text entry. While they require only a single input, they are typically slow [8].

Speech-to-text is another commonly used alternative input method for computer-supported communication. Speech recognition systems can partially solve the text input problem in some languages, but they are not suitable for mouse pointer control. In addition, noisy environments can make speech recognition difficult [9]. To support the accuracy of speech recognition, image processing technology can be used to identify speech from lip images [10]. However, this technology is still mostly inaccessible to people with dysarthria because it cannot detect mild to severe dysarthric speech [11].

Beyond speech-based systems, cameras and computer vision can be used for generating input by monitoring the user’s actions [12,13]. Due to the continuous progression of the disease and the associated loss of upper extremity function, image processing-based eye tracking systems are commonly used [14–16]. It has been found that compared to scanning objects on the screen and selecting the desired object with a switch, eye-tracking generates faster messages with reduced error rates for people with amyotrophic lateral sclerosis (ALS) [17]. To create an input in such systems, it is necessary to either use a mechanical switch or a dwell time. Although eye tracking systems are inconspicuous and provide accurate and precise results, occasional impairments such as oculomotor apraxia, reduced saccade speed and opening/closing of the eyelids make the use of eye tracking devices indispensable for determining the effectiveness [18]. In addition to eye tracking, head tracking solutions [19–22] are used as well. Head movements can be tracked with infrared (IR) [21,22] or video cameras [19,21,23]. Mouse clicks are generated by a physical switch, speech recognition [19,20] or a software interface using a dwell time [21,22]. While keeping the head or eyes still for a moment can be interpreted as a clicking process, the dwell time required delays the input. Although commercial solutions often require high-performance cameras that capture light in the near infrared spectrum to detect pupil or head movements, there are also cost-effective solutions that use only standard web cameras [19,20,24]. However, some low-cost vision systems exhibit object recognition issues under poor lighting conditions [19].

Biopotentials such as electroencephalography (EEG) [25,26], electrooculography (EOG) [27] and electromyography (EMG) [27,28] can be used as input signals as well. However, all these methods require physical electrode contact, which limits their practical application. In addition, fatigue of the eye muscles and dryness of the cornea limit the use of the EOG method [29]. A drawback of the EMG is that the user is hardly able to move his body since the electrodes also react to movement. Such communication systems are required by patients with severe motor impairments, e.g., late-stage ALS, severe cerebral palsy, head trauma and spinal cord injuries [30,31].

Moreover, special control devices attached to the head such as lip control [32] and systems based on inertial measurements [33] exist. These systems realize mouse pointer control and are complemented by secondary input devices such as sip-puff sensors, foot switches, push buttons or piezoelectric sensors generating mouse clicks [34]. These investigated systems, however, are difficult to install and cosmetically undesirable.

Reviewing the aforementioned studies, most inputs are generated either with a dwell time or a mechanical switch. Although mechanical switches offer a simple solution to user input problems, they exhibit a hardware-determined activation threshold constraining the adaptability to individual user needs and capabilities. An adaptable input method is particularly desirable since the muscle

strength of the upper extremities is different for each person and will change in the course of the disease [1,35]. Force sensing resistors (FSRs) are a simple alternative for the determination of muscle activity [36] and therefore constitute a method for realizing an individually adjustable switch. This can be used for mouse and keyboard inputs independent of dwell time and mechanical response threshold. Moreover, FSRs are non-invasive, non-obstructive, easy-to-use, robust and cost-effective.

In a previous work by Abrams et al. [37], the applicability of FSRs for keyboard input was tested with a potential user suffering from ataxia, handling a five-panel touchpad. The user claimed to have difficulty using a conventional keyboard. Especially releasing the force of her fingers in time after a keystroke was challenging. She also explained that she found it difficult to aim and press exactly on one key because of problems with distal fine motor control. Using the five-panel touchpad, she was able to generate valid keyboard signals. It was reported that it is comfortable for both hands and that the pressure exertion is effortless.

The solution proposed in this paper aims at enabling people with MD and related diseases to operate a computer with less effort. For this purpose, we developed a human–computer interface which considers reduced limb mobility by only using head movements, masticatory muscle contraction and finger pressure as input. This paper aims at proving the applicability of the proposed multi-sensor interface concept and thereby underlining the potential of our approach.

Section 2 presents the interface concept and system architecture. Sensor placement, electronics, hardware-based signal processing and the actuators for haptic feedback are described in Section 3. Section 4 outlines digital signal processing, mouse and keyboard input implementations and the graphical user interface (GUI). A functional evaluation is provided in Section 5 and a detailed discussion of technical aspects is given in Section 6. Section 7 provides a conclusion deducing relevant aspects of future work.

2. Interface Concept

A human–computer interface tackling the constraints caused by neuromuscular diseases and their development is required to allow for mouse operation and text entry with limited limb movements. The proposed concept relies on head movement and masticatory muscle contraction to realize mouse pointer control as well as an ambiguous keyboard concept which minimizes the necessary finger movements (Figure 1). The mouse alternative is designed to be head-mounted since we target a low-budget and robust solution which excludes costly hardware.

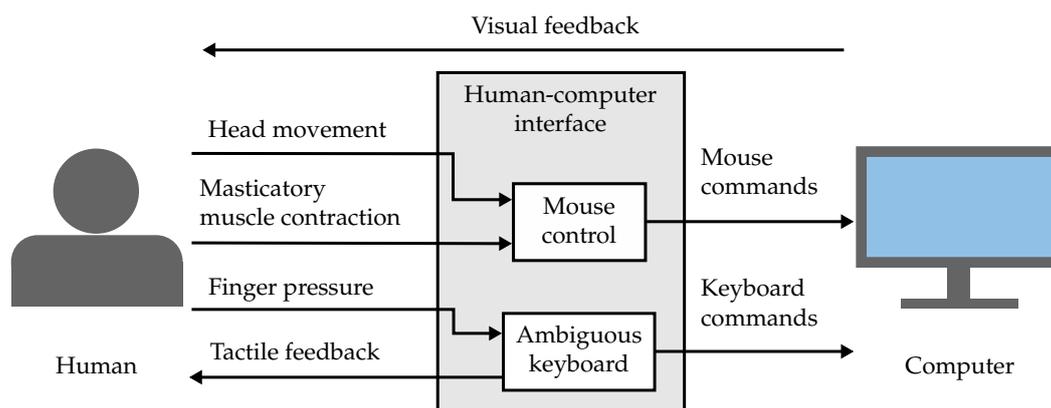


Figure 1. Overall concept of the human–computer interface. Head movements and masticatory muscle contractions are combined to generate mouse commands. Finger pressure functions as input of an ambiguous keyboard which is presented on the screen. Tactile feedback is given when a keystroke is recognized by the ambiguous keyboard.

2.1. Keyboard Concept

A first prototype of the keyboard alternative was created in the preliminary work of Abrams et al. [37]. It consists of five FSRs measuring minimal pressure loads from each finger of

one hand. In this work, we extend the first prototype to a 10-panel touchpad allowing the usage of both hands. However, the keyboard alternative is designed to be modular, so it can be customized to use fewer than ten fingers dependent on the user's individual capabilities.

To enable text entry with only ten keys, the human–computer interface implements an ambiguous keyboard based on SAK by MacKenzie et al. [5]. Our ambiguous keyboard extends this concept through the multiple input possibilities from up to ten fingers and employs the word matching method by Molina et al. [38]. The keyboard layout, which is shown on the screen, consists of eight letter keys and two function keys (Figure 2). Similar to established ambiguous keyboards, e.g., T9 typing of old cell phones, the letters are mapped alphabetically to the keys.

A word is entered by consecutively pressing the keys with the according letters. For example, the word “cat” would require pressing the keys 2, 2 and 8 in this order. The input is compared with a built-in dictionary and a word suggestion list is generated. The user can pick the currently selected word in the list by using the Space key whereupon the word is inserted into the text field. Error correction is achieved by pressing and holding the Next key.

Perspectively, the Next key will allow for browsing the suggestion list by briefly pressing it to select the next word. If the desired word is not listed, we envision the word to be entered via an on-screen keyboard in combination with the developed hands-free mouse.

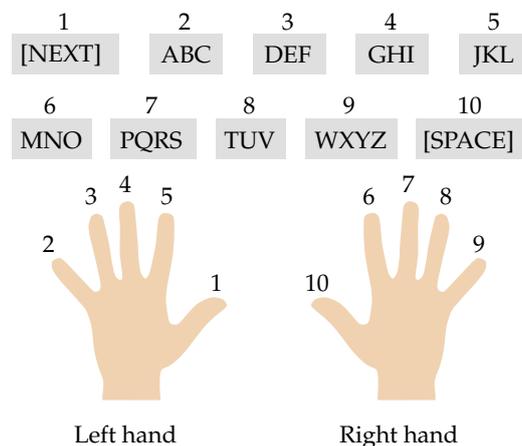


Figure 2. Ambiguous keyboard layout based on the SAK approach by MacKenzie et al. [5]. The input of letters is compared with a built-in dictionary and generates a word suggestion list. The Space key is used to choose the currently selected word. Perspectively, the suggestion list can be browsed by using the Next key. If the desired word is not listed, it can be inserted using the mouse alternative and an on-screen keyboard.

The sense of touch can provide extensive and detailed information about the environment and is a very promising modality for user interfaces [39]. Since the keyboard inputs are generated with minimal movements, haptic feedback during keystrokes is reduced compared to a conventional keyboard. The missing haptic information is partially substituted by additional vibrotactile feedback which is provided for a short duration after each detected keystroke. For a more simple technical design, we only use one vibration motor per hand providing feedback for all corresponding fingers.

2.2. Mouse Concept

The head-mounted mouse alternative is designed as a spectacle frame. In this way, it is not obstructive on the face and also easy to put on and off. Furthermore, the side pieces are located right next to the temporal muscle which belongs to the masticatory muscle group. Masticatory muscles were selected for discrete input since they represent a rather strong muscle group that is trained during everyday activities. Moreover, using temporal muscle contraction as input to generate mouse clicks is advantageous for people with a spinal cord related MD and spinal cord injuries in the neck region since

these muscles are not controlled by the spinal cord but by the hypoglossal nerve directly connected to the brain [40].

A sensor mounted on the spectacle frame measures the head orientation which is translated into cursor movements on the screen. Using a head-mounted sensor avoids the need for an external head tracking system. The interface considers only rotations around the pitch and yaw axis (Figure 3). To account for individual differences, the sensitivity can be modified by the user.

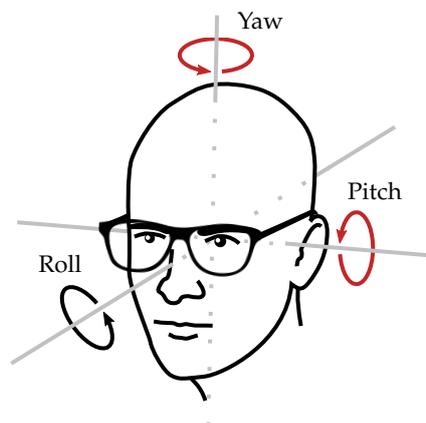


Figure 3. The roll, pitch and yaw angle in human head movement. For two-dimensional movement of the mouse pointer, only pitch and yaw angles are used. The angles are converted into a cursor position considering the screen resolution.

Sensors integrated into the side pieces of the spectacle frame detect masticatory muscle contraction which is used to generate mouse clicks. A short contraction generates a left-click. Maintaining this contraction for 0.6 s results in a right-click. Combining head movements and maintaining contraction activates the drag and drop function.

3. Sensors and Feedback

To implement the designed human–computer interface, appropriate hardware selection is necessary. Since the mouse alternative is intended to be head-mounted, the sensors for obtaining head motion and masticatory muscle contraction are required to be lightweight and non-obstructive. For measuring the finger contact pressures, the previous work of Abrams et al. [37] identified FSRs as a promising technology for keyboard input.

3.1. Sensors

Head movements are acquired via the MPU-6050 inertial measurement unit (IMU) by InvenSense Inc., San Jose, CA, USA mounted on a GY-521 breakout board by SparkFun Electronics, Boulder, CO, USA. The IMU contains a three-axis gyroscope, a three-axis accelerometer and an on-board processor exhibiting 16-bit resolution, a sampling rate of 200 Hz and an Inter-Integrated Circuit (I²C) communication interface at 400 kHz. Through the on-board processor, gyroscope and acceleration data are fused to improve head movement acquisition and, thus, the control of horizontal and vertical cursor movements. As shown in Figure 4a, the IMU chip is mounted on the left earpiece of the spectacle frame.

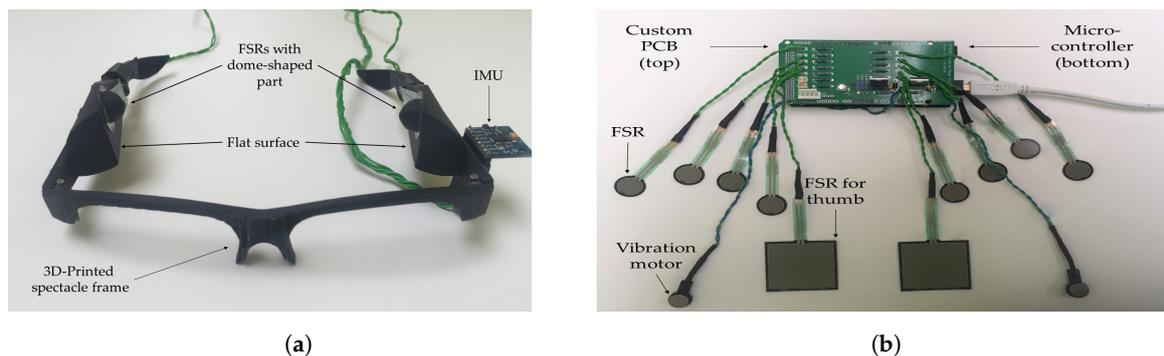


Figure 4. Implementations of the head-mounted mouse alternative (a); and the keyboard interface including the microcontroller board and electronics (b).

As pressure sensors, Interlink 402 FSRs, manufactured by Interlink Electronics Inc., Camarillo, CA, USA, were selected for generating mouse clicks and keyboard inputs. The FSRs rely on semiconducting polymers that exhibit resistance decrease in response to increasing pressure applied on their active surface. The active surface of Interlink 402 FSRs has a diameter of 14.7 mm. Resistance changes are sensed through measuring voltage deviations via a voltage divider connected to the analog input of the microcontroller. One characteristic of an FSR is that its resistance does not change linearly with the force applied. The resistance at first decreases rapidly. This can be advantageous to users with physical constraints using the mouse alternative, because it also makes the circuit sensitive to the first light touches on the FSR.

In addition to eight Interlink 402 FSRs for the fingers, the keyboard replacement contains two Interlink 406 FSRs, which exhibit a larger active area of 31.8 mm × 31.8 mm appearing appropriate for input with the thumbs. The positions of the FSRs responsible for the keyboard were designed to be freely movable to allow for individual adjustments. The adhesive back of the FSRs allows the sensors to be fixed in place.

A good mechanical fixation is essential for the FSR operation since a flat contact surface and a limitation of shear forces is required for accurate measurements. Due to the curvature of human body surfaces, measurement errors due to unevenly distributed load have been observed in body-worn sensors [36]. To ensure direct force application on the FSRs measuring temporal muscle contractions, the sensor is modified with two 3D-printed parts. A flat part fixates the rear and a dome-shaped part distributes the force on the sensor's active surface (Figure 5).

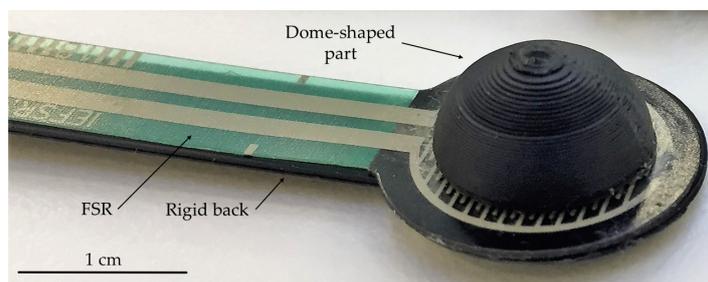


Figure 5. FSR with 3D-printed mechanical modifications for measuring contractions of the temporal muscle. The rigid back fixates the rear and the dome-shaped part distributes the applied force on the sensor's active surface.

3.2. Feedback

VPM2 vibration motors from Solarbotics Ltd, Calgary, AB, Canada, are applied to provide haptic feedback on keystrokes. A VPM2 is a coin-shaped eccentric rotating mass (ERM) vibration motor, and therefore it is suitable for mobile and wearable applications. The vibration amplitudes of the motors are controlled by pulse-width modulation (PWM) outputs of the microcontroller. The motors provide

haptic feedback through 100 ms vibrations on each keystroke. Housings were designed to increase the robustness of the vibrations motors against breakage of their connections. We used acrylonitrile butadiene styrene (ABS) filament to integrate them into the interface system. In addition, an RGB LED was integrated into the system to inform the user about the system's state. A blue LED indicates that all sensors are connected, and the system is ready for operation. When the LED turns green, the user is informed that the mouse alternative has been calibrated, and the system has started. Failures and errors are signaled by the LED lighting up red.

3.3. System Integration

Figure 4a shows the integration of the two FSRs to the left and right inside of a custom, 3D-printed spectacle frame as well as the attachment of the IMU. The flat surface on which the FSRs are mounted allows adjusting the sensor to the position of the temporal muscle. Adjustments of 2.1 cm in horizontal and 2.5 cm in vertical direction are possible.

An Arduino Mega 2560 development board, driven by an ATMEL ATmega 2560 microchip manufactured by Microchip Technology Inc., Chandler, AZ, USA, is used to continuously acquire data from the connected sensors, smoothen IMU data, convert the output voltage of the FSRs from analog to digital with 10-bit resolution and provide all collected sensor data to the computer via a serial connection through the computer's communication port. It is also used to control the vibration motors for tactile feedback.

The structure and signal flow of the overall system is shown in Figure 6. The microcontroller is connected to the IMU module to acquire the orientation data. Pressure values measured by the FSRs are read by the microcontroller's built-in analog-to-digital converter (ADC). The microcontroller transmits the collected sensory data via a serial USB interface to the computer for further processing. The entire system is powered via an internal voltage regulator of the microcontroller board. To ensure reliable operation of the system, the entire circuit was integrated in a custom-built printed circuit board, which directly connects to the microcontroller being implemented as an Arduino shield.

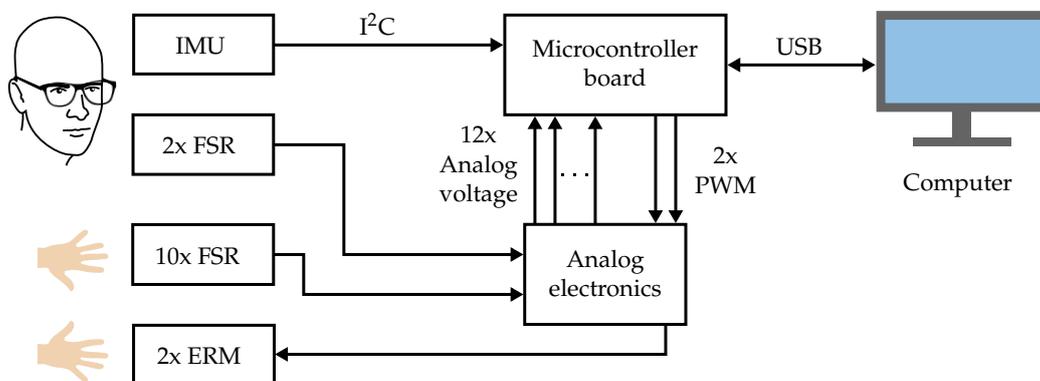


Figure 6. Block diagram of the human–computer interface. The hardware is connected to a computer via USB. A microcontroller board mediates communication between the user and the computer. The keyboard alternative consists of ten FSRs measuring finger contact pressures and two ERMs for vibrotactile feedback. One IMU measuring head orientation and two FSRs detecting masticatory muscle contraction attached to a spectacle frame constitute the mouse replacement. The measured head orientation is sent to the microcontroller via I²C. The resistance values of the FSRs are converted to analog voltages by voltage dividers and read by the microcontroller's built-in ADC. All the sensory data are sent to the computer for further signal processing. The computer sends commands for vibrotactile feedback to the microcontroller which in turn controls the ERMs with PWM outputs.

4. Interface Software

The interface software includes the microcontroller code on the sensor side and a program code on the computer side for operating the mouse pointer and writing with a dedicated user interface. The microcontroller code implements signal acquisitions, filtering and transmission while the software on computer side is responsible for signal processing and interpretation as well as mapping those to mouse and keyboard actions.

4.1. Data Acquisition and Preprocessing

The microcontroller code is responsible for collecting all FSR data, the IMU orientation information and sending them to the computer. After the connection is established, the IMU is configured by adjusting the accelerometer's full scale range to ± 2 g and the gyroscope's full scale range to ± 250 degrees per second. At each program start, a calibration procedure is carried out in which the active offsets of the accelerometer and gyroscope are determined to place the cursor in the center of the screen regardless of the initial position of the head. During the calibration phase, it is necessary to keep the head stable. However, the yaw value of the IMU can exhibit a drift, which is limited by the internal processor of the IMU. This value is considered and compensated by the program running on the computer. Therefore, the IMU must be kept stable until the readings have stabilized after each program start. The fused sensory data processed on the internal IMU processor, i.e., head pitch and yaw angles, are read as degrees. The read-out angles are then smoothed with an exponential moving average filter to reduce noise. In addition, depending on the screen resolution, a movement threshold tolerated was defined to ignore small head movements in order to prevent any trembling of the cursor.

$$S_t = \begin{cases} S_{t-1}, & S_t - S_{t-1} \leq \text{movement threshold} \\ 0.2Y_t + 0.8S_{t-1}, & S_t - S_{t-1} > \text{movement threshold} \end{cases}$$

where Y_t is the value of the IMU angle after fused sensory data in a period t and S_t is the value of the filtered IMU angle at any time period t .

The microcontroller converts the sensor values of all FSRs to digital 10-bit values yielding a resolution of 4.88 mV per unit. After acquiring all current sensory data, the microcontroller sends the current yaw, pitch and FSR data via the serial interface to the computer.

The microcontroller checks for specific byte-string commands from computer program via the serial interface to provide appropriate user feedback through the vibration motors and the LED. Depending on the received strings, the microcontroller commands the motors and the LED reacting to certain events.

4.2. Computer Input and User Feedback Generation

The software running on the computer is implemented in Python to collect and evaluate the data provided by the microcontroller. Furthermore, depending on user actions and sensor states, corresponding inputs are generated for the microcontroller.

Mouse clicks and keystrokes are determined from the FSR signals using upper and lower thresholds. If the pressure value exceeds the upper threshold, a mouse or keyboard input is generated. The program waits for the pressure value to drop below the lower threshold right after to generate a new input to represent discrete states of action. Except for the drag and drop function, all other defined click functions for mouse and keyboard may only be re-triggered when the pressure value has dropped below the lower threshold value. The threshold values for mouse and keyboard inputs were determined experimentally with five able-bodied participants and these values are specified accordingly in the program code. Typically, the two FSR values for the mouse click actions are averaged to compensate for the pressure differences of the FSRs placed on the two temporal muscles.

For mouse pointer control, the measured yaw angle and pitch angle are mapped to a pixel coordinate depending on the screen resolution. In a preliminary test with five able-bodied participants

the ranges required to reach all positions on the screen were $\pm(5-10)^\circ$ for yaw angle and $\pm(4-8)^\circ$ for pitch angle. Any movement beyond these ranges has no effect on the pixel coordinate controlling the mouse pointer. Furthermore, the position control of the mouse pointer is an absolute positioning, i.e., a specific angular orientation corresponds always to the same pointer coordinates and the speed of cursor movement is proportional to the angular speed of head movement.

An ambiguous keyboard GUI was developed to test the typing behavior of the ten inputs with a simple word matching algorithm (Figure 7). The user interface consists of five lines and buttons. In the top line, a target phrase from a database appears in the output field. Below, the sentence entered by the user is displayed. The next line displays the current button numbers that have been typed by the user. The next output field contains the mapping of number to letters. The area for the letter selection is arranged alphabetically on virtual keys with more than one letter per key according to the order defined in the constructor. For Space and Next, two additional virtual keys are added. The last line shows the suggestions made by the program.

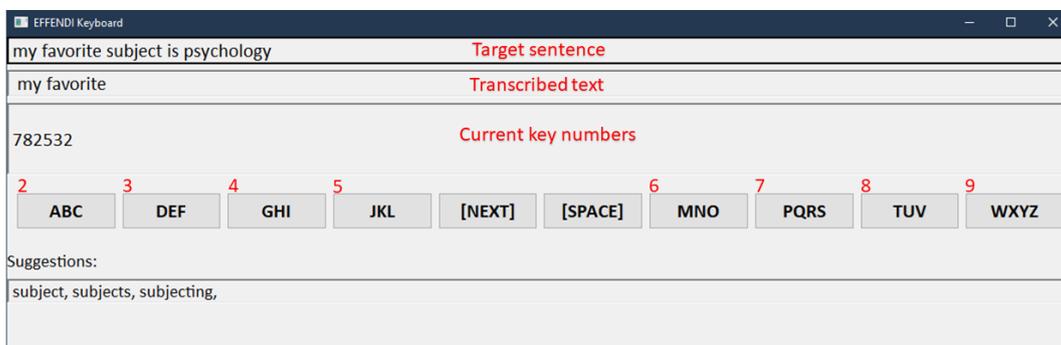


Figure 7. Visual representation of the ambiguous keyboard GUI.

A word matching algorithm is applied to achieve unambiguous entries. To this end, a list of 9022 unique words and frequencies based on a version of the British National Corpus, which was specially compiled for research [5], was used. After key numbers entered by the user are converted to letters, these letters are combined and finally compared with the word list.

Currently, the Next function of the keyboard concept described in Section 2 is not realized. Thus, only the first word in the suggestion list can be selected with the Space key and the input method for non-dictionary words is not complete. However, the error correction functionality with the Next key is executable.

5. Technical Evaluation

The functionality of the proposed alternative input system is technically evaluated focusing on the measurement of head motion by the IMU for mouse pointer movement, the FSRs as input devices and text input with the ambiguous keyboard.

5.1. Mouse Evaluation

The IMU behavior under human influence was tested. To distinguish between effects due to human influences and effects caused by the system itself, we used a motorized head model to record an application-oriented movement, i.e., moving the mouse pointer to different positions on the screen, without human variability (Figure 8a).

After a stationary start-up phase, the precision rotation stage (M-062.2S, Physik Instrumente (PI) GmbH & Co. KG, Karlsruhe, Germany) moved while the sensor output was monitored. The movement corresponded to moving the mouse pointer from the center of the screen to the left edge, followed by moving it to the right edge, and then back to the center. The mouse pointer positions at the left edge and at the right edge corresponded to 10° and -10° , respectively. The same test was performed with an able-bodied human who tried to replicate the movement sequence. The participant held his

head as stable as possible during the start-up phase. The resulting angle measurements were very similar (Figure 9). The deviations from around 18 s result from the non-ideal human timing and mouse pointer positioning. In the stationary start-up phase, a drifting yaw value is observable in the human case, which is not present in the motorized case. This drift is likely due to slight involuntary head movements during internal IMU calibration, which is executed after power-on and expects the IMU to remain stationary. Depending on the intensity and range of the movement, we observed drifts up to 3°.

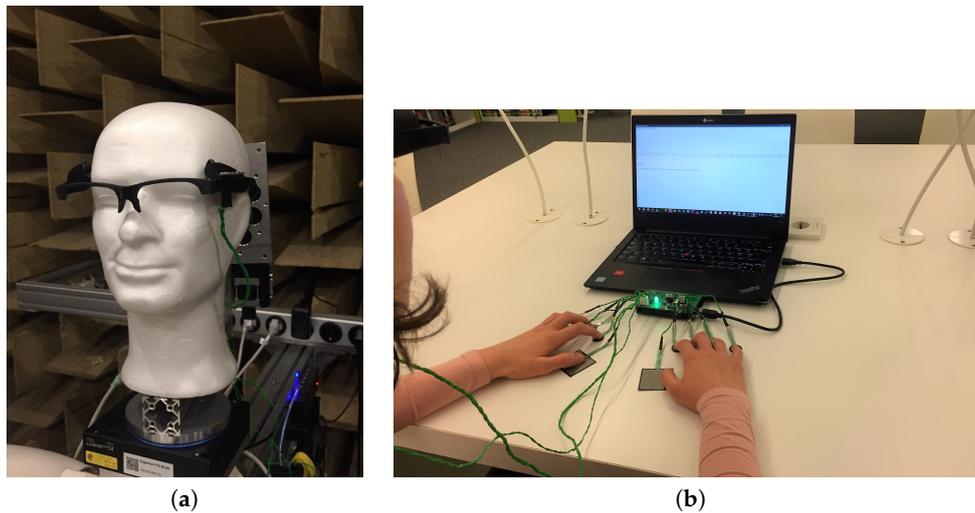


Figure 8. Test setups for assessment of the human–computer interface. To test the human influence on the angle measurement, a reference head movement was performed by a motorized rotation stage. Afterwards, an able-bodied human participant tried to replicate this movement (a). Controlling the mouse pointer and entering text with the interface was tested in a pilot experiment with five able-bodied subjects (b).

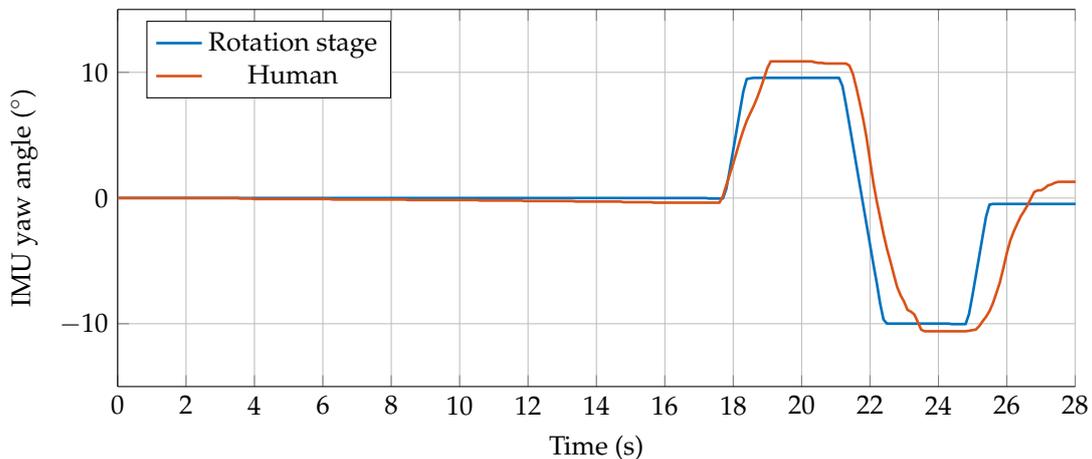


Figure 9. Test of human influence on the angle measurement. A motorized rotation stage simulated a head movement around the yaw axis. The human participant tried to replicate this movement. The measurement involving the human shows a drift at the beginning, which is not present in the motorized case. Despite slight deviations, which can be explained by the human doing an unconstrained movement, the measured angles are similar.

The results of the moving average filter applied to the IMU output data compared to signals without filtering are shown in Figure 10. To simulate the involuntary head movements or body tremors, an able-bodied human participant caused his head to tremble with intentional muscle contractions with both major and minor motions. It can be seen that the filter averages the output value as the head

moves impulsively. Furthermore, the implemented deadband suppresses angle variations below the specified threshold. Through applying the filter, sensory data and, thus, mouse pointer movement are smoothed.

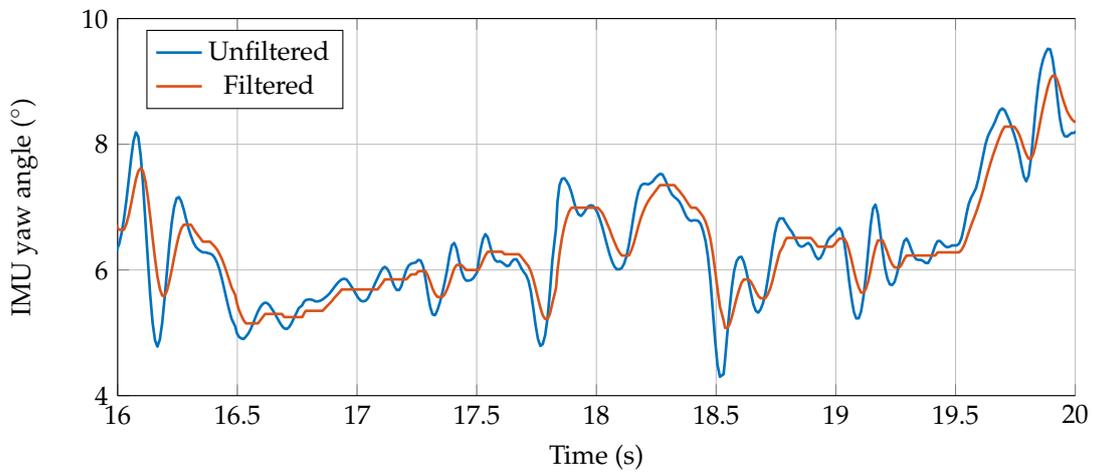


Figure 10. The moving average filter smoothens angle measurement (here yaw), but also adds a slight delay. The additional deadband suppresses angle variations below the specified threshold, which results in short segments of constant yaw angle.

Testing the clicking functionalities when using the mouse alternative shows that targeted muscle contractions can be determined with the FSR (Figure 11). Furthermore, good mechanical coupling and muscle force transmission can be ensured in the temporal musculature, which allows users to generate input signals even with very low muscular activity.

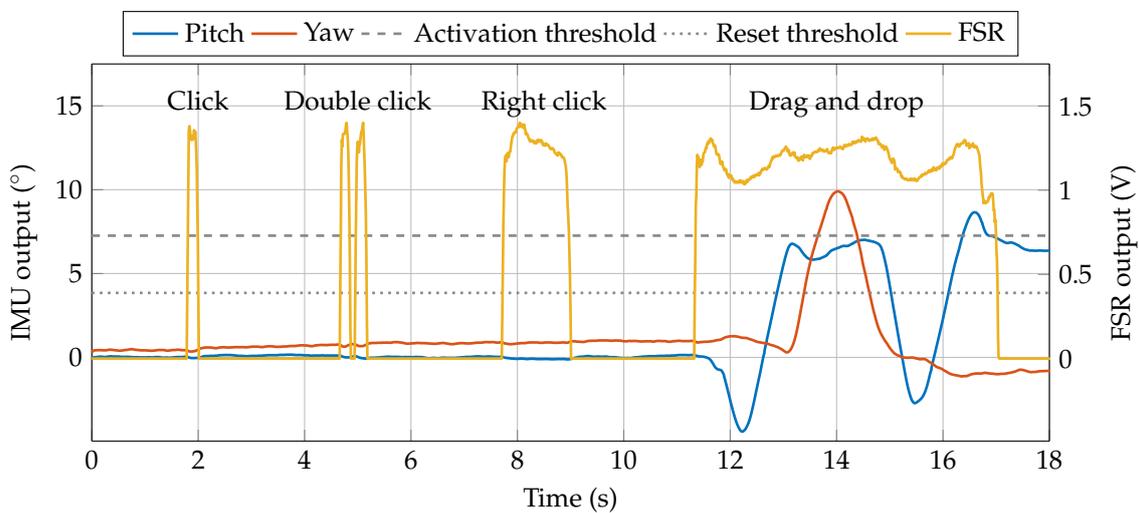


Figure 11. The implemented mouse functionalities. An input is detected when the FSR value exceeds the activation threshold. To generate a further input, the value has to fall below the reset threshold first. A short activation generates a left click, whereas maintaining the activation for at least 0.6 s results in a right click. A head movement during activation starts the drag and drop function.

5.2. Keyboard Evaluation

The 10-panel touchpad keyboard addresses the user’s difficulty of positioning, pressing and releasing fingers. As the fingers remain stationary on the sensors (Figure 8b), these problems due to the loss of fine motor control are solved. The previous work published by Abrams et al. [37] also confirms the aforementioned result.

Figure 12 illustrates the sensor drift on FSR when used with the alternative keyboard based on data from one exemplary user. The index and middle finger of the right hand were placed on the two FSR sensors. No deliberate force, additional to the fingers resting on the FSRs was applied to the sensors during the first 59 s. Beyond 59 s, keystrokes were simulated. The pressure value of the opposite finger was drastically reduced when pressing one finger. This interaction constrains the drift to a certain extent during the sensors’ continuous operation.

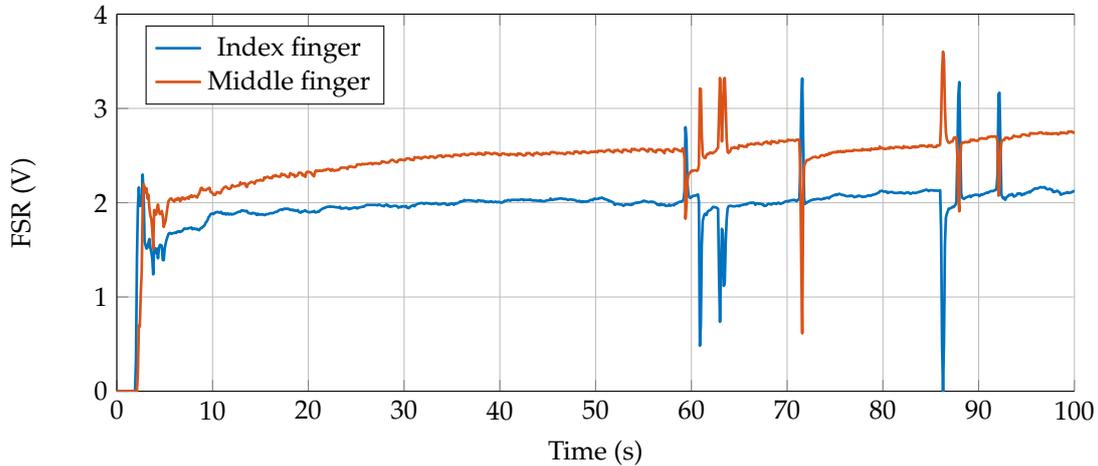


Figure 12. Index and middle finger are put on the keyboard alternative around 2 s and stay stationary up to 59 s. Subsequently, multiple keystrokes are executed. The stationary phase in the beginning shows a drift of the FSR values that adds to the approximately constant load of the fingers. A keystroke results in a positive spike of the corresponding FSR value and a drop of the other signal due to slightly releasing pressure with the resting finger.

A generic set of 23 phrases out of 500 was used as the target phrase [41], to test the keyboard interface and related software. Some examples are given in Table 1. The English phrases range 22–40 characters (mean = 28.3) in length and were presented to a test user. The user typed the sentences with the ambiguous keyboard interface and the number of keystrokes per character (KSPC), which exclusively depends on the interface, was counted. KSPC is the number of keystrokes required, on average, to generate a character of text for a given text entry technique in a given language. For conventional text entry using a QWERTY keyboard, KSPC = 1 since each keystroke generates a character [42]. Because the Next function is not included in the current version of the system, the calculations were performed for two cases. Practical results give the numbers of the KSPC in the current state, while theoretical results are given for the case that the Next function would have been implemented.

Table 1. Five examples of target phrases for assessing the ambiguous keyboard [41].

my bank account is overdrawn
we are having spaghetti
my favorite subject is psychology
great disturbance in the force
a steep learning curve in riding a unicycle

The practical KSPC analysis refers only to the Space key’s functionality and yields a mean of 0.867. Theoretical results simulating the Next button show a mean of 0.843 and outline that the Next function does not cause a drastic difference in KSPC. Out of 23 sentences, only 20 could be used for the experiment. The three sentences could not be written entirely because the target word in the suggestion list is always displayed in the second or third position. This is the case if the target word

has less occurrence in the dictionary used and is therefore always displayed at the second or third position in the suggestion list. In this case, the Next button must be used.

6. Discussion

The combination of scanning ambiguous keyboard layout by MacKenzie et al. [5] and the extended design of Abrams et al. [37] with a 10-panel touchpad is functional. The fingers remain on the sensor device during use, thus avoiding aiming and releasing the force. This appears to be a technically feasible alternative to a regular keyboard for the targeted user group. In comparison to the system developed by MacKenzie et al. [5], our system has a total of 10 possible inputs and replaces mechanical switches with FSRs. Thus, the inputs can be generated with increased sensitivity, and the thresholds can be adjusted user-dependent without a scanning cycle, resulting in more inputs per time. Furthermore, our system allows producing an input independent of the dwell time enabling more user-specific input functions and therefore a more intuitive keyboard operation. In addition, mouse pointer control was realized by cost-effective IMU and FSRs. Since IMU-based alternative mouse interfaces require discrete mechanisms to generate clicks, this study underlines that FSRs are a viable alternative to click generation approaches such as EOG, mechanical switches, dwell time and voice control.

FSRs are low-cost, small, unobtrusive, robust and suited for wearable application, but are constrained by sensor drift. The sensor drift of the FSR is a drawback for our keyboard application. If an FSR is exposed to a constant load over an extended period of time, a mechanical creep behavior can be observed [43] which leads to a drift of the sensor output. In the study by Hollinger et al., the FSR sensor's drift was observed for 10 min with a stationary weight. It was found that the Interlink 402 sensor drifted by 4.41% over 10 min [44]. While this can be circumvented by application-specific activation thresholds, a final assessment of the impact on usage by the target group of individuals with limited upper limb mobility should be subject of future work. More intelligent algorithms, e.g., using adaptive thresholds or implementing personalization through learning, will address this issue and further tackle the non-linearity of the sensor output. Actually, the nonlinear behavior appears beneficial for low power consumption in load-free states and causes more sensitive temporal muscle activation detection. Besides the FSR drift, the IMU also exhibits a drift in the yaw angle measurement. This drift can be traced back to head movements in the initial stage of IMU calibration. A start-up phase of the head-mounted mouse alternative in a stationary state does not cause sensor drift, which may motivate an improved calibration process in future work.

Another hardware-related issue can be addressed to the 3D-printed preliminary hardware of the mouse alternative, as it seems to be not sufficiently adjustable since the frame might not precisely fit the heads of different users. Moreover, this might result in the active area of the FSRs deviating from the center of the temporalis muscles resulting in measurement errors. Although the adjustable FSR positions allow better positioning due to the constructed rigid backs, these position adjustments are insufficient. While we could invest sufficient time for adjusting the positioning in our experiments, practical applications will demand for a more robust solution.

Theoretical analysis shows that using the proposed ambiguous keyboard with word prediction yields a lower KSPC value than conventional computer keyboards. An improved KSPC value using the Next button was also confirmed with theoretical results. Although it can be observed that the ambiguous keyboard allows us to write with less effort, there is still no clear evidence about the typing speed, since some functions remain unimplemented and extensive user studies are subject to future works. Enhanced error correction methods combined with the Next key should help to improve the writing speed.

The quantitative results of the system look promising and motivate us to further develop the system. An additional calibration step can eliminate sensor-related restrictions such as IMU drift in the yaw value before placing the glasses on the head, allowing the system to be started without human influence. In addition, a more anthropometric and robust frame can be designed to provide a better

contact area between FSRs and temporalis muscles and a better muscle force transmission. To allow for the final application, i.e., the full-featured replacement of conventional mouse and keyboard, the functionalities of browsing the suggestion list and entering an unknown word have to be implemented. Moreover, the influence of the keyboard layout mapping letters to keys could be examined [38,45]. Most importantly, future user studies, especially with participants from the target group, will allow assessing and improving the system by examining user performance and thus providing further knowledge regarding the specific needs.

7. Conclusions

In this paper, we report on a modular multisensory human-machine interface designed to provide computer access for people with physical disabilities in the upper extremities. The implementation of the mouse interface in combination with inertial measurement unit and force sensing resistors is technically functional. Furthermore, the force sensing resistor-based ambiguous keyboard approach allows the user to type with minimal finger movements requiring less effort. Since all input thresholds can be set according to the user's needs, the system is customizable for each individual user. Further studies with the target group will provide detailed information about users' needs and foster further improvement of the system. Perspectively, we think that the head-mounted mouse and the ambiguous keyboard based on force sensing resistors could likely be an attractive and cost-effective option enabling computer access for people with limited upper limb mobility.

Author Contributions: Conceptualization, D.G. and P.B.; methodology, D.G. and P.B.; software, D.G.; validation, D.G., N.S. and P.B.; formal analysis, N.S., M.K. and P.B.; investigation, D.G., N.S. and P.B.; resources, P.B.; data curation, D.G. and N.S.; writing—original draft preparation, D.G., N.S., M.K. and P.B.; writing—review and editing, D.G., N.S., M.K. and P.B.; visualization, D.G. and N.S.; supervision, P.B.; project administration, P.B.; and funding acquisition, P.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received support from the German Research Foundation (DFG) through grants FE 936/6 and KU 3498/3-1.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pearson, C.M. Muscular Dystrophy. *Am. J. Med.* **1963**, *35*, 632–645. [[CrossRef](#)]
2. Schulz, J.B.; Pandolfo, M. 150 Years of Friedreich Ataxia: From Its Discovery to Therapy. *J. Neurochem.* **2013**, *126* (Suppl. 1), 1–3. [[CrossRef](#)] [[PubMed](#)]
3. Lunn, M.R.; Wang, C.H. Spinal Muscular Atrophy. *Lancet* **2008**, *371*, 2120–2133. [[CrossRef](#)]
4. Krishnaswamy, K.; Ordóñez, P.; Beckerle, P.; Rinderknecht, S.; Felzer, T. OnScreenDualScribe with Point-and-Click Interface: A Viable Computer Interaction Alternative Based on a Virtual Modified Numerical Keypad. In Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility, New York, NY, USA, 23–26 October 2016; pp. 257–262. [[CrossRef](#)]
5. Mackenzie, I.S.; Felzer, T. SAK: Scanning Ambiguous Keyboard for Efficient One-Key Text Entry. *ACM Trans. Comput.-Hum. Interact.* **2010**, *17*, 1–39. [[CrossRef](#)]
6. Felzer, T.; MacKenzie, I.S.; Beckerle, P.; Rinderknecht, S. Qanti: A Software Tool for Quick Ambiguous Non-Standard Text Input. In *Computers Helping People with Special Needs; Lecture Notes in Computer Science*; Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J.M., Mattern, F., Mitchell, J.C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B., et al., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6180, pp. 128–135. [[CrossRef](#)]
7. Miró-Borrás, J.; Bernabeu-Soler, P.; Llinares, R.; Igual, J. Ambiguous Keyboards and Scanning: The Relevance of the Cell Selection Phase. In *Human-Computer Interaction—INTERACT 2009; Lecture Notes in Computer Science*; Gross, T., Gulliksen, J., Kotzé, P., Oestreicher, L., Palanque, P., Prates, R.O., Winckler, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5727, pp. 1–4. [[CrossRef](#)]
8. Ghedira, S.; Pino, P.; Bourhis, G. Conception and Experimentation of a Communication Device with Adaptive Scanning. *ACM Trans. Access. Comput.* **2009**, *1*, 1–23. [[CrossRef](#)]

9. Li, J.; Deng, L.; Gong, Y.; Haeb-Umbach, R. An Overview of Noise-Robust Automatic Speech Recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 745–777. [[CrossRef](#)]
10. Yi, Z.; Quan-jie, L.; Yan-hua, L.; Li, Z. Intelligent Wheelchair Multimodal Human-Machine Interfaces in Lip Contour Extraction Based on PMM. In Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guilin, China, 19–23 December 2009; pp. 2108–2113. [[CrossRef](#)]
11. Young, V.; Mihailidis, A. Difficulties in Automatic Speech Recognition of Dysarthric Speakers and Implications for Speech-Based Applications Used by the Elderly: A Literature Review. *Assist. Technol. Off. J. RESNA* **2010**, *22*, 99–112. [[CrossRef](#)] [[PubMed](#)]
12. Erol, A.; Bebis, G.; Nicolescu, M.; Boyle, R.D.; Twombly, X. Vision-Based Hand Pose Estimation: A Review. *Comput. Vis. Image Underst.* **2007**, *108*, 52–73. [[CrossRef](#)]
13. Poole, A.; Ball, L.J. Eye Tracking in HCI and Usability Research. In *Encyclopedia of Human Computer Interaction*; Ghaoui, C., Ed.; IGI Global: Hershey, PA, USA, 2006; pp. 211–219. [[CrossRef](#)]
14. Panwar, P.; Sarcar, S.; Samanta, D. EyeBoard: A Fast and Accurate Eye Gaze-Based Text Entry System. In Proceedings of the 2012 4th International Conference on Intelligent Human Computer Interaction (IHCI), Kharagpur, India, 27–29 December 2012; pp. 1–8. [[CrossRef](#)]
15. Tobii Dynavox. PCEye Plus, I-Series Product Page. Available online: <https://www.tobiidynavox.com/products/devices/> (accessed on 10 September 2020).
16. Eyetechnical Systems. TM5-Mini by Eyetechnical Digital Systems Product Page. Available online: <https://www.eyetechnical.com/tm5mini.html> (accessed on 10 September 2020).
17. Gibbons, C.; Beneteau, E. Functional Performance Using Eye Control and Single Switch Scanning by People With ALS. *Perspect. Augment. Altern. Commun.* **2010**, *19*, 64–69. [[CrossRef](#)]
18. Zadikoff, C.; Lang, A.E. Apraxia in Movement Disorders. *Brain A J. Neurol.* **2005**, *128*, 1480–1497. [[CrossRef](#)]
19. Karpov, A.; Cadiou, A. Hands-Free Mouse Control System for Handicapped Operators. In Proceedings of the 11-th International Conference SPECOM, St. Petersburg, Russia, 25–29 June 2006; pp. 525–529.
20. Loewenich, F.; Maire, F.D. Motion-Tracking and Speech Recognition for Hands-Free Mouse Pointer Manipulation. In *Speech Recognition*; Mihelic, F., Zibert, J., Eds.; IN-TECH: Rijeka, Croatia 2008; pp. 427–434.
21. NaturalPoint. SmartNav by NaturalPoint Product Page. Available online: <https://www.naturalpoint.com/smartnav/> (accessed on 10 September 2020).
22. Origin Instruments. HeadMouse Nano Product Page. Available online: <https://www.orin.com/access/headmouse/> (accessed on 10 September 2020).
23. Tsai, P.S.; Wu, T.F.; Hu, N.T.; Chen, J.Y. Assistive Computer Input Device for Muscular Atrophy Patients. *Adv. Mech. Eng.* **2016**, *8*, 168781401664910. [[CrossRef](#)]
24. Dalka, P.; Czyzewski, A. Human-Computer Interface Based on Visual Lip Movement and Gesture Recognition. *Int. J. Comput. Sci. Appl.* **2010**, *7*, 124–139.
25. Orhan, U.; Erdogmus, D.; Roark, B.; Purwar, S.; Hild, K.E.; Oken, B.; Nezamfar, H.; Fried-Oken, M. Fusion with Language Models Improves Spelling Accuracy for ERP-Based Brain Computer Interface spellers. In Proceedings of the 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Boston, MA, USA, 30 August–3 September 2011; Volume 2011, pp. 5774–5777. [[CrossRef](#)]
26. Fabiani, G.E.; McFarland, D.J.; Wolpaw, J.R.; Pfurtscheller, G. Conversion of EEG Activity Into Cursor Movement by a Brain-Computer Interface (BCI). *IEEE Trans. Neural Syst. Rehabil. Eng.* **2004**, *12*, 331–338. [[CrossRef](#)] [[PubMed](#)]
27. Dhillon, H.S.; Singla, R.; Rekhi, N.S.; Jha, R. EOG and EMG Based Virtual Keyboard: A Brain-Computer Interface. In Proceedings of the 2009 2nd IEEE International Conference on Computer Science and Information Technology, Beijing, China, 8–11 August 2009; pp. 259–262. [[CrossRef](#)]
28. Chin, C.A.; Barreto, A.; Adjouadi, M. Integration of EMG and EGT Modalities for the Development of an Enhanced Cursor Control System. *Int. J. Artif. Intell. Tools* **2009**, *18*, 399–414. [[CrossRef](#)]
29. Yamagishi, K.; Hori, J.; Miyakawa, M. Development of EOG-Based Communication System Controlled by Eight-Directional Eye Movements. In Proceedings of the 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, New York, NY, USA, 30 August–3 September 2006; Volume 1, pp. 2574–2577. [[CrossRef](#)]
30. Krusienski, D.J.; Sellers, E.W.; McFarland, D.J.; Vaughan, T.M.; Wolpaw, J.R. Toward Enhanced P300 Speller Performance. *J. Neurosci. Methods* **2008**, *167*, 15–21. [[CrossRef](#)]

31. Pfurtscheller, G.; Neuper, C.; Guger, C.; Harkam, W.; Ramoser, H.; Schlögl, A.; Obermaier, B.; Pregenzer, M. Current Trends in Graz Brain-Computer Interface (BCI) Research. *IEEE Trans. Rehabil. Eng.* **2000**, *8*, 216–219. [[CrossRef](#)]
32. Jose, M.A.; de Deus Lopes, R. Human-Computer Interface Controlled by the Lip. *IEEE J. Biomed. Health Inform.* **2015**, *19*, 302–308. [[CrossRef](#)]
33. Bozkurt, F.; Çağdaş Seçkin, A.; Coşkun, A. Integration of IMU Sensor on Low-Cost EEG and Design of Cursor Control System with ANFIS. *Int. J. Eng. Trends Technol.* **2017**, *54*, 162–169. [[CrossRef](#)]
34. Felzer, T.; Rinderknecht, S. Text Entry by Raising the Eyebrow with HaMCoS. In *Proceedings of the 16th International ACM SIGACCESS Conference on Computers & Accessibility—ASSETS '14*; Kurniawan, S., Richards, J., Eds.; ACM Press: New York, NY, USA, 2014; pp. 355–356. [[CrossRef](#)]
35. Beukelman, D.R.; Fager, S.; Ball, L.; Dietz, A. AAC for Adults with Acquired Neurological Conditions: A Review. *Augment. Altern. Commun.* **2007**, *23*, 230–242. [[CrossRef](#)]
36. Esposito, D.; Andreozzi, E.; Fratini, A.; Gargiulo, G.D.; Savino, S.; Niola, V.; Bifulco, P. A Piezoresistive Sensor to Measure Muscle Contraction and Mechanomyography. *Sensors* **2018**, *18*, 2553. [[CrossRef](#)]
37. Abrams, A.M.H.; Weber, C.F.; Beckerle, P. Design and Testing of Sensors for Text Entry and Mouse Control for Individuals with Neuromuscular Diseases. In *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility*; Hwang, F., McGrenere, J., Flatla, D., Eds.; ACM: New York, NY, USA, 2018; pp. 398–400. [[CrossRef](#)]
38. Molina Cantero, A.J.; Rivera Romero, O.; Gómez González, I.M.; Merino Monge, M.; Ropero Rodríguez, J. Comparison Among Ambiguous Virtual Keyboards for People with Severe Motor Disabilities. *IJMER Int. J. Mod. Eng. Res.* **2011**, *1*, 288–305.
39. Pacchierotti, C.; Sinclair, S.; Solazzi, M.; Frisoli, A.; Hayward, V.; Prattichizzo, D. Wearable Haptic Systems for the Fingertip and the Hand: Taxonomy, Review, and Perspectives. *IEEE Trans. Haptics* **2017**, *10*, 580–600. [[CrossRef](#)] [[PubMed](#)]
40. Barrett, K.E.; Barman, S.M.; Boitano, S.; Brooks, H.L. *Ganong's Review of Medical Physiology*, 25th ed.; McGraw-Hill Education: New York, NY, USA, 2016.
41. Mackenzie, I.S.; Soukoreff, R.W. Phrase Sets for Evaluating Text Entry Techniques. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems—CHI '03*; Cockton, G., Korhonen, P., Eds.; ACM Press: New York, NY, USA, 2003; p. 754. [[CrossRef](#)]
42. Mackenzie, I.S. KSPC (Keystrokes per Character) as a Characteristic of Text Entry Techniques. In *Human Computer Interaction with Mobile Devices*; Paternò, F., Ed.; Springer: Berlin/Heidelberg, Germany, 2002; pp. 195–210.
43. Stassi, S.; Cauda, V.; Canavese, G.; Pirri, C.F. Flexible Tactile Sensing Based on Piezoresistive Composites: A Review. *Sensors* **2014**, *14*, 5296–5332. [[CrossRef](#)] [[PubMed](#)]
44. Hollinger, A.; Wanderley, M.M. Evaluation of Commercial Force Sensing Resistors. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Paris, France, 4–8 June 2006; pp. 4–8.
45. Venkatagiri, H. Efficient Keyboard Layouts for Sequential Access in Augmentative and Alternative Communication. *Augment. Altern. Commun.* **1999**, *15*, 126–134. [[CrossRef](#)]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).