A multimodal virtual keyboard using eye-tracking and hand gesture detection

H. Cecotti[1] *Senior Member, IEEE*, Y. K. Meena[2] and G. Prasad[2] *Senior Member, IEEE*

***Abstract*—A large number of people with disabilities rely on assistive technologies to communicate with their families, to use social media, and have a social life. Despite a significant increase of novel assistive technologies, robust, non-invasive, and inexpensive solutions should be proposed and optimized in relation to the physical abilities of the users. A reliable and robust identification of intentional visual commands is an important issue in the development of eye-movements based user interfaces. The detection of a command with an eye-tracking system can be achieved with a dwell time. Yet, a large number of people can use simple hand gestures as a switch to select a command. We propose a new virtual keyboard based on the detection of ten commands. The keyboard includes all the letters of the Latin script (upper and lower case), punctuation marks, digits, and a delete button. To select a command in the keyboard, the user points the desired item with the gaze, and select it with hand gesture. The system has been evaluated across eight healthy subjects with five predefined hand gestures, and a button for the selection. The results support the conclusion that the performance of a subject, in terms of speed and information transfer rate (ITR), depends on the choice of the hand gesture. The best gesture for each subject provides a mean performance of 8.77 ± 2.90 letters per minute, which corresponds to an ITR of 57.04 ± 14.55 bits per minute. The results highlight that the hand gesture assigned for the selection of an item is inter-subject dependent.**

I. Introduction

With novel assistive technologies, subject specific and adaptive solutions can be proposed to better take into account the constraints of a type of disability. Such an approach can substantially develop the independence of severely disabled people. Disabilities such as patients with neuro-locomotor disabilities or amyotrophic lateral sclerosis are a challenge for carer, nurses, and assistive technology [1]. Individuals with severe speech and motor impairment may be unable to speak nor use sign language to communicate, and they require adapted human-computer interfaces to communicate [2], [3].Furthermore, devices have to be customized in relation to the type of impairment and the constraints imposed by the user. These constraints can be avoided with the adaptation of commercial devices, such as keyboard, joystick; or the creation of new technologies such as brain-machine interfaces for locked-in patients [4]. While brain-computer interface (BCI) can be the only means of communication for a small number of people, a large number of severely disabled people are able to control their gaze, and their gaze can used as a means of communication (e.g., wheelchair control [5], [6]). Severely disabled people may also be able to do some gestures, and the detection of a gesture can be used as a signal to validate an item pointed by the user with his gaze, e.g., people with quadriplegia. The ability of gaze control is actually least affected by disabilities. For instance, eye movement is not affected by severe disabilities such as spinal cord injuries. Virtual keyboards using eyetracking can therefore serve a substantial number of patients and disabled people.

A fundamental issue in human-computer interface with eyetracking is the measure of intention. It can be difficult to interpret because of the amount of involuntary eye movements that lead to involuntary selections of items (i.e., the Midas touch [7], [8]). If a gaze-based interface is realized in a naive fashion then each fixation on an interface control will lead to its activation although the user has no such intention to activate a command. Two solutions to this problem can be considered. The first one is to consider an explicit motor action from the user as an indicator of user's intention to run a command. In this solution, gaze is only used for the selection but not for the control, e.g., voluntary blinks, facial muscle contraction [9]. The type of motor action that is available depends on the user and can lead to involuntary actions, thus involving false positives, and the Midas touch problem is only reduced. Furthermore, it is possible to point at an item with the eyetracker, and to select the item with another input device, such as a switch or gesture detection. Another issue is the unnatural way for the selection of an item as the gaze is typically used to only point at an item, i.e., an action is not directly executed. The direct selection of the desired item is achieved through another modality, such as a motor action or through speech recognition. Depending on the type of disability or constrained it is not necessary to be limited to "facial" commands, which requires sensors to be placed on the face of the user. The second solution to the Midas touch is to measure the total time user's gaze rests within an interface control (the surface of a button) by using a dwell time. If the dwell time exceeds a threshold value then the associated command is enabled. This approach is usually slower and not convenient. In addition, the duration of attention on a particular item (dwell time) has to be determined carefully for a user [10]. A new field of applications has recently emerged with relatively inexpensive remote camera-based eyetracker solutions [11], [12]. With this type of non-invasive system, the eyetracker is located between the user and the computer screen. These inexpensive eyetrackers open new possibilities for affordable assistive

[1] Department of Computer Science, College of Science and Mathematics, Fresno State University, Fresno, Ca, USA.hcecotti@csufresno.edu

[2] Intelligent System Research Centre, Ulster University, Magee Campus, Derry $\sim$ Londonderry, N. Ireland, UK.

978-1-5386-3646-6/18/$31.00 ©2018 IEEE

technology devices such as virtual keyboards. Moreover, they can be used in combination to BCI, where the detection of a brain response enables the selection of an item [13]. Finally, BCI are typically considered more difficult and less reliable approaches than eyetracking due to the low signal to noise ratio in the EEG signal.

For efficient assistive technology devices that can be used daily, a key challenge for the implementation of a robust portable and affordable virtual keyboard based on gaze detection and motion detection is to take into consideration the limitations of the eyetracker in terms of accuracy, the gesture control armband, and human-computer interaction design. For instance, the layout of a regular keyboard may not be used with an eyetracker due to the small distances between the commands: the proximity of the commands in the GUI increases the confusion of the interpretation of the gaze coordinates. For this reason, we propose a virtual keyboard with only ten commands corresponding to ten main nodes in a menu to write 74 different characters (letters, digits, and symbols). This layout is a significant advance compared to classic systems that only focus on letters [14]. The system includes a command for the correction of errors during typing. The selection of an item requires two consecutive actions from the user to enable a command. First, the user has to point to the item that must be selected. A pointer on the screen can be moved to the chosen location, and a visual feedback is provided on the chosen location, if it is a button. Second, the user has to validate the location of the pointer in order to select the corresponding item to enable a command through gesture detection. The accuracy of the eyetracker may limit the number of commands that can be accessible at any moment as the calibration data should be updated when the user changes his head and body position over time.

II. System overview

There are two main components of the graphical user interface (GUI) of the virtual keyboard: the first part is the center of the screen, where the user's input text is displayed, and second part corresponds to the edge of the screen, which displays all the different command buttons. The virtual keyboard, which has ten commands ($C1$ to $C10$) (see Fig. 1), is designed to operate on a tree selection method. The tree has two levels, and allows the user to select any letter with only two commands. In the first level of the tree structure, nine commands (all except $C6$, which is used to delete a character) are dedicated to the selection of the letters, digits, and punctuation marks: 'ABCDabcd', 'EFGHefgh', 'IJKLijkl', 'MNOPmnop', '0123456789', 'QRTSTqrst', 'UVWXuvwx', 'YZ!?yz.,' and '+-/%#_'. Selection of any one of these nine commands opens the second level of the tree. Upon selecting one of the first level's nine commands, the underlying eight characters appear as the commands. The remaining two commands ($C5$ and $C6$) in the second level of the tree are dedicated for 'Undo', allowing the user to cancel the previous action. For instance, selection of the first command 'ABCDabcd' changes the layout, the commands $C1$, $C2$, $C3$, and $C4$ become 'A', 'B', 'C', 'D'; $C7$, $C8$, $C9$, and $C10$ become 'a', 'b', 'c', 'd'. For each block of characters, the four upper case characters are displayed on the upper side of the screen while the lower case characters are displayed on the lower side. Owing to the fact that ten commands are present, it is possible to display all ten numeric digits simultaneously on the screen. However, owing to only ten commands, for deleting an incorrect digit, the user has to return to the first level of the tree, unlike with characters where the Undo command of the second level may be used.

While using a virtual keyboard based on gaze detection, a user may forget or not pay attention to what is already written in the center of the screen as the user has to continuously look at the items to select them if speed is the main measure of performance. It must be noted that while using a regular keyboard, an experienced user does not look at the keyboard but rather focuses on the screen. This may apply to a virtual keyboard based on eyetracking as well, but in an opposite way: an experienced user may only pay attention to the commands that can be selected, and not on the output message box that is displayed in the middle of the screen. To improve the impact of the feedback and the user experience, the last five characters that were spelled-out are displayed under each command. This feature is useful during copy spelling as the user can see what is written in the output without gazing towards the middle of the screen. An auditory stimulus (a beep sound) is played to signify to the user that an item has been selected. Furthermore, a visual feedback for the selected item is given to the user by changing the color of the buttons from a light green to a bright green color to green as the dwell time increases. Finally, an additional feedback was provided to the user to display the current estimation of the gaze location to help the subjects to adapt their head and body position in relation to the error between the expected gaze location and the current position of the detected gaze.

III. Experimental protocol

Eight healthy adult participants (age=28.4 ± 4.9, 2 females) took part of the experiments. The experimental protocol was as per the Helsinki Declaration of 2000, and it was further reviewed by the Faculty Ethics Filter Committee of Ulster University. The eyetracker was an Eyetribe [11]. The recorded gaze data were acquired at 30 Hz, and contained the coordinates and the pupil size for both eyes. The system was calibrated prior to each experiment, where user had to look at a series of dots on the screen (about 20 seconds). Gesture recognition was obtained with the Myo armband by Thalmic Labs for recording sEMG. This non-invasive device includes a 9 degree-of-freedom (DoF) Inertial Measurement Unit (IMU), and 8 dry surface electromyogram sensors. The Myo device can be worn by the user without any particular preparation. The Myo can be slipped directly on the arm to read sEMG signals with no preparation needed for the subject (no shaving of hair or skin-cleaning). The Myo armband provides a sEMG sampling frequency of 200 Hz per channel. Electrode placement was set empirically in relation the size of the subject's forearm because the Myo

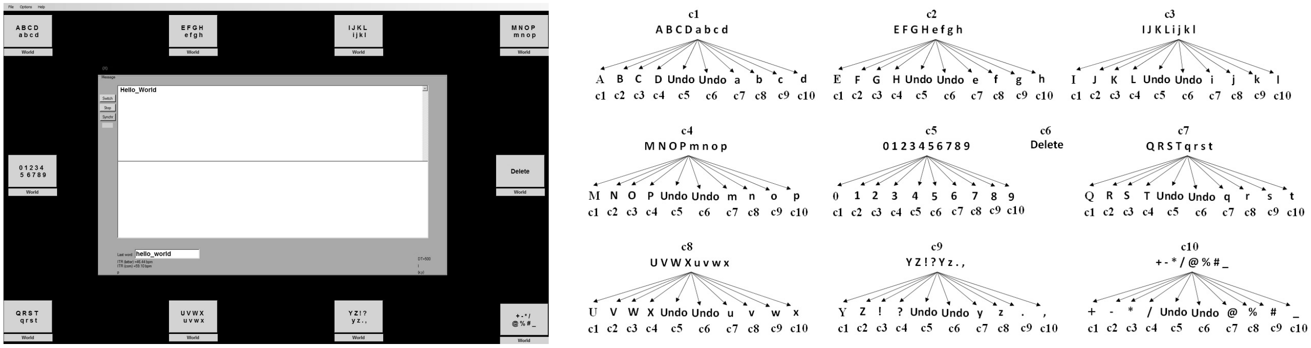

Fig. 1. GUI of the application (**left**), and the tree structure showing the sequence of commands for item selection (**right**).

armband's minimum circumference size is about 20 cm. An second calibration was performed for each subject with the Myo (about 1 min). Participants were comfortably seated in a chair at about 80 cm from the computer screen (Asus VG248, 24 inch, resolution: 1920x1080, 144 Hz refresh rate, 350 cd/m2). The horizontal and vertical visual angles were approximately 36 and 21 degrees, respectively. Each button on the screen had a size of 4×2.5 cm. During the experiment, each subject had to copy a string of 15 characters (30 commands if there are no errors). The eyetracker was used to point towards the commands. Any gaze coordinated in the middle of the screen were discarded. The user could select a command item by paying attention towards it. To help the user, a visual feedback was provided. Then, the command selection was achieved through hand gesture by using predefined functions from the Myo SDK. Six conditions were evaluated: (G1) a button press on a regular keyboard (the space bar); gesture control with the Myo: fist (hand close) (G2), wave left (wrist flexion) (G3), wave right (wrist extension) (G4), finger spread (hand open) (G5), and double tap (G6).

The performance of the virtual keyboard was assessed with the typing speed (i.e., number of letters spelled-out per minute), the information transfer rate (ITR) at the command level (ITR$_{com}$) and at the letter level (ITR$_{symb}$) [15], and both the mean and standard deviation of the time to produce each command. At the command level, we denote by M_{com} the number of possible commands corresponding to the total number of items that can be selected by the eyetracker at any moment ($M_{com} = 10$). At the letter level, i.e., at the application level, we denote by M_{symb} the number of commands ($M_{symb} = 74$) (the 26 characters in upper [A..Z] and lower case [a..z], digits [0..9], space, punctuation marks {'.',',','?','!'}, and symbols {'+','-','*'';','%','#'}. The ITR is based on the total number of actions (direct commands, letters), and the amount of time that is required to perform these commands. By considering an equiprobability between the different possible commands and letters, the ITR is defined by: ITR$_{com} = log_2(M_{com}) \cdot N_{com}/T$, and ITR$_{symb} = log_2(M_{symb}) \cdot N_{symb}/T$, where N_{com} is the total number of produced commands to spell N_{symb} characters. By considering an average execution time of 2 s, the maximum theoretical typing speed, ITR$_{symb}$, and ITR$_{com}$ are 15 letters/min, 93.14 bits/min, and 99.65 bits/min, respectively.

TABLE I
PERFORMANCE CORRESPONDING TO THE OPTIMAL HAND GESTURE FOR EACH SUBJECT.

Subj.	Condition	Speed (letter/min)	ITR_{com} (bits/min)	ITR_{symb} (bits/min)
1.	wave left	10.48	66.72	58.87
2.	wave right	11.22	69.05	64.73
3.	wave right	7.83	57.59	45.46
4.	finger spread	11.36	72.04	63.57
5.	wave right	11.87	69.11	66.88
6.	finger spread	4.86	36.10	21.61
7.	finger spread	4.71	38.95	27.26
8.	fist	7.81	46.73	45.22
mean	-	8.77	57.04	49.20
std	-	2.90	14.55	17.43

IV. Results

The performance across subjects for each condition is presented in Table II. The typing speed for the switch was 9.34 ± 4.62 letters per minute while the other conditions provided an average typing speed of 6.44 ± 3.55, 5.62 ± 2.25, 8.01 ± 3.40, 6.52 ± 2.84, and 6.59 ± 2.15 letters per minute for fist (G2), wave left (G3), wave right (G4), finger spread (G5), and double tap (G6), respectively. Pairwise comparisons indicated that there are no significant differences between conditions across subjects. Yet, this analysis highlights the variability present between the different gestures across subjects. The results confirm some comments from the subjects who did mention that some gestures are more natural or easier to perform than others. The best individual performance for the gesture conditions is given in Table I. The average typing speed across subjects, by selecting the best gesture for each user, is 8.77 ± 2.90 letters per minute, which is translated to an ITR of 57.04 ± 14.55 bits per min at the command level, and 49.20 ± 17.43 at the letter level.

V. Discussion and conclusion

The naturalness and high typing speed are major characteristics of interfaces based on gaze detection, determining their acceptance by the end-users. The user aspect must be taken into account for gaze detection and for the detection of a command through an external device to avoid issues related to the Midas touch. The performance of a new

TABLE II
PERFORMANCE FOR EACH HAND GESTURE: SWITCH (G1), FIST (G2), WAVE LEFT (G3), WAVE RIGHT (G4), FINGER SPREAD (G5), AND DOUBLE TAP (G6).

Condition		Speed	ITR_{com}	ITR_{symb}	Average time
		(letter/min)	(bits/min)	(bits/min)	(ms)
G1	mean	9.34	56.92	52.67	3600
	std	4.62	22.29	25.18	1794
G2	mean	6.44	38.66	36.77	6326
	std	3.55	19.46	19.57	4536
G3	mean	5.62	39.04	32.69	4952
	std	2.25	12.90	12.34	1700
G4	mean	8.01	52.71	46.20	4010
	std	3.40	19.16	19.09	2067
G5	mean	6.52	44.69	36.63	4303
	std	2.84	17.45	16.64	1261
G6	mean	6.59	42.93	37.65	4173
	std	2.15	8.75	11.65	1020

virtual keyboard using both hand gesture detection and gaze control with a portable non-invasive eyetracker has been presented. Six conditions were tested to quantify the change in performance across the different proposed modalities. With the combination of low cost devices, the performance is high enough (about 9 letters/min) to be used efficiently. Moreover, the choice of the graphical user interface layout enhances the distance between items. It has for effect to increase the robustness of the accuracy of the commands detection. Virtual keyboards are difficult to evaluate because the performance is subject dependent, i.e., the motivation of the users, and the duration of the experiment, i.e., the length of the text to write. The goal of the proposed system is to improve the communication means of disabled people, further evaluations will be required to evaluate the performance with patients who may benefit most from the proposed application. When the eyetracker is used to point at a particular item on the screen, the switch that allows the selection can be replaced by any other switch (e.g., eye blinking, a pedal, or the detection of voluntary brain responses [16]). Yet, the choice of this switch should take into account the user experience, and to what extent the switch can be easily accessed. The present multimodal interface does not take into account the head position and orientation. As users can change adapt their posture throughout the experiment, it can degrade the estimation of the gaze coordinate and the overall performance, as participants did not use a chinrest, and some of them would change their head position throughout the experiment. Users naturally orient their head toward the desired item when they are located on the left or the right side of the screen. The addition of the position and orientation of the head could increase the robustness of gaze detection as participants changed their position on the chair despite the need to be in a steady position. The creation of a robust virtual keyboard for a particular script is difficult because the errors of the gaze position estimation must be taken into account. The number of commands can be increased to propose a larger number of items that can be selected with a single command, however, a trade-off must be chosen between the number of steps in the tree menu to access an item and the number of items that can be selected directly through the GUI. Finally, we have shown that the mode of selection should be chosen in relation to each individual as there exists a strong difference across the different modes. While the regular switch button provides the best overall performance, there are situations where a disabled person needs an alternative mode of selection for communication or rehabilitation purposes. Further work will be carried out to improve the interface based on feedback from people with severe disabilities who will benefit from the outcome of this research.

REFERENCES

[1] R. Lupu, R. Bozomitu, F. Ungureanu, and V. Cehan, “Eye tracking based communication system for patient with major neuro-locomotor disabilities,” in *Proc. IEEE 15th ICSTCC*, Oct. 2011, pp. 1–5.

[2] V. Raudonis, R. Simutis, and G. Narvydas, “Discrete eye tracking for medical applications,” in *Proc. 2nd ISABEL*, 2009, pp. 1–6.

[3] H. A. Caltenco, L. N. Andreasen Struijk, and B. Breidegard, “Tonguewise: Tongue-computer interface software for people with tetraplegia,” in *Proc IEEE Eng. Med. Biol. Soc.*, 2010, pp. 4534–7.

[4] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, “Brain-computer interfaces for communication and control,” *Clin Neurophysiol*, vol. 113, pp. 767–791, 2002.

[5] Y. Matsumoto, T. Ino, and T. Ogasawara, “Development of intelligent wheelchair system with face and gazebased interface,” in *Proc. of the 10th Int. Workshop IEEE Robot & Human Interactive Communication*, 2001, pp. 262–267.

[6] D. Purwanto, R. Mardiyanto, and K. Arai, “Electric wheelchair control with gaze direction and eye blinking,” *Artif Life Robotics*, vol. 14, pp. 397–400, 2009.

[7] Aristotle, *Aristotle in 23 Volumes*. Cambridge, MA, USA: Harvard University Press, 1944, vol. 21.

[8] R. J. K. Jacob, “The use of eye movements in human-computer interaction techniques: What you look at is what you get,” *ACM Transactions on Information Systems*, vol. 9, no. 3, pp. 152–169, Apr. 1991.

[9] O. Tiusku, V. Surakka, T. Vanhala, V. Rantanen, and J. Lekkala, “Wireless face interface: Using voluntary gaze direction and facial muscle activations for humancomputer interaction,” *Interacting with Computers*, vol. 24, pp. 1–9, 2012.

[10] P. Majaranta, I. S. MacKenzie, and K.-J. Aula, A. Räihä, “Effects of feedback and dwell time on eye typing speed and accuracy,” *Universal Access in the Information Society*, vol. 5, no. 2, pp. 199–208, 2006.

[11] “The eye tribe, copenhagen, denmark,” https://theeyetribe.com/, 2016, accessed: 2015-06-01.

[12] “Tobii technology, danderyd, sweden,” http://www.tobii.com/, 2016, accessed: 2015-06-01.

[13] X. Yong, M. Fatourechi, R. K. Ward, and G. E. Birch, “The design of a point-and-click system by integrating a self-paced braincomputer interface with an eye-tracker,” *IEEE J. Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 4, pp. 590–602, Dec. 2011.

[14] H. Cecotti, “A multimodal gaze-controlled virtual keyboard,” *IEEE Trans. Human-Machine Syst.*, pp. 1–6, 2016.

[15] H. Cecotti, I. Volosyak, and A. Graeser, “Evaluation of an SSVEP based brain-computer interface on the command and application levels,” in *4th Int. IEEE/EMBS Conf. Neural Eng.*, 2009, pp. 474–477.

[16] Y. Meena, H. Cecotti, K. Wong-Lin, and G. Prasad, “Towards increasing the number of commands in a hybrid brain-computer interface with combination of gaze and motor imagery,” in *37nd Int. IEEE Conf. Eng. Med. and Bio. Soc.*, 2015, pp. 1–4.