Shuffle Speller: User-Adaptive Spelling

Fernando Quivira, Matt Higger, Deniz Erdogmus

1 Problem statement and Background

Many individuals can benefit from augmentative and alternative communication (AAC) while hospitalized, including regular AAC users and patients with temporary communication impairments (e.g. due to intubation). A person with severe speech and physical impairment (SSPI) typically obtains AAC equipment after consulting with a speech-language pathologist (SLP) who will match their needs to an existing solution. This decision takes time and not generalizable across users. Hospitals may not have access to the wide variety of AAC solutions required to serve the entire SSPI population. Moreover, clinics and smaller hospitals may not have AAC-experienced SLPs on call. With limited time for evaluation, the SLP would have difficulty selecting an appropriate device and supporting its continuous use. Overall, there is a clear need for simple, easy-to-use typing interface that will work for most literate patients without extensive trials or training. The system should be accessible with various common input modalities while setting optimal configuration automatically. The ideal solution would be accessible to patients who may have temporary communication impairments (e.g. due to intubation) as well as patients with wide variety of physical abilities.

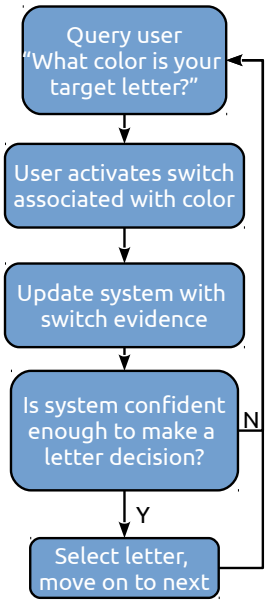


Figure 1: System Flowchart. Unlike traditional decision trees (e.g. switch scanning), Shuffle only selects a letter when it is confident in its decision.

Besides enabling individuals with SSPI to communicate with their loved ones, a robust communication channel would allow them to relay medically relevant messages, health-care decisions, and environmental control preferences, which are vitally important in hospital settings. The development of access methods to reach every person has been humbling though some people are still under-served.

For example, can we empower a user who can sip-and-puff accurately 90% of the time to effectively use a speech-generating device? How about a user who is 80% accurate? And the woman for whom it is only 60% accurate? Rather than prescribe a one size fits all solution, we have constructed a system that learns and adapts to the user's capabilities. When access methods are less dependable, we automatically adjust the query pacing so that users in need are given more opportunities to express themselves. In other words, when a user's physiology precludes them from accurate switch (sip-and-puff, button switch, among others) input, we aggregate the more uncertain inputs until we reach

sufficiently high confidence in a letter to move forward in their typing task. By doing so, we support accurate letter prediction for users with less-than-trustworthy access methods. Moreover, this adaptive paradigm does not slow down a user with crisp, classifiable inputs. Our system shares the same advantage[1] as David MacKay's DASHER [1] which served as an inspiration. DASHER does a marvelous job at enabling users to type quickly and accurately. This system builds on that strength but adds stepwise user input (no continuous, fast-paced interface is needed) and explicit error modelling to adapt to each user's capabilities. For this prototype, we focus on visual brain-computer interface (BCI) input, although the system can be easily extended to other modalities (EMG, switches, etc).

2 Methods and Solutions

2.1 Input Method

Each user has their own unique input strengths and weaknesses. For this reason we design a framework which accepts any arbitrary discrete switch (button switch, sip-and-puff, voice activated, EMG, EOG, EEG etc). Note that Shuffle Speller supports two or more switches from any modality, or combination of modalities, which suits the user's needs and abilities.

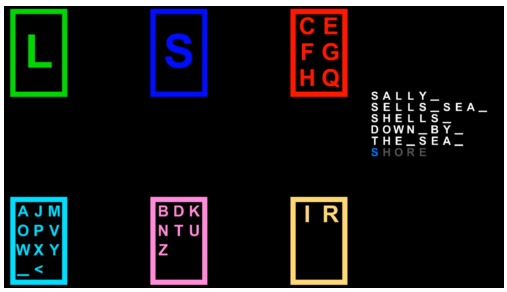


Figure 2: Shuffle Speller Display in 'copy phrase' validation mode. Each colored rectangle is associated with a switch. The user is asked to activate the switch associated with the colored rectangle containing their target letter. In other words, to progress towards picking 'B' activate the pink switch.

Given our funding background we have constructed the Shuffle Speller prototype using steady state visually evoked potentials (SSVEP). SSVEP stimulation consists of blinking lights at different frequencies at various positions within the user's field of vision. The user selects an SSVEP switch by focusing her gaze on one of these flickering lights, which produces a brain signal response associated with the frequency of the target light. This response can be measured by EEG electrodes placed over the occipital lobe. Full details are available in ([2, 3, 4]).

Eye gaze responses may be best suited for some AAC users. An eye gaze 'switch' can be constructed by defining some set of rectangles on the screen; the user selects a rectangle by focusing his gaze inside it.

Shuffle Speller currently supports both SSVEP and eye gaze input methods. To simplify our description we will refer to a user's production of a single response within any modality as a 'switch input'.

2.2 Intent Inference

Our first task in designing a user-centered system is understanding how reliable their access methods are. The system undergoes a one time training in which users are asked to generate each switch input a few times and the classifier does its best to estimate which input the user

[1] Shuffle maximizes the expected reduction in entropy in the user's target letter

was generating. Mistakes will happen, but having an estimate of the reliability of switch access allows us to incorporate evidence appropriately. Imagine a user who has a strong right finger which produces a reliable button press 95% of the time. This same user may have a weaker left finger button press which is only detected accurately 70% of the time. Knowing this allows the system to trust right button inputs and hedge against errors associated with the left button. As much as possible we strive to build a system which learns the user so that the user isn't burdened with learning the system.

The text on the right side of Fig 2 is the 'dashboard' which displays letters previously selected. Shown in Fig 2 is a 'copy phrase' validation mode where we ask the user to type a target phrase ('Sally sells sea shells ...'). In the dashboard letters are colored white to indicate previous selections, grey for future selections and light blue to show the current target letter.

After training is completed, the user may type by a process described in Fig 1. Letters are first animated into colored boxes (Fig 2). The user is then asked to activate the switch associated with the color of their target letter. For example, in the copy phrase task shown the user's target is 'S', so they would activate the switch associated with the top-center dark blue rectangle. The system will then collect and incorporate switch evidence, combining it with a language model. If it has enough confidence in a single letter this letter will be selected and the user will move on to the next in their message. Otherwise the letters are 'shuffled' via an animation to new rectangles and the process is repeated. This process is demonstrated in the videos[2] below.

Shuffle Speller was designed to be robust to single switch errors, as is validated experimentally in Sec 4. To this end, no letters are ever removed from the display because they have been selected against. This is natural given that evidence is probabilistic; that the system thinks a switch was activated does not guarantee that this was the user's intent. Put another way, it would not be wise to discard potential selections when we receive an input which is known to be potentially troublesome like the left button in the example above.

3 Final Approach and Design

To maximize the flexibility of our design, we have followed the human-in-the-loop cyber-physical system design framework [5]. This framework has three sub-systems: (1) human interaction through physiological signal extraction and stimulation, (2) fusion of physiological and non-physiological evidence for intent inference, and (3) communication via the user interface.

Shuffle Speller works as follows: the user has an intended character based on what he wishes to spell (detailed functional diagram is shown in figure 3). The symbols are arranged in the target boxes according following the optimal mapping found using the language model, user input confusion, and current evidence. The user focuses on the target box containing his intended symbol. The data acquisition collects the physiological signal (EEG or eye tracking) during the user selection. The signal is processed and passed to the probabilistic models to estimate the switch dependent probability distribution. The recursive Bayesian

[2] Shuffle Video (demo mouse switch), Shuffle Video (SSVEP switch)

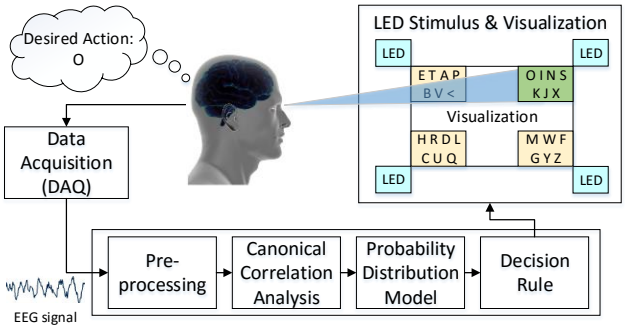


Figure 3: Shuffle Speller processing flow in EEG mode.

update rule is then used to estimate the probability distribution of characters given all the evidence. If a given character's probability exceeds a certain threshold (high confidence), that symbol is chosen and typed [2].

3.1 Data Acquisition

We have built Shuffle Speller to be compatible with the commercially available, research-grade EEG system g.USBAmp, our low-cost custom data acquisition system EEGu2 [6], and the Tobii EyeX. For the EEG data collection, signals were sampled at 256 Hz using either the g.USBamp amplifier and active g.Butterfly electrodes or our custom EEGu2 with passive gold electrodes. We tested the systems with only three electrodes, using a non-slop headband to position them over the occipital lobe at Oz, O1, and O2. For the Tobii EyeX, the pre-filtered eye tracker data streams were acquired using custom software and the EyeX API.

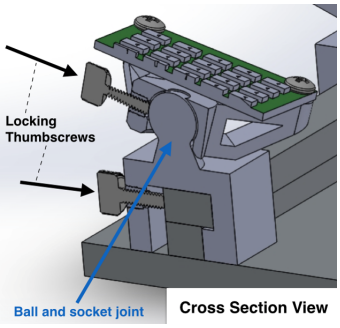


Figure 4: 3D schematic of the adjustable LED arrays that are placed around the monitor.

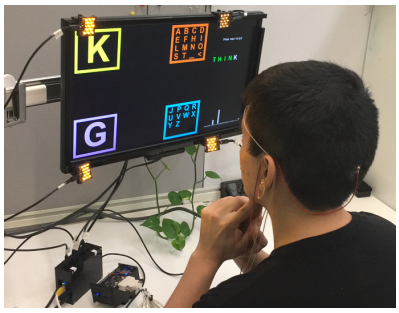
3.2 Visual Stimulus Hardware

The hardware component focuses on presenting accurate visual stimuli to the user according to the patterns designed by the researcher. Timing constraints require the hardware to send precise start-of-stimulation events (or triggers) to the data acquisition component, and to deliver consistent stimuli to the user. The visual stimulus is presented with a set of LED arrays (2 to 6 channels) driven by a custom platform with a Xilinx Spartan3E FPGA [7]. Frequency, pattern, and brightness are configurable using software settings.

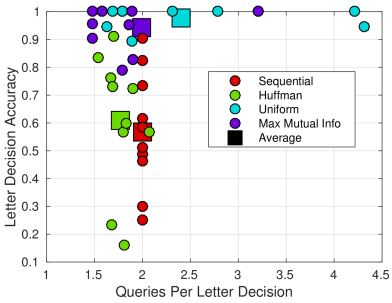
4 Outcome

Does the probabilistic incorporation of user evidence in Shuffle Speller outperform traditional switch scanning? To answer this question, we contrast two modes of operation. We have described recursive updating which queries the user until a sufficiently high confidence is reached in a single letter. Alternatively, decision tree style methods discard letters which are

(a) Calibration session: the system learns which switches are reliable to provide a comfortable, accurate pace per individual user.



(b) Spelling session: The user activates the switch associated with their target letter.



(c) Shuffle Speller's recursive querying scheme (Max Mutual Info, Uniform) offers more reliable communication than decision tree style querying (Sequential, Huffman).

Figure 5: Experiment

selected against. For example, in Fig 2, if the user activates the red switch only the letters in the red rectangle (C, E, F, G, H, Q) remain in future queries; all others are removed. Switch scanning can be a decision tree as once a row is selected a user may not select a character from another row.

To measure the performance difference we had 10 healthy[3] users compare two recursive (Max Mutual Info and Uniform) to two decision tree style (sequential and Huffman) inference schemes. Users were asked to type five words using the Shuffle system, we recorded the speed (queries per letter decision) and accuracy of their typing (Fig 5c). We noticed a dramatic improvement in accuracy for a relatively modest cost in speed. This result validates our claim that Shuffle speller offers a typing scheme which is robust to single switch errors. **Using the Shuffle Speller, all users could type accurately even if their switches were error prone. Individuals who use AAC and have inaccurate or inconsistent control of their access methods need a system which adapts to them to ensure their voices are heard.**

5 Cost

Below is a table outlining the cost for our prototype. Some of these options depend on the acquisition device of choice, most notably the usage of a research grade EEG amplifier. We have shown with our experiments that users can communicate effectively with Shuffle even when using an inexpensive EEG acquisition system. Moreover, to reduce costs to a minimum, an off-the-shelf eye tracker such as the Tobii EyeX will suffice.

[3]Healthy users were selected for inital testing and proof of concept, and SSVEP was selected as the access method because we are funded for exploration of brain-computer interfaces (BCI). Due to the inconsistency and lack of reliability of current BCI technology as an access method, we advocate EOG, EMG or eye gaze switches for Shuffle access for people with SSPI. We are currently developing such switches for the Shuffle framework, designing around the separate, unique challenges of two men who have ALS and a brain stem stroke respectively.

Item	Cost w/ eye tracker ($)	Cost w/ EEGu2 ($)	Cost w/ research EEG ($)
Laptop Computer	1000	1000	1000
BeagleBone Rev C	-	49.99	49.99
Custom PCB	-	841.95	841.95
Custom Enclosure	-	158.79	158.79
Auxiliary Parts	-	285	285
Data acquisition	116	700	12000
Total	**1116**	3335.73	14335.73

6 Significance

Despite creative advances in assistive technologies there are still people who are left with disappointing solutions. In particular, many systems require that an access method be fairly accurate; errors in switch activation are often propagated to errors in letter selection. The novelty of Shuffle is its ability to aggregate the weak evidence of error-prone switches over multiple user queries to produce confident letter decisions. People who use AAC, like everyone, have all sorts of strengths and weaknesses. Those who are hospitalized often present additional challenges, as they may be new or temporary AAC users, or experienced AAC users who do not have access to their typical methods. In either case, these individuals must quickly learn a new means of communication access, often without an ideal level of support from SLPs or other medical staff. We seek to design a system which learns and adapts to the user as much as possible. In doing so the Shuffle Speller framework speeds up to reap the benefits of those with robust, accurate access method as well as slowing down to ensure that those without are not left behind.

We acknowledge that Shuffle, as a project, is at a turning point. The algorithms were born out of mathematical curiosity first and later migrated to an AAC application where they can help people. We have developed partnerships with speech-language pathologists who work with people with SSPI, and have visited individuals with disabilities to learn more about their abilities, needs, and challenges. Through multidisciplinary collaboration and a focus on user-centered design, we hope to continue to improve Shuffle and improve its usability and performance for a wide range of potential users. We have learned, for example, that the Shuffle Speller, despite its potential advantages, may pose obstacles for users with visual or cognitive impairments due to its busy and constantly changing interface. Currently, we offer a robust letter selection scheme for users with sufficient visual and cognitive skills but unreliable switch input abilities.

Future goals include redesigning Shuffle's access method and visualization so that it is more user friendly and cheaper for people to use. A trip to RESNA would help us establish the network and credibility needed to launch this future work.

Additionally, if chosen we look forward to demonstrating Shuffle's use with a Tobii eyeX gaze based switch.

Acknowledgements

We would like to thank our collaborators at the Cognitive Systems Laboratory and the Embedded Systems Laboratory at Northeastern University as well as the team at Oregon Health and Science University's REKNEW lab for keeping our eye on the ball.

References

[1] S. A. Wills and D. J. C. MacKay. Dasher-an efficient writing system for brain-computer interfaces? *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):244–246, June 2006.

[2] M. Higger, F. Quivira, M. Akcakaya, M. Moghadamfalahi, h nezamfar, M. Cetin, and D. Erdogmus. Recursive Bayesian Coding for BCIs. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, PP(99):1–1, 2016.

[3] Hubert Cecotti. A self-paced and calibration-less SSVEP-based brain-computer interface speller. *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, 18(2):127–33, apr 2010.

[4] Z. Lin, C. Zhang, W. Wu, and X. Gao. Frequency recognition based on canonical correlation analysis for ssvep-based bcis. *IEEE Transactions on Biomedical Engineering*, 54(6):1172–1176, June 2007.

[5] Gunar Schirner, Deniz Erdogmus, Kaushik Chowdhury, and Taskin Padir. The Future of Human-in-the-loop Cyber Physical Systems. *IEEE Computer*, 46(1):1–8, 2013.

[6] S. Feng, M. Tang, F. Quivira, T. Dyson, F. Cuckov, and G. Schirner. Eegu2: An embedded device for brain/body signal acquisition and processing. In *27th International Symposium on Rapid System Prototyping*, 2016.

[7] S. Feng, F. Quivira, and G. Schirner. Framework for rapid development of embedded human-in-the-loop cyber-physical systems. In *2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 208–215, Oct 2016.