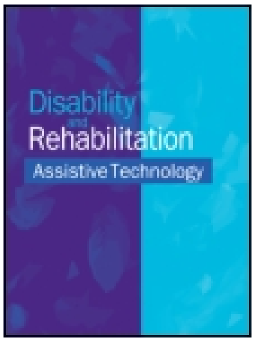

Taylor & Francis
Taylor & Francis Group

Disability and Rehabilitation: Assistive Technology

ISSN: 1748-3107 (Print) 1748-3115 (Online) Journal homepage: http://www.tandfonline.com/loi/iidt20

Validating a model of row–column scanning

Robert Mankowski, Richard C. Simpson & Heidi H. Koester

To cite this article: Robert Mankowski, Richard C. Simpson & Heidi H. Koester (2013) Validating a model of row–column scanning, Disability and Rehabilitation: Assistive Technology, 8:4, 321-329, DOI: 10.3109/17483107.2012.721159

To link to this article: https://doi.org/10.3109/17483107.2012.721159

Disability and Rehabilitation: Assistive Technology, 2013; 8(4): 321–329

ISSN 1748-3107 print/ISSN 1748-3115 online
DOI: 10.3109/17483107.2012.721159

informa
healthcare

RESEARCH PAPER

Validating a model of row–column scanning

Robert Mankowski[1], Richard C. Simpson[1] & Heidi H Koester[2]

[1]*Department of Rehabilitation Science and Technology, University of Pittsburgh, Pittsburgh, PA, USA and*
[2]*Koester Performance Research, Ann Arbor, MI, USA*

***Purpose*: For individuals with severe motor and communicative disabilities, single switch scanning provides a way to access a computer and communicate. A model was developed that utilizes scanning interface settings, error tendencies, error correction strategies, and the matrix configuration to predict a user's communication rate. *Method*: Five individuals who use single switch scanning transcribed sentences using an on-screen keyboard configured with the settings from their communication devices. Data from these trials were used as input to a model that predicted TER for the baseline configuration and at least three other system configurations. Participants transcribed text with each of these new configurations and the predicted TER was compared to the actual TER. *Results*: Results showed that predicted TER was accurate to within 90% on average. The scan rate was also entered into a previously published model which assumes error-free performance. For our model, the average error for each participant was 10.49%, compared to 79.7% for the model assuming error-free performance. *Conclusions*: Our model of row–column scanning was much more accurate than a model that did not consider the likelihood of an error occurring. There is still room for improvement, however, and the results of the study will lead to additional modifications of the model.**

Keywords: Wwitch access, row-column scanning, augmentative communication

Implications for Rehabilitation

- Although row-column scanning is a very slow method of selection, changes in the configuration of the interface can produce noticeable changes in performance.
- When configuring a row-column scanning interface, clinicians should consider the type of errors their client is likely to commit to target interface features to their client's specific needs.
- Some clients who use row-column scanning may not benefit from advanced interface features, even if they are available.

Introduction

Row–column scanning is a technique used by some individuals with physical disabilities for entering text and other data into computers and augmentative and alternative communication (AAC) devices. It is an important method because it can be used with as little as one switch for input. A common implementation of row–column scanning with one switch requires three switch hits to make one selection from a two-dimensional matrix of letters, numbers, symbols, words, or phrases. The first switch hit initiates a scan through the rows of the matrix. Each row of the matrix, beginning with the first, is highlighted in turn until the second switch hit is made to select a row. Each column of the row is then highlighted in turn until the target is highlighted, when the third switch hit is made to select the target. Variations on this theme are abundant and include column-row scanning and continuous row scanning which eliminates the first switch hit needed to initiate row scanning [1].

Single switch scanning is an extremely slow method of text entry. A very fast user may achieve 8 words per minute [2–5], while rates of 1 word per minute and lower are common [6–8]. Although single-switch scanning is very slow, it is often the only alternative for individuals who cannot use other interfaces. Technologies such as eye gaze and speech recognition may be out of reach for individuals with severe spasticity, poor head control, or limited verbal abilities. Direct brain interfaces, while promising, are still early in the development stage [9].

On-screen keyboards and AAC devices offer a wide variety of options for an individual using single switch row–column

Correspondence: Richard C Simpson, Department of Rehabilitation Science and Technology, University of Pittsburgh, Pittsburgh, PA, USA. E-mail: ris20@pitt.edu

(Accepted August 2012)

scanning. Table I summarizes the results of a survey analyzing the adjustable scan settings and user input methods of assistive technology software products that support switch scanning. Proper configuration of the features available within scanning systems can make a major difference in communication rate [1,10]. A case study by Koester [11] demonstrated how modifications to both item layout and scan rate yielded a text entry rate (TER) enhancement of 321% for one individual and the five individuals in Bhattacharya's study [14] showed differences of 20 to 25% when using different configurations.

A real barrier to progress is the lack of an effective and efficient method for tailoring a scanning interface to a particular user. The current standard of care is for clinicians and users to arrive at appropriate settings by trial and error. This makes it difficult to effectively define an optimal configuration. Often, so much time is spent identifying a reliable switch site and a basic scan layout appropriate for the user's needs that very little time is left to properly adjust the remaining options. Proper setting of the software parameters by trial and error would likely require many hours, if not a full day or more. This is simply not a practical solution. The result is that many end up using a system under its default configuration.

An alternative is to use models of user performance to select the most appropriate settings. Several models of scanning have been published [12–18], but they are not complete. Most focus on only one scan pattern (typically row–column scanning), none consider the range of scanning control and error correction methods available in real products, and none integrate the likelihood of errors into their predictions of TER. In fact, up to 63% of selections can involve some type of error [6,13,14]. Without incorporating errors and their consequences, a model can significantly overestimate performance and cannot suggest accurate correction strategies.

We are developing a software tool that makes use of a model of row–column scanning that includes the likelihood of an error occurring and the effect of different configuration options. Our software will help clinicians determine a configuration for a client's scanning system that maximizes TER for that individual. As an early step in this development process, we conducted the study described here, in order to evaluate the accuracy of the model under different configuration options. Successfully validating the model is important because it forms the basis for the software we plan to develop.

Related research

Damper was one of the earliest investigators to model row–column scanning [16]. He developed equations based on the number of scan steps to each matrix item and the frequency with which each item is selected. He assumed error-free performance and did not consider the effects of errors or error correction methods on selection time.

Lesher [17] and Venkatagiri [18] used computer software to simulate text entry, rather than developing equations to calculate text entry rate. The simulations were used to compare different keyboard layouts, scan patterns, and text entry rate enhancement methods (character prediction and word prediction). Both simulations assumed perfect performance by the user.

Abascal studied the effects of errors in a D-dimensional scanning system [12] by calculating the delay introduced by an error and adding that to the time for an error-free selection. Abascal considered errors of *commission* (pressing the switch at the wrong time) but not errors of *omission* (failing to press the switch at all). Within errors of commission, Abascal only considered selecting the wrong group, and not the wrong item, and the only type of error-correction mechanism he considered was a "stop scanning item." Finally, Abascal assumed a single probability *e*, for the occurrence of all types of errors.

Table I. Configuration options found in 16 commercially-available scanning interfaces.

Configuration Option	Supported by %	Explanation
Scan rate	100%	The amount of time an item is available for selection (i.e., highlighted)
Recovery delay	50%	An additional delay added to the first row or column to provide time for the user to recover from a previous switch activation. Different values may be used for rows and columns in some systems.
Loop count	81%	Determines how many times the system will scan through the columns within a row before resuming between rows
Reverse scan	19%	The ability to reverse the directionof scanning through a row
Stop scanning	38%	The ability to stop scanning a row by selecting an item at the beginning or end of each row
Rescan	19%	The ability to re-scan the row by selecting an item at the beginning or end of each row
Automatic/manual scan initiation	88%	Determines whether the user must press a switch to initiate scanning, or if scanning is automatic (and continuous). This setting dictates whether two or three switch presses are required to make a selection.
Switch repeat	50%	Allows user to hold the switch down to register multiple switch activations.
Repeat delay	50%	How long the switch must be held down to register the second activation.
Repeat rate	44%	The length of time between switch activations after the second activation is registered.
Acceptance delay	69%	The length of time a switch must be activated before the activation is registered.
Switch hold escape	6%	The length of time a switch must be held before an exit/escape of the current row or column occurs. Scanning restarts at the top of matrix.
Character prediction	13%	One or more items in the matrix are dynamically updated based on which letters are most likely to be selected next.
Word completion/prediction	100%	One or more items in the matrix are dynamically updated based on what word the user is most likely entering or is likely to enter next.

Table adapted from [6].

Most recently, Bhattacharya [15] empirically evaluated two models of scanning performance (one for one-switch scanning, one for two-switch scanning) with 6 disabled and 2 able-bodied subjects. The model assumed error-free performance, and the investigators removed erroneous selections from the data log. For one-switch scanning, model error when predicting error-free performance ranged from 3% to 10% for 3 subjects with disabilities and ranged from 17% and 19% for able-bodied subjects. For two-switch scanning, model error for error free performance ranged from 1 to 9% for subjects with disabilities and 11–23% for able-bodied subjects. Bhattacharya also developed a model of the occurrence of timing errors (errors of omission) and selection errors (errors of commission) during scanning [13,14] but has not yet combined his performance model with his error model to predict actual TER.

Modeling row–column scanning

Our modeling approach (described in detail elsewhere [19]) is similar to the approach taken by other models [12,14,16–18]. The time required to select a given item is the sum of the time required to scan to the item and the time required to press the switch the required number of times. Our model also includes the delay imposed by each type of error and error correction method, along with the likelihood of each error occurring. For example, if the user fails to select the target row the first time it is highlighted, the system will scan through all the rows in the matrix once and then scan through the rows again until it reaches the target row. The delay (D) due to the timing error in this case is the scan rate (T_s) multiplied by the number of rows (r):

$$D = T_S \cdot r$$

The delay associated with an error is determined by the type of error (omission or commission), where it occurred (i.e., which row or column) and the configuration of the scanning interface (Table I).

Average selection time ($\overline{T_{ij}}$) for the item in row i and column j is the sum of the time for an error free selection (T_{ij}) and the delay (D_x) associated with each type of error (e_x) multiplied by the probability of each error's occurrence ($P(e_x)$):

$$\overline{T_{ij}} = T_{ij} + \sum_x D_x \cdot P(e_x)$$

The average selection time for each item in the matrix is then weighted by that item's frequency of use (F_{ij}) to calculate an average selection time ($\overline{T}$):

$$\overline{T} = \sum_i \sum_j \overline{T}_{ij} \cdot F_{ij}$$

TER in words per minute is a function of the average selection time and the average selections per word ($\overline{W}$):

$$TER = \frac{1}{\overline{T}} \cdot \frac{1}{\overline{W}} \cdot 60$$

The scanning model was implemented within a Java-based program designed and developed for this study.

Methodology

A preliminary study was performed to test the accuracy of this model. Five individuals who use single switch scanning transcribed sentences using an on-screen keyboard that was configured to match the scanning system settings used on their own communication devices. The protocol and consent form were approved by the University of Pittsburgh's Institutional Review Board. All participants provided informed consent.

Participants

Eligible participants were between the ages of 21 and 65. Participants had to be single switch scanners with the cognitive ability to transcribe sentences. Participants also had to possess the visual acuity to see a computer with the screen resolution set to 1024 × 768 pixels.

Five individuals participated in this study. The primary diagnosis for all five participants was cerebral palsy. Each participant used a wheelchair for mobility and an AAC device to communicate. All five accessed their AAC device using single-switch scanning. All participants except P1 used their own switch. For P1, a similar button switch was substituted and functioned properly. Table II shows the switches used by each participant.

On-screen keyboards

The two on-screen keyboards used for the study were the WiViK (www.wivik.com)on-screen keyboard version 3.2 by Bloorview Kids Rehab and the Reach Interface Author (RIA) (www.ahf-net.com) on-screen keyboard version 5.0 by Applied Human Factors. Each of these products has the ability to create custom on-screen keyboards for single switch scanning. RIA was used in all cases, except for two configurations in which a Stop Scanning item was part of the scanning matrix (at the end of each row). The base structure of the custom keyboards contained five rows and six columns. Frequency-based and alphabetic-based layouts were designed for each on-screen keyboard to look as similar as possible between products. Figure 1 is a screen shot of the Reach Interface Author frequency-based keyboard. Figure 2 is a screen shot of the WiViK frequency-based layout with a Stop Scanning item at the end of each row.

Protocol

The flow chart in Figure 3 illustrates the data collection protocol. All data were collected using Compass (www.kpronline.com) computer access assessment software. The Compass Switch Test asks the user to press a switch in response to a prompt, in this case configured to be both auditory and visual prompts. The Switch Test was used to acquire the mean and standard deviation of the participant's switch-press time. The Switch Test

Table II. Participant switches.

Participant	Switch
P1*	Jelly Bean Button Switch
P2	Jelly Bean Button Switch
P3	Tash Micro Light Switch
P4	Electromyographic (EMG) switch
P5	Jelly Bean Button Switch

*Indicates that the switch was supplied by the investigator.

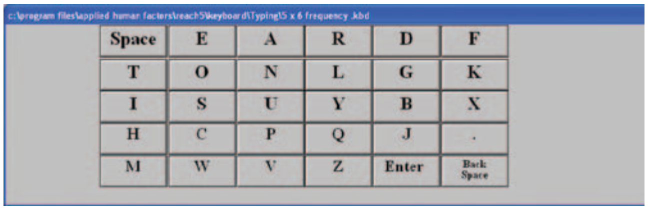

Figure 1. Reach interface author.

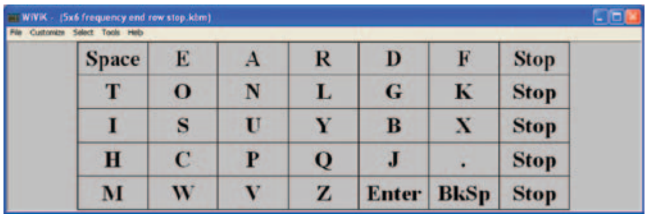

Figure 2. WiViK keyboard with an end of row stop.

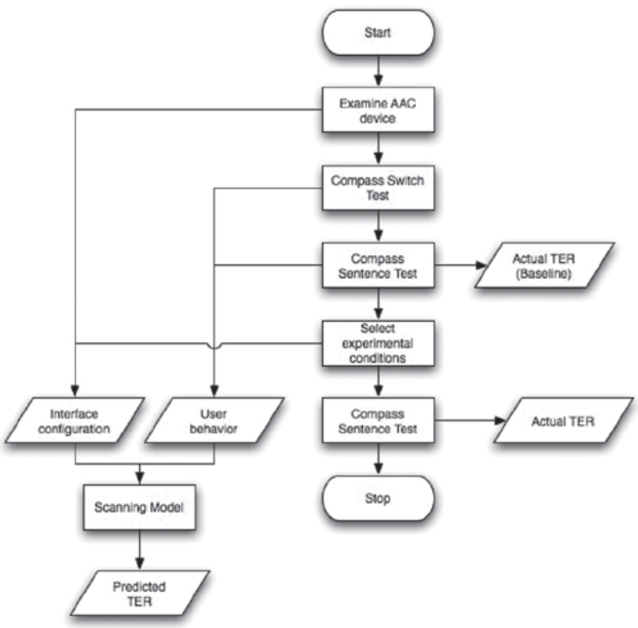

Figure 3. Protocol.

results also included a recommended scan rate based upon the .65 rule [7] (i.e., a reasonable scan rate can be found by dividing average switch press time by .65). For all but one participant, the scan rate recommended by the .65 rule and the participant's own scan rate were the same or within roughly 0.20 seconds, so the participant's current rate was used. For participant P5, an average of the recommended and current scan rate was used.

The Compass Sentence Test was used to record text entry data. Each trial consisted of two sentences entered with a single on-screen keyboard under a single interface configuration. A target sentence was presented in a window with a text entry box below it. Participants were instructed to transcribe the sentences as quickly and accurately as possible. Participants were also asked to correct all errors.

The participant used the on-screen keyboard to select characters via single-switch row–column scanning. Once the sentence had been copied and punctuation selected, the "Enter" key was selected to present the next sentence. Participants were given 6 minutes to enter each sentence. If the sentence was not completed after 6 minutes, the next sentence was presented. The sentences included characters in accordance with their frequency of occurrence in standard English text as per MacKenzie and Soukoreff [20]. The sentences were between 22 and 40 characters long.

For the Baseline condition, the system configuration parameters such as *scan rate*, *recovery delay*, matrix layout, and *loop count* were determined by examination of the participants' current AAC device scan settings. After the participant completed a two-sentence trial of the Sentence Test under the Baseline condition, the results were analyzed to characterize the participant's errors. The tracked errors were a) a selection immediately before the target row, b) a selection immediately after the target row, c) a selection immediately before the target column, d) a selection immediately after the target column, e) no row selected, and f) no column selected within the target row. The probabilities were determined by the following formulas:

Total Selection Events = (correct characters + total number of all types of errors)

Probability Of Error type x = (# of occurrences of Error Type x /Total Selection Events)

TER was calculated by dividing the number of correctly transcribed characters by the total trial time (seconds) and converting the result to characters per minute.

TER = (correct characters/total trial time) * (60)

The total trial time was adjusted in certain circumstances to obtain an accurate TER. A bug with the on-screen keyboard, start delays by the study participant, and various interruptions added time to the trial length. The duration of these events were calculated based on video analysis and subtracted from the total trial time.

At least three interface configurations which were different than the baseline configuration were chosen for each participant. Two participants (P1 and P3) had four configurations as time allowed for additional testing. The configur-ations were determined by examining the Baseline results to identify interface settings most likely to impact the participants' TER. Table III contains the interface configurations used for each participant. Each configuration was implemented by modifying the configuration settings and the matrix layout of the on-screen keyboards. Each participant completed a two-sentence Sentence Test under each configuration chosen for them. Each configuration was also entered into the model to obtain a predicted TER.

Results

The predicted and actual TER for each trial are shown in Table IV. Model error was calculated as:

$$\left|\frac{\text{predicted TER} - \text{actual TER}}{\text{actual TER}}\right| \times 100$$

Table III. System configurations.

Par	Configuration	Matrix	Scan Rate (sec)	Recovery Delay (sec)	Loop Count	Scan Method	Kbd
P1	Baseline	Alpha	1.2	0	1	Normal	RIA
	1	Alpha	**1.25**	0	**5**	**Stop-End**	WiVik
	2	**Freq**	1.2	0	1	Normal	RIA
	3	Alpha	1.2	**0.8**	1	Normal	RIA
	4	**Freq**	1.2	0.8	1	Normal	RIA
P2	Baseline	Alpha	1.4	0	1	Normal	RIA
	1	**Freq**	1.4	0	1	Normal	RIA
	2	Alpha	1.4	**0.8**	1	Normal	RIA
	3	Alpha	**1.5**	0	1	**Stop-End**	WiVik
P3	Baseline	Alpha	1.5	0	1	Normal	RIA
	1	**Freq**	1.5	0	1	Normal	RIA
	2	**Freq**	**1.0**	0	1	Normal	RIA
	3	**Freq**	**1.0**	**0.5**	1	Normal	RIA
	4	Alpha	**1.0**	0	1	Normal	RIA
P4	Baseline	Alpha	1.0	0	1	Normal	RIA
	1	Freq	1.0	0	1	Normal	RIA
	2	Freq	1.0	**0.5**	1	Normal	RIA
	3	Freq	**0.8**	0	1	Normal	RIA
P5	Baseline	Freq	0.9	0	1	Normal	RIA
	1	**Alpha**	**1.2**	0	1	Normal	RIA
	2	Freq	**1.2**	0	1	Normal	RIA
	3	Freq	**1.2**	**0.3**	1	Normal	RIA

Bold values indicate values that differ from baseline.

Table IV. Predicted vs actual text entry rate.

	Configuration	Actual TER (cpm)	Predicted TER (cpm)	Difference (cpm)	Error
P1	Baseline	5.57	6.03	0.46	8.27%
	1	5.73	5.89	0.16	2.83%
	2	7.02	7.34	0.32	4.51%
	3	6.02	5.22	0.79	13.20%
	4	6.73	6.37	0.35	5.25%
	Avg	6.21	6.17	0.42	6.81%
P2	Baseline	4.92	5.39	0.47	9.56%
	1	7.32	6.55	0.77	10.46%
	2	5.12	4.74	0.38	7.33%
	3	5.73	5.18	0.56	9.70%
	Avg	5.77	5.46	0.52	9.26%
P3	Baseline	5.19	4.57	0.62	11.93%
	1	5.59	5.44	0.15	2.72%
	2	6.79	6.67	0.11	1.67%
	3	5.67	6.15	0.48	8.39%
	4	6.03	5.81	0.22	3.58%
	Avg	5.85	5.73	0.32	5.66%
P4	Baseline	6.12	7.05	0.93	15.17%
	1	7.38	8.57	1.19	16.11%
	2	5.37	7.58	2.21	41.07%
	3	8.99	10.02	1.03	11.43%
	Avg	6.97	8.31	1.13	20.94%
P5	Baseline	4.96	5.48	0.52	10.49%
	1	5.28	4.55	0.73	13.86%
	2	5.90	5.23	0.67	11.42%
	3	5.15	4.98	0.17	3.32%
	Avg	5.32	5.06	0.65	9.77%
All	Grand Avg*	6.03	6.15	0.61	10.49%

*Grand Avg is the average of all averages, not all individual values.

The 95% confidence interval for model accuracy is shown in Table V. Figure 4 shows the model accuracy for all participants across both the baseline and experimental configuration.

The frequency of error-free selections and the frequency of each type of error are shown in Table VI. P2 was the most accurate (an average of 85.22% error-free selections) and P4 was least accurate (an average of 62.90% error-free selections). For all participants but P4, the most frequent error was failing to select the correct row. The lowest instance was 7.09% (P2) and the highest instance was 22.77% (P5).

For comparison, the scan rate was also entered into Damper's model [16], which assumes error-free performance. The predicted TER of both models was compared to the actual TER measured during sentence transcription with each configuration. As shown in Figure 5, our model was significantly more accurate than Damper's ($p < 0.05$). For our model, the average error for each participant ranged from 5.66% to 20.94%, with an overall average of 10.49% and a 95% confidence interval of [0, 22.8%]. For the Damper model, average model error was 79.7%.

Discussion

The purpose of this study was to test the accuracy of a model's predictions for the TER of individuals who use single switch row–column scanning as their method of communication. The actual TER generally ranged from five to seven characters per minute for all system configurations. Results showed that predicted TER was within one character per minute (cpm) of actual TER on average. The average difference between actual and predicted TER for all trials was 10.49% with a difference of 0.61 cpm. Due to interruptions during one participant test, a very large TER difference occurred. If this one trial is removed from the average calculations, the TER difference becomes 8.62% and 0.53 cpm.

Table V. 95% confidence intervals for prediction error.

	Units	Mean	StdDev	Low	High
Baseline	Error (%)	11.09	0.02	6.45	15.72
	Difference (cpm)	0.60	0.17	0.26	0.94
Experimental Conditions	Error (%)	9.81	0.09	0.00	27.30
	Difference (cpm)	0.60	0.51	–0.39	1.60
All Conditions	Error (%)	10.49	0.06	0.00	22.88
	Difference (cpm)	0.61	0.34	–0.07	1.28

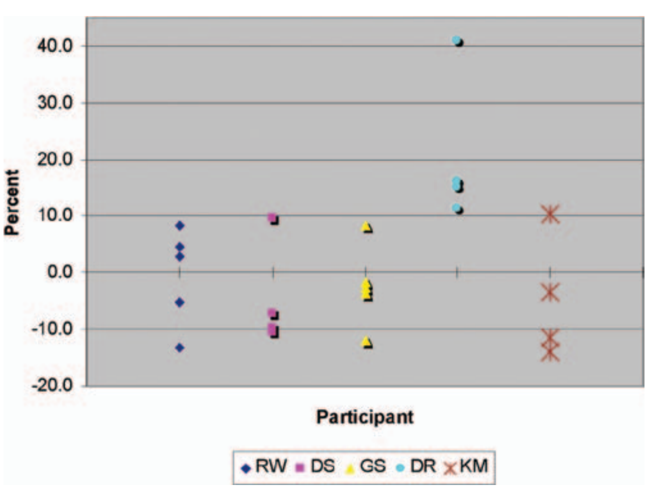

Figure 4. Predicted vs. actual TER.

We also considered the TER predictions for the Baseline and experimental conditions separately. Participants had more experience using the Baseline condition than any of the experimental conditions. More importantly, the error data used to make predictions about the Baseline condition was exactly right because it was recorded during the Baseline condition, meaning the only variation between actual performance and the model was caused by variations in switch press time. The predicted TER for the Baseline conditions was within one cpm of the actual TER, with a 95% confidence interval of [0.26, 0.94] cpm. For the experimental conditions, the predicted TER was within one cpm of actual TER for four of the five study participants, with a 95% confidence interval of [−0.39, 1.60] cpm.

Overall, TER predictions were lower than actual TER for 12 of the 22 trials (11 of 17 trials under experimental conditions; 1 of 5 trials under Baseline conditions). This under estimation can be attributed to the reduction of errors by participants after the Baseline condition. Three of the five participants reduced errors in all trials and one participant's error rates remained relatively stable. For example, P1 reduced errors for all configurations when compared to the baseline error-free selection rate of 70.98%. This could be attributed to practice, sentences that were easier to transcribe, or use of a configuration better suited to P1's tendencies. In some cases, however, errors increased relative to Baseline. For example, in configuration 3 of P3 the addition of a recovery delay caused more selection errors.

Since the baseline error probabilities/tendencies are used by the model for TER predictions, a change in performance in terms of errors can result in a difference between actual and predicted TER. This theory was tested for configuration 3 by running the model with the configuration 3 system parameters and error probabilities (in place of the baseline probabilities). The results showed a 4.3% difference in TER's compared to the 13.20% difference using the baseline error probabilities.

Although most TER predictions were within one character per minute of the actual TER, several issues may have affected accuracy. The most important issues appear to be the change in error frequency between the Baseline and experimental conditions and the need for error frequencies specific to row selections and column selections. The model currently assumes that errors occur at the same rate for row and column selections, which was clearly not true for the participants in this study.

The question of whether baseline probabilities apply equally well to all configurations is critical to determining how the model will be used in practice (or if the model can even be used at all). From both clinical and empirical observations, we know that some changes to the configuration of a scanning interface will change the likelihood of some errors. For example, adding a recovery delay is *supposed* to reduce the probability of a switch press occurring after the target row or column. Furthermore, some configuration options (like the option to reverse the direction of column scanning) take time

Table VI. Error frequencies.

	Configuration	Error Free Selection	Before Target Row	After Target Row	Before Target Col	After Target Col	No Row Selected	No Column Selected
P1	Baseline	70.98	1.34	4.02	0.89	0.89	20.54	1.34
	Config. 1	78.87	0.00	1.41	0.00	0.00	18.31	1.41
	Config. 2	76.14	1.14	6.82	0.00	0.00	15.91	0.00
	Config. 3	89.47	0.00	0.00	0.00	0.00	10.53	0.00
	Config. 4	78.46	0.00	3.08	0.00	0.00	18.46	0.00
	Avg	78.78	0.50	3.07	0.18	0.18	16.75	0.55
P2	Baseline	78.29	3.88	3.88	2.33	1.55	9.30	0.78
	Config. 1	82.86	0.00	2.86	0.00	1.43	12.86	0.00
	Config. 2	89.23	1.54	3.08	0.00	1.54	4.62	0.00
	Config. 3	90.48	3.17	1.59	0.00	3.17	1.59	0.00
	Avg	85.22	2.15	2.85	0.58	1.92	7.09	0.20
P3	Baseline	73.49	2.41	0.00	1.20	1.20	20.48	1.20
	Config. 1	71.15	3.85	3.85	1.92	0.00	17.31	1.92
	Config. 2	63.41	2.44	4.88	0.00	0.00	26.83	2.44
	Config. 3	64.47	7.89	2.63	2.63	1.32	21.05	0.00
	Config. 4	64.47	7.89	2.63	2.63	1.32	21.05	0.00
	Avg	67.40	4.90	2.80	1.68	0.77	21.34	1.11
P4	Baseline	62.30	20.77	2.73	3.28	1.09	8.20	1.64
	Config. 1	64.21	21.05	3.16	5.26	0.00	6.32	0.00
	Config. 2	55.96	27.52	2.75	4.59	5.50	1.83	1.83
	Config. 3	62.62	14.02	5.61	0.93	0.00	14.95	1.87
	Avg	61.27	20.84	3.56	3.52	1.65	7.83	1.34
P5	Baseline	54.84	2.42	14.52	0.00	4.84	22.58	0.81
	Config. 1	68.92	5.41	5.41	2.70	0.00	17.57	0.00
	Config. 2	61.18	4.71	4.71	0.00	0.00	28.24	1.18
	Config. 3	66.67	4.00	4.00	0.00	1.33	22.67	1.33
	Avg	62.90	4.14	7.16	0.68	1.54	22.77	0.83
All	Grand Avg	71.29	6.16	3.80	1.29	1.14	15.51	0.81

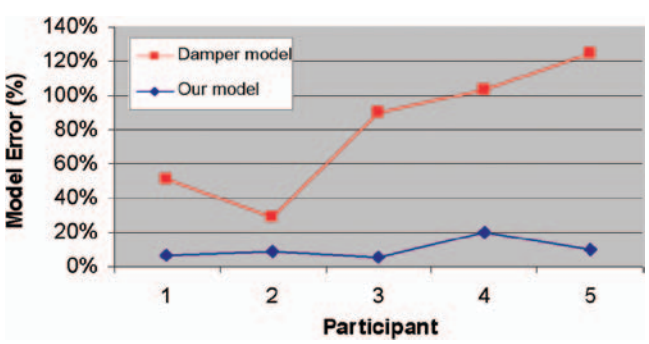

Figure 5. Results from preliminary studies, showing the % error of predicted vs. actual TER for our model as compared to the Damper model.

to learn to use efficiently, and performance may even decrease initially. Unfortunately, no one has characterized the relationship between most configuration options and the likelihood of each type of error occurring. One use of the model, then, is to examine trade-offs between configuration changes and error rates. For example, a clinician can use the model to calculate how much the likelihood of a late switch press error needs to decline to offset a 0.25 second increase in the recovery delay.

Additional issues include incorrect error classification, premature transcription termination, and the transcriptions timeouts. When re-examining the error correction data for P4, it was observed that several errors were not counted appropriately. The majority of P4's incorrect target selections occurred in a row other than the row where the target character was located due to inadvertent switch activations. These errors were different in that they did not fall neatly into the category of errors counted. The inadvertent selection was sometimes several row and columns away from the target character. This classification error caused an inaccurate error count for use in the model and higher TER predictions for this participant. That was even more significant for the second configuration test because of the increased amount of errors the participant accrued. This test was also interrupted and temporarily paused. Although the trial was reviewed to adjust the timing issue that occurred, it was difficult to discern the exact time of the interruption. The combination resulted in 41% difference between the actual and predicted TER.

Due to the placement of the Backspace key next to the Enter key on the on-screen keyboard, the opportunity to terminate a sentence transcription by accidentally selecting Enter instead of Backspace existed. This occurred halfway through the transcription of three sentences by participant P5. This may have affected P5's average TER due to the reduced amount of data available.

An automated error counting mechanism might improve the accuracy of the error count and classification. This could be integrated into the Sentence test itself. Timeouts and early transcription termination allow for the possibility of letter frequencies disproportionate to the frequencies of English usage utilized by the model. Premature termination of transcription can be resolved by simply rearranging the matrix layout to reduce the probability of selecting enter inadvertently.

After a sentence was presented for transcription in the Sentence Test, many participants would delay the start of sentence transcription until they had read and processed the sentence. This "processing" time was subtracted from the trial time when calculating actual TER. A "processing" time added as a model parameter would account for this delay. Inadvertent switch presses and subsequent selections were also observed during sentence transcription. The current model did not account for these events very well. The probability of this event occurring will be added as a model input parameter in future work.

Limitations related to the design of scanning method configuration options in on-screen keyboards (i.e. reverse scan, continue scan) relegated system configuration changes primarily to the areas of scan rate, recovery delay, and matrix layout, but manipulation of the settings did result in TER improvement for most participants. The maximum TER gains for each participant were in the range of 1.0–2.5 characters a minute. Each participant's highest TER occurred with a frequency-based keyboard configuration and a scan rate equal to or less than their Baseline scan rate. Even with increased TER, however, all participants were still below two words per minute.

Several observations were made throughout the various stages of the data acquisition and analysis. The four participants who normally used an alphabetic layout did not care for the frequency-based layout and preferred the alphabetic layout. Interestingly, the TER for all participants was higher when using the frequency-based layout compared to the alphabetic layout despite the participants lack of familiarity with (or dislike of) the layout. The majority of the time the increased TER was achieved even with a larger percentage of errors. The frequency-based matrix is designed to reduce scan steps and time to the most often used letters, but the lack of familiarity may have also increased participants' cognitive load.

Targets in the first row resulted in more errors than targets in other areas of the scanning matrix. This can be attributed to lack of recovery delay under most conditions. Some scanning configurations did include a recovery delay. Although the intention of this delay is to reduce errors for selections in the first row or column of a matrix, it actually increased errors for two participants. It appears that their switch activation was based on anticipation and was initiated prior to the highlighting of the desired matrix selection. As a result, their timing was disrupted and switch activation would occur prematurely. This occurred despite sentence transcription practice with the recovery delay setting.

System configurations with other features were tried, but participants either did not take advantage of these features (if they were available) or refused to use keyboards with these features altogether. One original goal of this study was to evaluate configurations using matrices with reverse, continue, and stop scanning functionality. Unfortunately, we could not address this goal because the cognitive load to use these features dissuaded study participants from considering them as an option during sentence transcription.

Conclusions

Row–column scanning is a very slow method of selection, but our results indicate that changes in the configuration of a row–column scanning interface can produce noticeable changes in performance. When configuring a row–column scanning interface, clinicians should consider the type of errors their client is likely to commit to target interface features to their client's specific needs. Clinicians should also keep in mind that some clients may not use some advanced interface features, even if they are available.

Our model of row–column scanning was much more accurate than a model that did not consider the likelihood of an error occurring. There is still room for improvement, however, and the results of the study will lead to additional modifications of the model. Models of other interfaces, such as keyboards operated by direct selection, might also be made more accurate if errors are considered.

Acknowledgements

The authors would like to thank the individuals who volunteered to participate in this research.

Declaration of interest

This research was funded by a grant from the National Institutes of Health (#2R44HD045015-03) and the National Science Foundation (#0333420). The authors report no conflict of interest.

References

1. Anson DK. Alternative computer access: a guide to selection. 1st ed. Philadelphia, PA: F.A. Davis Company; 1997. p 280.
2. Koester H, Levine SP. Modeling the speed of text entry with a word prediction interface. IEEE Trans Rehabil Eng 1994;2(3):177–187.
3. Lesher GW, et al. Acquisition of scanning skills: The use of an adpative scanning delay algorithm across four scanning displays., in RESNA 2002 Annual Conference 2002. Minneapolis, MN: RESNA Press; pp 75–77.
4. Lesher GW, Rinkus GJ. Leveraging word prediction to improve character prediction in a scanning configuration, in RESNA 2002 Annual Conference 2002. Minneapolis, MN: RESNA Press.
5. Simpson RC, Koester HH. Adaptive one-switch row-column scanning. IEEE Trans Rehabil Eng 1999;7:464–473.
6. Mankowski RE. Predicting Communication Rates: Efficacy of a Scanning Model. Pittsburgh, PA: Department of Rehabilitation Science and Technology 2009, University of Pittsburgh.
7. Simpson R, Koester H, LoPresti E. Selecting an appropriate scan rate: the ".65 rule". Assist Technol 2007;19:51–8; quiz 59.
8. Simpson RC, Koester HH, LoPresti EF. Evaluation of an adaptive row/column scanning system. Technol Disabil 2007;18(3):127–138.
9. Higginbotham DJ, Shane H, Russell S, Caves K. Access to AAC: present, past, and future. Augment Altern Commun 2007;23:243–257.
10. Angelo J. Comparison of three computer scanning modes as an interface method for persons with cerebral palsy. Am J Occup Ther 1992;46:217–222.
11. Horstmann HM. Quantitative Modeling in Augmentative Communication – A Case Study, in RESNA '90 Annual Conference 1990. Washington, DC: RESNA Press.

12. Abascal J, Gardeazabal L, Garay-Vitoria N. Optimisation of the Selection Set Features for Scanning Text Input, in Computers Helping People with Special Needs. Miesenberger K, et al., editors. Berlin: Springer-Verlag; 2004. pp 788–795.
13. Bhattacharya S, Basu A, Samanta D. Computational modeling of user errors for the design of virtual scanning keyboards. IEEE Trans Neural Syst Rehabil Eng 2008;16:400–409.
14. Bhattacharya S, Samanta D, Basu A. User errors on scanning keyboards: empirical study, model and design principles. Interacting with Computers 2008;20:406–418.
15. Bhattacharya S, Samanta D, Basu A. Performance models for automatic evaluation of virtual scanning keyboards. IEEE Trans Neural Syst Rehabil Eng 2008;16:510–519.
16. Damper RI. Text composition by the physically disabled: a rate prediction model for scanning input. Appl Ergon 1984;15:289–296.
17. Lesher GW, Moulton BJ, Higginbotham DJ. Techniques for augmenting scanning communication. Augment Altern Commun 1998;14(2):81–101.
18. Venkatagiri HS. Efficient keyboard layouts for sequential access in augmentative and alternative communication. Augment Altern Commun 1999;15(2):126–134.
19. Simpson RC, Mankowski RE, Koester HH. Modeling one-switch row-column scanning with errors and error correction methods. Open Rehabilitation J 2011;4:1–12.
20. MacKenzie S, Soukoreff W. Phrase sets for evaluating text entry techniques. in CHI '03 extended abstracts on Human factors in computing systems. Ft. Lauderdale, FL, USA: ACM Press; 2003.