Noname manuscript No.
(will be inserted by the editor)

Enhancing typing performance of older adults on tablets

Élvio Rodrigues · Micael Carreira · Daniel Gonçalves

Received: date / Accepted: date

Abstract Touchscreen interfaces are increasingly more popular. However, they lack haptic feedback, making it harder to perform certain tasks, such as text-entry, where users have to constantly select one of many small targets. This problem particularly affects older users, whose deteriorating physical and cognitive conditions, combined with their unfamiliarity with technology, can discourage them from using touch devices. The goal of this work is to thoroughly understand older adults touch typing behavior, in order to develop text-entry solutions more appropriate to their needs, which will enhance their typing performance. On a first phase a baseline QWERTY keyboard and five different variants were developed, that mostly used a text prediction algorithm to suggest the most probable keys or words. These keyboards were evaluated on a baseline study with 20 younger adults in order to find the most promising ones, which were then used in a study with 20 older adults. The older adults study revealed more about their typing behavior, and therefore created four new variants to be used in a simulation study. Results show that visual changes should be kept to a minimum; touch points should be shifted upward and to the opposite side of the hand used to type; single touch keyboards perform better than multi-touch; and omitting keys below a certain time threshold minimizes accidental insertions.

Keywords Older adults · Touchscreen · Text-entry · Tablet · Pre-Attentive Interface · Accessibility

Élvio Rodrigues
Instituto Superior Técnico, Avenida Rovisco Pais, 49, 1050-001 Lisboa, Portugal
E-mail: elvio.rodrigues@ist.utl.pt

Micael Carreira
INESC-ID, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
E-mail: micaelcarreira@ist.utl.pt

Daniel Gonçalves
INESC-ID, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
E-mail: daniel.goncalves@vimmi.inesc.pt

1 Introduction

Several studies have presented evidence indicating that the population is aging across the world [48]. This is a consequence of socioeconomic development, where living conditions improve, access to health care increases, and birth rates are lower than the past. Therefore we are more likely to be physically and cognitively active until older ages. Still, as we age, we experience changes on several dimensions. This includes perceptual, psychomotor, cognitive, physical, psychological and social changes [37, 11, 24, 33, 7]. Besides these changes, older adults are also more likely to suffer from several diseases that also debilitate their capacities. Still, this work is focused only on healthier older adults. Therefore, older adults with Parkinson, Alzheimer, Osteoporosis, among other diseases, are out of the scope of this work.

At the same time studies show that touchscreen devices are widely used worldwide, with an increased tendency to grow[1]. There is also some evidence that shows that, older users, in particular, benefit with the use of such devices, since it allows them to interact more easily and direct with digital content [25, 47, 2, 44, 3, 38, 46]. In this context it is expected that these kinds of devices will be increasedly adopted by older adults. This is an opportunity to develop more inclusive interfaces for older

[1] http://www.comscoredatamine.com/2012/03/exponential-tablet-adoption-in-2011-ushers-in-era-of-convergent-consumption/

adults, since these kind of devices rely much less on physical keys, to rely more on elements controlled by software. However, touch devices have also the disadvantage of lacking the haptic feedback of physical buttons, which makes it harder to accurately select targets.

This characteristic particularly hinders certain tasks, such as text-entry, on which the user has to constantly select one of many small targets. Furthermore, this task is of great importance since it is transversal to many applications such as, basic communications, managing contacts, editing documents, note taking, web browsing, searching, among others [34]. Although several studies focus on virtual keyboards for smartphones and tablets for young users, not much research has been performed regarding older adults. Since these new and updated technologies are often designed for the younger generations, who are familiar with using such technologies, there is a need to understand what are the special needs of the older adults. These needs include not only the fact that older adults are generally less experienced with technology, but also the fact that aging implies several changes that might limit their capacity to interact with such devices, if not taken into account. Furthermore, the literature is more focused on smartphones than tablet devices. Even though tablets can overcome some of the smartphones' problems, such as "buttons being too small", they are also touch devices, and therefore share several problems such as the lack of haptic feedback. This is why it was decided to focus this research on understanding and developing more appropriate solutions to improve typing performance of older adults on virtual keyboards for tablet devices.

Therefore two studies were performed to better understand how one can improve typing speed and/or reduce the error rate of older adults on tablets. On a first approach these keyboards were evaluated on a baseline study with 20 younger adults, since these were easier to find. Users were asked to enter sets of individual sentences using the different keyboards, while metrics such as *Words Per Minute* (WPM) and *Minimum String Distance* (MSD) were collected. Afterwards, on a second approach, a similar study with 20 older adults was performed, with the most promising variants of the previous study. From the older adults study more data was revealed about their typing behavior, and therefore created four new variants to be used in a simulation study along with *Shifted* and *Size Invisible* variants. One of the new variants downgrades the baseline QWERTY keyboard into a single touch keyboard (*Single Touch* variant), while the others reject interactions based on time features (*Intra-key*, *Inter-key* and *Combined Timed* variants).

The main contribution of this work is a thorough understanding of the older adults' touch typing behavior on virtual keyboards for tablet devices. This knowledge will help the development of virtual keyboards that are more accessible and appropriate for the older adults needs.

The remainder of this article is organized in twelve sections. **Section 2** presents some works that try to understand how people interact with touch screen devices in general, and then particularly with virtual keyboards. **Section 3** presents two types of prediction techniques to anticipate what the user is going to type: *word prediction* and *next letter prediction*. An evaluation of the efficiency of the prediction system is also presented. **Section 4** presents the traditional QWERTY keyboard and the five initial keyboard variants. **Section 5** presents a user study we conducted with 20 younger adults and 6 of the developed virtual keyboards. **Section 6** presents the results of the aforementioned study. WPM, error rate and typing errors for each variant are discussed. Touch typing behavior for this type of user is also investigated. **Section 7** presents a user study we conducted with 20 older adults and 3 of the developed virtual keyboards. **Section 8** presents the results of the aforementioned study. WPM, error rate and typing errors for each variant are discussed. Touch typing behavior for this type of user is also investigated. **Section 9** presents four new keyboard variants that take into account what was learned from the previous studies. **Section 10** presents a simulation study that uses the QWERTY input data from the older adults study to simulate 6 different variants. Error rate and typing errors are discussed. **Section 11** presents design implications for text-entry solutions in tablet devices for younger and older adults. **Section 12** presents the major results and an overall discussion of the key findings of our research. Additionally, several possibilities are presented regarding future work .

2 Related Work

Generally, older adults easily adapt to touch technology. Loureiro et al. [27] analyzed different aspects of eight touch-based tabletop interfaces for the older adults. In all surveyed works, they concluded that touch yields a natural, direct and intuitive way of interaction with a device, allowing easier human-computer interaction for older adults. Indeed, people with low or no computer literacy found using touch screens easy and motivating [25,47,2,44,3,46,38].

Stone [41] was concerned with the fact that devices such as the iPhone and iPod Touch are not so accessible for older adults, since buttons are too small. The author argues that this kind of device should have a gesture that allows to choose multiple sizes for interface elements (fonts, buttons and icons). Furthermore, this should also be extended to virtual keyboards, giving the opportunity to the user to choose between different layouts, depending on the current situation and the users' capabilities. Other than the already established layouts (traditional QWERTY keyboard and 12 button mobile phone interface) a binary interface could also be used. In this case, the characters would be divided in two sets of approximately equal total probability and the first choice is to select in which of the two sets the desired character is. The process continues until a set is chosen containing only one character. However, no implementation nor experimental evaluation was performed.

Other researchers have focused on optimal target size, spacing and positioning to derive a set of guidelines that should improve the usability of touch interfaces for older adults [23]. Indeed, Hwangbo et al. [22] found that the target size is an important factor in pointing performance. They recommend square targets with a side of at least 12mm, since it achieved the best performance in their user study. They also found that when target size reaches this level, the spacing between targets looses importance. In this case, the layout that maximizes performance has narrow target spacing (1mm). However, these studies neglect the particular case of text-entry, which can be considered one of the most difficult tasks to perform on touch devices, due to the large number of targets and small key size and spacing.

Nicolau et al. [35] focused on the particular problem of text-entry. They performed a user study with 15 older adults, measuring the speed and accuracy of participants while performing text-entry tasks, both on a smartphone and a tablet. They also analyze the users' hand tremor profile and its relationship to typing behavior. After analyzing the results, the authors derive a set of guidelines that should be taken into account when developing accessible virtual keyboards. Among others, they concluded that hand tremor is strongly correlated with substitution errors; there is a right-bottom substitution pattern; keys should be wider than taller; and propose a user-dependent solution for inter-key interval to address insertion errors by accidental touches.

Trewin et al. [43] presented a user model which identifies keyboard users who are having difficulty with the default keyboard configuration, and recommend an appropriate alternative configuration, which can help reduce input errors, and make the keyboard easier to use. Four of the most commonly observed difficulties with keyboards are: *long key press errors* (when a key is unintentionally pressed for longer than the default key repeat delay), *difficulty in using modifier keys* (particularly harder for one handed typists), *additional key errors* (when the press and release of both the unintended and intended keys are overlapped in time) and *bounce errors* (when the user unintentionally presses a key more than once). Results show that the model was able to identify the participants with the greatest difficulty in all four of the areas studied, and to make good predictions of the configuration that would suit them. The evaluation also showed encouraging results on the stability of the model in general.

Although the body of work regarding older adults is relatively small, there is an extensive body of work that tries to understand and maximize the performance of young adults without physical degradation and familiarized with the QWERTY keyboard, on touch interfaces. Henze et al. [18] argue that shifting touch events can improve the typing error rate. The authors developed a typing game that recorded how users touch the standard Android keyboard, so they could investigate their typing behavior. After analyzing the data, the authors concluded that events are systematically skewed towards the lower-right corner of keys.

Findlater et al. [10] opted for an adaptive keyboard. The authors evaluated two novel personalized keyboard interfaces specifically for ten-finger typing, both of which adapt their underlying key-press classification models. One of the keyboards also visually adapts the location of keys, while the second always maintains a visually stable rectangular layout. Results show that the *NonVisual-Adaptive* keyboard provided a typing speed improvement over *Conventional* (baseline keyboard), but *Visual-Adaptive* did not (visualizing adapted key layouts negatively impact speed).

As noted by Cheng et al. [8] on a recent study, people use different hand postures to type on tablets, depending on the situation. This study showed that 98% of the users preferred different keyboard layouts and positions depending on how they were holding these devices. The authors developed *iGrasp*, which automatically adapts the layout and position of virtual keyboards based on how and where users are grasping the devices without requiring explicit user input.

Another way to significantly reduce the error rate of soft keyboard usage is through language models. Several approaches to highlight keys have been studied which involve making the rendered keys larger or smaller, depending on their likelihood [1]. The authors reported that users were faster and more accurate with this variant than with the regular QWERTY keyboard. Gunawardana et al. [15] developed a method that expands or contracts the keys' underlying area, keeping the visual feedback intact, based on a language model. A simulation suggests that it reduces the error rate.

The absence of haptic feedback on touchscreens makes it harder to perform certain tasks. Poupyrev et al. [39] performed a study were participants were asked to perform certain tasks in three different conditions: with audio feedback, with tactile feedback and with no feedback. Tactile feedback was exceptionally well-received by the users who often remarked how similar tactile feedback felt to an actual mechanical switch. Brewster et al. [6] also investigated tactile feedback on mobile interactions, but in this case specifically in text entry tasks. Results showed that with tactile feedback users entered significantly more text, made fewer errors and corrected more of the errors they made. A second study was performed with users seated on an underground train to see if positive effects transferred to realistic use. There were fewer beneficial effects, with only the number of errors corrected significantly improved by the tactile feedback. However, strong subjective feedback was found in favor of the tactile display. A similar study was again conducted by Brewster et al. [20], with the added condition of a physical keyboard. Keyboards were evaluated in both static and mobile environments. The results showed that the addition of tactile feedback to the touchscreen significantly improved finger-based text-entry, bringing it close to the performance of a real physical keyboard. A second experiment showed that higher specification tactile actuators could improve performance even further.

Dunlop et al. [9] presented two new touchscreen keyboard layouts based on Pareto front optimization and three design metrics: minimizing finger travel distance in order to maximize text entry speed, a new metric to maximize the quality of spell correction by reducing tap ambiguity, and maximizing familiarity through a similarity function with the standard QWERTY layout. In an initial study users were faster with the QWERTY keyboard (21 wpm) when compared with the new keyboard (13 wpm). After four short trial sessions, users were able to improve their typing speed with the new keyboard up to 18 wpm. Oulasvirta et al. [36] also developed a new keyboard layout entitled KALQ. which is a split keyboard for fast text entry with two thumbs for mobile touchscreen devices. Users were able to reach a rate of 37 words per minute (with a 5% error rate) after a training program with the new layout.

As has been shown, previous studies are mainly focused on finding solutions for able-bodied adults. Although some studies have already analyzed the touch patterns and the optimal target size, spacing and positioning for older adults, none have presented and evaluated different alternatives to improve the typing experience for older adults.

3 Text Prediction

In order to develop more advanced variants of the virtual QWERTY keyboard, two types of prediction were used to anticipate what the user is going to write: word prediction and next letter prediction. If the prediction system is able to guess correctly, the number of keystrokes needed to write a sentence decreases. This way, it can also enhance the speed of writing and reduce the physical effort required to compose messages. In addition, the prediction software may also fix spelling mistakes, reorder sentences and more generally enhance the quality of the composed messages. The most advanced prediction systems have learning features, are able to make inferences and are adaptable. [14].

There are several techniques to predict the text a user is trying to input, some more complex than others. However, by increasing the complexity of the predictions systems, the prediction results only increase marginally [14]. This way, and since the aim of this study was not developing a novel and more efficient prediction algorithm, the authors opted for a simplistic one. The proposed predictor only takes word frequencies [12,16,21,42,45] into account and, when the user writes the beginning of a word, the system offers the most probable words beginning with the same character(s).

To implement the word prediction system, the CETEMPúblico Portuguese text corpus [2] was used, which contains approximately 180 million words. From that corpus the word frequencies were processed and then stored in a dictionary structure [13] that contains all the information about each word and its prefixes frequencies, so that the information can be efficiently accessed. When

[2] http://www.linguateca.pt/cetempublico

the user is typing, the predictor shows an ordered list of the most frequent words that start with the typed prefix.

After implementing the word prediction system, it was decided that the next letter prediction should be based on the same algorithm in order to avoid the case of the letter prediction algorithm suggesting a letter that is not present in any of the suggested words. For instance, imagine the user wants to write "home", and at this point has already typed "ho". If the letter prediction algorithm suggests the letter "t" (hot) and the word prediction system suggests the word "home" it could be confusing for users. It was thus decided to implement the letter prediction algorithm through the word prediction system. What happens is, since the most probable word is "home", and the user has already typed "ho", the letter prediction algorithm will choose to highlight the "m" key.

3.1 Prediction Results

Since most text prediction methods are heterogeneous, and since the measurements offered by authors are based on heterogeneous parameters (not always clearly described) [14], it is hard to assess how well the algorithm performs when compared with others. Therefore, to evaluate the efficiency of the implemented prediction system, 88 sentences were used that were extracted from a written language corpus from another study [34]. Each sentence had 5 words with an average size of 4.48 characters and a minimum correlation with the language of 0.97. Figure 1 shows the result of the word prediction. Only words between 6 and 12 characters long were considered, because any smaller lengths do not represent considerable savings in key presses, and above that there were not many words in our set of sentences.

As expected, the more the suggested words, the greater chance of success. However, the success rate does not seem to increase much when presenting a list of more than 6 words (only an increase of 3% between suggesting 6 and 7 words). Moreover, it must be taken into account that the more words are suggested, the more cognitive effort is required for the users to process the suggestions list. Therefore, there should be a balance between the number of words suggested (which affect directly the success rate) and cognitive effort required to process the suggestions list (which increases with the number of words).

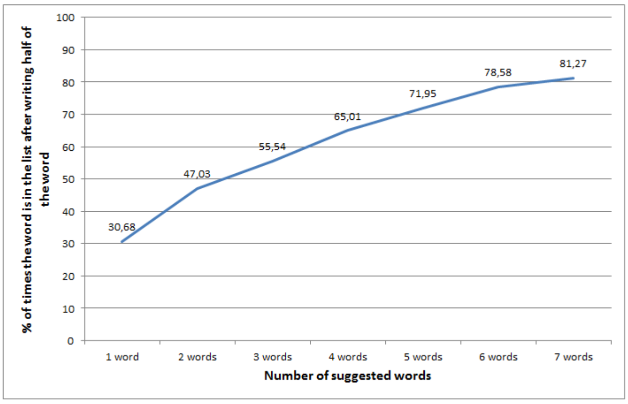

Fig. 1 Performance of the word prediction algorithm.

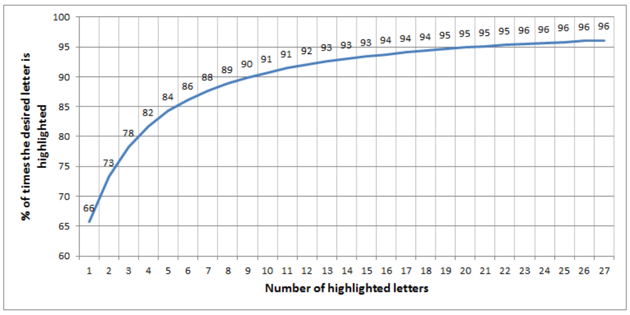

Fig. 2 Performance of the letter prediction algorithm.

The same evaluation was also performed for next letter prediction. As can be seen in Figure 2, it is much easier to correctly predict the next letter (space included) than to predict the full word the user is typing, since the same next letter is shared for several words.

Until 4 letters, the success rate increases from 4-7% and after that, only an increase of 0-2% is found. Note that 100% success was never hit, even if all the letters of the keyboard were highlighted and that is because the sentences contained a surname that was not in the prediction system, so the system could not predict it.

4 Keyboard alternatives

In this section the developed keyboards for the baseline and elderly studies will be presented. The rationale for each variant and the features that make them different from each other are explained. Section 9 presents the remaining keyboards that were developed after what was learned from the user studies. All the keyboard variants were implemented as a Windows Metro App for Windows 8.

4.1 Traditional QWERTY Keyboard

The *traditional QWERTY* keyboard is the baseline keyboard developed. It is similar to the other virtual keyboards existing on most touch devices, apart from the fact that letters are entered using a lift-off strategy; that is, letters are only inserted when the user lifts its finger from the keyboard. This strategy avoids multiple insertions, since older users' key presses are usually long [34]. Also, a letter is only inserted if the released key is equal to the pressed key.

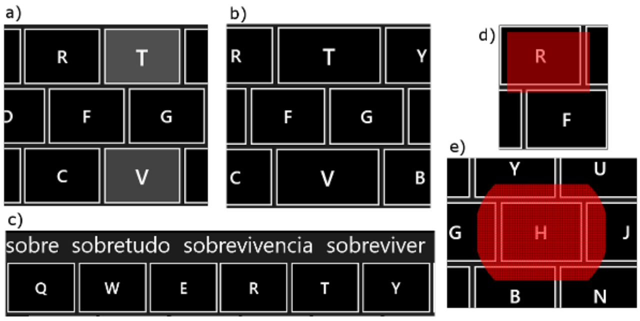

Fig. 3 (a) Color variant; (b) Width variant; (c) Predicted Words variant; (d) Shifted variant; (e) Size Invisible variant. Red areas are not visible to the users.

4.2 Color variant

The *Color* variant uses the *prediction system* described in section 3 to highlight the next most likely letters of the current word. Regarding the number of keys to highlight, it was decided to highlight four keys because Faraj et al. [1] have previously evaluated highlighting one, two and four keys, obtaining better results with the latter. Also, the results of the letter prediction evaluation showed that highlighting four letters has an increased success rate when compared to highlighting fewer letters. Therefore, it can be argued that highlighting four letters is a good balance between the cognitive effort the user needs to apply to choose between the highlighted keys and success rate of the algorithm. It was decided to highlight the keys by changing their color from black to gray, which is a neutral color (Figure 3 (a)), to ensure that cultural connotations associated with particular colors are avoided (e.g., green and red colors have positive and negative connotations, respectively). Moreover, the size of the key's label is increased. The highlight is continuous: the more probable the letter, the brighter the color and bigger the label on the key. The biggest goal of this variant is to help users who are not completely familiarized with the QWERTY layout, to locate faster the key they want to type. Users are also expected to commit fewer errors by noticing if they are about to press a key that is not highlighted, or by acknowledging they missed a key press. Several studies used similar approaches in other contexts [30, 28].

4.3 Width variant

The *Width* variant uses the same principle as the *Color* variant. The difference is that, instead of highlighting the keys by changing its color, it highlights the keys by increasing their width by 30% (Figure 3 (b)). However, for this variant a continuous increase in size based on the probability of the letter was not used, because it was much harder to tell which buttons were highlighted if the size increased was little. As happens with the *Color* variant, the label of the key increases in size proportionally to its probability. With this variant we expect users to commit fewer substitution errors by hitting the desired key instead of the neighbor keys, since the most likely keys are bigger. A previous study [1] has shown that this approach can improve both the input and error rates of the typed sentences on smartphones.

4.4 Predict Words variant

The *Predict Words* variant is a common alternative that can be selected as typing method in most touch devices. While the user is typing, a list of the most likely words is shown in a horizontal ribbon above the keyboard (Figure 3 (e)). If the word the user wants to type is on the suggested list, some key touches can be saved by tapping it so the full word along with a space character will be inserted. In the literature, there is no conclusive study about the optimum number of words to suggest [14]. Since there is a trade-off between the number of suggested words (that directly affects the success rate) and the cognitive effort required for the user to process the list, it was decided to suggest 4 words. Although this is not a novel approach, the aim was to systematically confirm if this variant would possess any advantage over the normal QWERTY keyboard, either in typing speed or quality of the transcribed sentences (fewer errors). It is a fact that users save some time by tapping fewer keys, but they also waste time in the cognitive effort of continuously checking the suggestion list.

4.5 Shifted variant

The approach of shifting the real touch area of keys from its visual representation is also common in many virtual keyboards [17,18]. In small touch devices, such as smartphones, this approach has proven its benefits [17, 18]. However, no systematic studies have been performed for tablet devices. These devices vary from the former not only in screen size, but also in the typing posture users assume when using them; when using smartphones users usually type with the two thumbs, while with tablets they can type with all fingers. Previous studies have consistently shown that users' touch points are skewed to the bottom-right, on smartphone devices [17,18]. Still, neither of those studies indicates the optimum shift needed to apply to compensate the users' tendency to touch in the bottom right of targets. Taking this into account, it was chosen to deviate the real touch area of the key 10% of the key's height to the bottom, and 10% of the key's width to the right in our implementation (Figure 3 (c)). Note that 10% was a value chosen by us, because it seemed to work well. The user studies will help us verify if this is the best value indeed. Visually for the user, this variant is exactly the same as the *QWERTY* keyboard. Users are expected to commit fewer *neighbor substitution* errors with this variant.

4.6 Size Invisible variant

Similar to the *Width* variant already described, this variant increases the size of the most likely keys. However, this variant does it only internally; to the users it remains visually the same as a regular *QWERTY* keyboard. This approach has also been the aim of previous studies [15]. In the present implementation, the likely keys' width was increased to 50% (25% to the left and 25% to the right) and 50% in height (25% to the top and 25% to the bottom). Note that these values were chosen by the authors, because they seemed to work well. The user studies will help verify if these are indeed the best values. A maximum distance from the center of the key (125% of half of its original diagonal) was also imposed so that the final shape of the touch area of the highlighted keys had rounded corners (Figure 3 (d)). The touch point is always assigned to the key that has the lowest Euclidean distance from its center to the touch point. With this variant users are expected to commit fewer *neighbor substitution* errors by hitting the desired key instead of the neighbor keys, since the most likely keys are internally bigger.

To the authors' knowledge, neither of these variants were evaluated with older adults.

5 Baseline Study

To evaluate the performance of the developed variants, a study with 20 younger adults was conducted. The following subsections, depict the performed evaluation.

5.1 Participants

Twenty participants, 13 males and 7 females, took part in the user study. All ages were between 19-30 years, except for one user that was 52 years old. Only 2 participants were left handed. All participants had a college degree, except one that had a high school degree. Every single participant had previous experience with QWERTY keyboards and use them every day. Most participants (13) also use virtual QWERTY keyboards on a daily basis, 1 weekly, 4 rarely, and only 2 had never used them at all. Only 6 participants use a tablet at least weekly, while 13 use virtual keyboards on smartphones daily.

5.2 Procedure

The user study had two main phases: training and evaluation. At the beginning of the first phase, the researchers explained to each participant that the aim of the study was to evaluate each variant of the virtual QWERTY keyboard, and not the users themselves. Users were free to type in the position they found most comfortable: with one or two hands, with the tablet supported on the table, on the lap or on the free hand. Since participants were not familiar with the developed keyboard variants, they were allowed to try each keyboard variant for two minutes, except for *Shifted* and *Size Invisible* variants. These variants behaved visually just like the *QWERTY* condition, so users were not aware about their existence at this point.

The task in both phases consisted of copying a sentence that was displayed at the top of the screen (see Figure 4). After entering the sentence, the user could proceed to the next sentence by pressing the "Próxima Frase" ("Next Sentence") button. Copy typing was used to reduce the opportunity for spelling and language errors, and to make error identification easier. Both required and transcribed sentences were always visible. Sentences were randomly chosen from a set of 88 sentences extracted from a Portuguese language corpus such that no sentence was written twice per participant. These were the same sentences used to perform the text prediction evaluation, which were extracted from another study [34]. Each sentence had five words with an

Fig. 4 Screen shot of the evaluation application.

average size of 4.48 characters and a minimum correlation with the language of 0.97. Backspace/delete keys were removed, so users were instructed to continue typing if an error occurred. The approach followed has both advantages and disadvantages. The main reason why the delete and backspace buttons were removed, is because there was concern that different users would employ different correction strategies, that would impact typing speed differently. The authors were also concerned that different correcting strategies would make it difficult to compare results from different users. The downfall of this approach could be that, users cannot correct errors, which is not the most natural behavior, since they will be able to correct errors in a real situation. Still, it can be argued that this is not limiting, since access to all the data related to the performed errors still exists. This is why this approach was selected instead of Soukoreff et al. [40].

On the evaluation phase, participants were instructed to type the sentences as quickly and accurately as possible. Each user was asked to type 5 sentences for each variant, the first one being a practice trial. Before the test, users were informed that they would perform tests on 2 more variants that were only slightly different from QWERTY. During the evaluation users did not know whether they were using the *Shifted* or the *Size Invisible* variants, or even the traditional QWERTY. This way, it was ensured that their typing pattern was not influenced by that knowledge. The order of conditions was counter balanced to avoid bias associated with experience. In the end, users were asked to answer a survey with some demographic data, as well as satisfaction regarding each variant. The whole process took approximately thirty minutes per participant.

5.3 Apparatus

A Samsung ATIV Smart PC Pro 11.6" was used in the user study. Each key had 20mm of width and 15mm of height. Visually, there is a space of 2mm between keys, horizontally and vertically. However, this implementation does not allow pressing between keys: each touch is always assigned to a key. This makes the keyboard more responsive, thus avoiding the frustration of performing a touch that does not produce a character. This is inline with the recommendations provided by Hwangbo et al. [22] since the results of their study show that users perform better with a narrower target spacing, when using targets wider than 12mm. A letter was entered when the user lifted the finger from the key. All participants' actions were logged through the evaluation application, for posterior analysis.

5.4 Dependent Measures

Performance during the text-entry task was measured by several quantitative variables: *Words Per Minute* (*WPM*), *Minimum String Distance* (*MSD*) error rate, and character-level errors (substitutions - incorrect characters, insertions - added characters, and omissions - omitted characters) [29]. Qualitative measures were also gathered in the end of the experiment by debriefing each participant.

5.5 Design and Analysis

A within-subjects design was used, where each user evaluated all conditions. For each keyboard condition each user entered 5 sentences (1 practice + 4 test), resulting in a total of 30 sentences per user. In summary the study design was: 20 users × 5 sentences × 6 keyboards.

Shapiro-Wilkinson tests of the observed values for *WPM*, *MSD* error rate, types of errors were performed, to assess if they were normally distributed. If they were, parametric statistical tests, such as repeated measures ANOVA, t-test, and Pearson correlations were applied. On the other hand, if measures were not normally distributed, non-parametric tests were used: Friedman, Wilcoxon, and Spearman correlations. Bonferroni corrections were used for post-hoc tests.

6 Baseline Study Results

In this section input speed and accuracy for the six conditions (*QWERTY* keyboard and *Color*, *Width*, *Pre-*

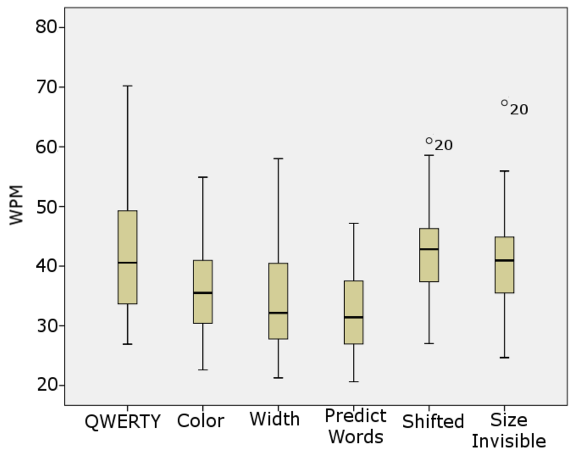

Fig. 5 Typing speed for each variant with outliers.

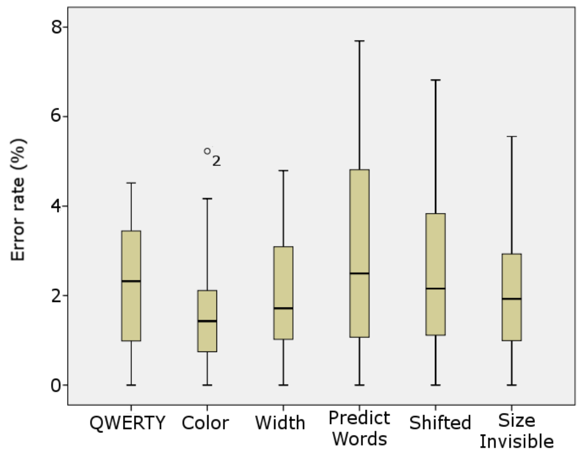

Fig. 6 Error rate for each variant with outliers.

dict Words, *Shifted* and *Size Invisible* variants) are analyzed, focusing on the type of errors. Outliers were found using the *labeling rule* [19]. They were removed to perform the statistical analysis described in this section.

6.1 Input Speed

In this subsection input performance is analyzed regarding speed for each keyboard condition. To assess speed, the *Words Per Minute* (*WPM*) [31] text input measure was used, calculated as:

$$\frac{(transcribed\,text - 1) \times (60\,seconds \div time\,in\,seconds)}{5\,characters\,per\,word} \quad (1)$$

"Transcribed text" (in equation 1) is the number of characters the user inserted. Minus 1, because the timer only starts after the user presses the first key. Figure 5 illustrates WPM for *QWERTY*, *Color*, *Width*, *Predict Words*, *Shifted* and *Size Invisible* conditions.

Effect of the virtual keyboard on typing speed. Repeated ANOVA measures revealed significant differences between keyboard variants on text-entry speed ($F(5, 90) = 18.787p < 0.001$). Bonferroni post-hoc tests showed significant differences between *QWERTY* and *Color*, *Width* and *Predict Words* variants, meaning that participants typed significantly slower in these 3 variants. This result is comparable to the results obtained by Henze et al. [18], on which users were slower when visual feedback regarding the tapped area was shown.

The *Predict Words* variant is also slower than the *traditional QWERTY*, which indicates that the saved keystrokes do not make up for the time and cognitive effort required to constantly check the suggestions list. Having said that, it is expected that these variants will help older users improve speed, since most of them might not be acquainted with the QWERTY layout. Therefore, if the *prediction system* suggests the right letter, the user will not need to waste time scanning all the keys. As expected, there were no significant differences between the input rate of the *traditional QWERTY* and the *Shifted* and the *Size Invisible* variants.

6.2 Quality of Transcribed Sentences

The quality of the transcribed sentences was measured using the *Minimum String Distance* (*MSD*) error rate, calculated as:

$$\frac{MSD(required\,text, transcribed\,text)}{Max(|\,required\,text\,|, |\,transcribed\,text\,|)} \times 100 \quad (2)$$

First the MSD algorithm is applied to calculate the number of operations required to transform the transcribed text into the required text. This is divided by the one that has more characters, and finally multiplied by 100. Figure 6 illustrates the *MSD* error rate for the *QWERTY* keyboard and its variants.

Effect of the virtual keyboard on quality of transcribed sentences. All variants slightly improved the overall quality of the typed sentences, since the error average was highest on *QWERTY*. However, a Friedman test did not reveal significant differences between keyboard conditions on error rate ($\chi^2(5) = 2.933,\ p =$

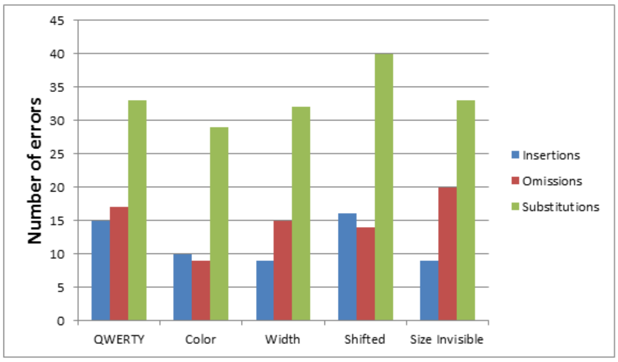

Fig. 7 Contribution of each type of error for the total amount of errors on each variant.

0.710). But one should not forget that these results regard all types of errors, and that, for instance, the *Shifted* and *Size Invisible* variants only aim to correct neighbor substitution errors. These variants will be further analyzed these variants in subsections 6.3.1 and 6.3.2, respectively. Furthermore, a t-test was performed between the *QWERTY* and the *Color* variant, since this was the variant with least errors. The t-test confirmed that there are significant differences between these variants on error rate ($t(17) = 3.151$, $p = 0.006$). This means that, despite the fact that participants were already familiarized with the QWERTY layout, they were committing fewer errors with this variant, especially omission errors. This variant will be further analyzed in subsection 6.3.3.

6.3 Typing Errors

This section further explores the types of errors that occurred in each variant. The errors are categorized as: insertions - added characters; substitutions - incorrect characters; and omissions - omitted characters [29]). The errors each variant was able to correct were also explored.

Figure 7 shows the contribution of each type of error for the total amount of errors, on each variant. *Insertion* errors are the least common type of error on the *QWERTY* keyboard, *Width* and *Size Invisible* Variants. On the *Color* and *Shifted* variants this slot belongs to the omission errors. *Substitution* errors are consistently the most common error, across variants. This result differs from the result reported by Nicolau [34]; that is, the most common error among older adults were omission errors, which further emphasizes the differences between populations.

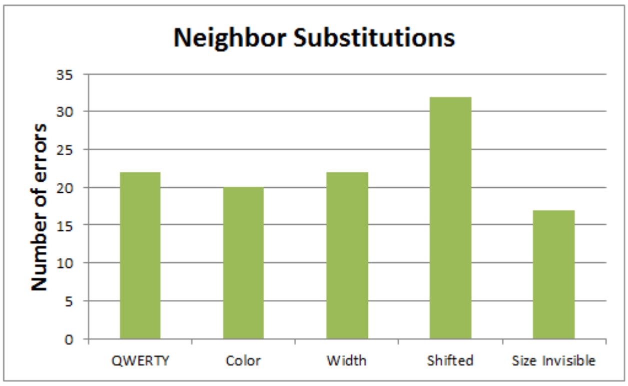

Fig. 8 Number of neighbor substitution errors on each variant.

6.3.1 *Shifted variant*

This section presents the results obtained with the Normal *Shifted* variant (10% shift) and the Improved *Shifted* variant (optimum shift). The latter obtained better results.

Normal Shifted variant. As stated previously (Subsection 4.5), the Shifted variant only aims to correct *neighbor substitution* errors, which occur when the user touches a key immediately adjacent to the expected key. Still, as can be seen in Figure 8 *neighbor substitutions* were more frequent on the *Shifted* variant than on the others. This is due to the vertical and horizontal shifts we applied to the touch points. Although other authors [34, 17, 18] reported that users generally touch on the bottom-right of targets, they never clarified what was the optimum shift to be applied in order to compensate the skewness. Therefore, these poor results were obtained, because a non-optimum value was used.

Improved Shifted variant. In order to find the optimum horizontal and vertical shifts, the necessary shift to transform each bad touch point into a good one was calculated, and the necessary shift that avoids transforming good touch points into bad ones. It was found that shifting the touch points to the top 6% of the height of the key, and to the left 7% of the width of the key yields the best results. **This allowed the *Shifted* variant to intervene correctly 87.18% of the times, correcting 48.65% of *neighbor substitution* errors.**

6.3.2 *Size Invisible variant*

This section presents the results obtained with the Normal *Size Invisible* variant (5% size increase) and the Improved *Size Invisible* variant (optimum size increase).

The latter obtained better results.

Normal Size Invisible variant. Just like the *Shifted* variant, the *Size Invisible* variant only aims to correct *neighbor substitution* errors. As seen in Figure 8, *neighbor substitutions* were lower on the *Size Invisible* variant, suggesting that the variant was able to intervene correctly. As stated previously, this variant increases the height and width of the underlying area of the four most probable keys. Since this was the first study, the values were set by experimentation; the aim was to calculate the optimum size increase. Therefore, it was decided to increase 50% of the height of the key vertically, and 50% of the width of the key horizontally. Results show that this variant was able to intervene correctly 68.97% of the times, correcting 37.04% of *neighbor substitution* errors, when compared with the same input as if participants were typing on a QWERTY keyboard.

Improved Size Invisible variant. In order to find the optimum size increase, the necessary size increase to transform each bad touch point into a good one was calculated, and the necessary size increase that avoids transforming good touch points into bad ones. It was found that the best size increase is 37% of the height of the key vertically, and 21% of the width of the key horizontally, maintaining the rounded corners. **This allowed the *Size Invisible* variant to intervene correctly 93.10% of the times, correcting 62.96% of *neighbor substitution* errors.**

6.3.3 Color variant

When looking at Figure 7 one can verify that omission errors were lower on the *Color* variant. Omissions are most likely to occur when users miss a key or when their finger slips (they press one key and release on another, generating no output). It was verified that this type of error is most frequent on the space bar (47% of all the omissions are spaces, on the *QWERTY* condition). This occurred because the space bar is located at the bottom of the touchscreen, and sometimes users completely missed the touch area captured by the tablet, hitting its bevel instead. On the *Color* variant, when participants missed the space bar, they were able to detect it because the key remained highlighted, indicating that the key was not correctly pressed. As a matter of fact, space *omissions* were lowered to only 33% of all *omissions* on the *Color* variant.

6.3.4 Width variant

The *Width* variant was not so popular between users. Their performance on the *Width* variant regarding error rate was comparable to *QWERTY's*. This was achieved at the cost of reducing significantly the typing speed. Still, participants' comments were mostly negative, because the keys were always changing position which highly increased the cognitive effort to not commit mistakes.

6.3.5 Predict Words variant

Since this variant behaved mainly as the *QWERTY* keyboard, focus will mainly be on the different characteristics and types of errors that emerged from using this variant. Participants accepted 32.50% of the words they could have accepted. Regarding errors, there were 18 errors that were specific of the use of the *Predict Words* variant. Although participants were instructed at the beginning of the test that, after accepting a suggested word, a space would be automatically inserted, they forgot this several times, and inserted another one (11 times). Also, when trying to accept the suggested word at the top of the keyboard, two participants tapped on the "q" key instead. Therefore, the resulting word would contain the beginning of the word they had written and the letter "q" attached to the end. The second problem can easily be solved by increasing the size of the area that allows accepting the suggested words.

6.3.6 Touch Typing Patterns

Even though the *Shifted* variant was able to correct 48.65% of the substitution errors in the optimized version, the bottom-right touch pattern described by Nicolau [34] was not verified. The author hypothesized that the bottom-right pattern was related with hand dominance, since in the aforementioned study users only interacted with their right hand. In the present study, 18 users used both hands, while 2 users used only their right hand.

The tests showed an overall tendency to touch on the bottom-right side of the keys on the left side of the keyboard, and to touch on the bottom-left side of the keys on the right half of the keyboard, when using only the right hand (Figure 9 (a)) and when using both hands(Figure 9 (b)). This result is inline with Azenkot's et al. [4] results for the y-axis when interacting with index finger, one thumb or two thumbs, and for the x-axis when using the index finger.

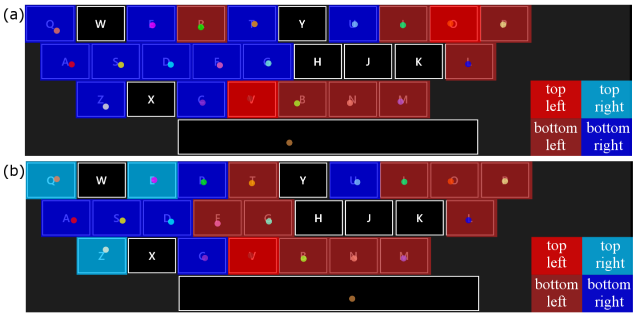

Fig. 9 Key deviation from the center of the key for (a) participants who interacted only with their right hand and (b) participants who interacted with both hands.

When analyzing the individual key deviation of the two participants who used only their right hand, it was verified that their pattern was contradictory; while the pattern of one participant was in accordance with the bottom-right pattern reported by Nicolau [34] (except for the "o" key), the other participant was not. It was not clarified whether the latter is a special case, or if indeed, every user has their own typing behavior, since only two participants interacted only with their right hand.

Regarding participants' who used both hands to type (Figure 9 (b)), there seems to be an overall tendency to touch on the bottom-right side of the keys on the left side of the keyboard, and touch on the bottom-left side of the keys on the right half of the keyboard. However, when analyzing the average deviation from the center of the key of each user, it was found that this deviation is strongly user-dependent.

Therefore, an adaptive model that constantly updates the center of each individual key seems to be the best solution to correct the *neighbor substitution* errors, without resorting to a predictive system.

7 Study with Older Adults

The results obtained in the baseline study described in the previous section, showed that the *Color*, *Shifted* and *Size Invisible* variants are the most promising variants. Therefore, one of the aims of this study is to verify if these variants are indeed advantageous for the older users. Since an attempt was made not to overload the participants with too many tasks, it was decided to test only the variants that involved visual changes. *Shifted* and *Size Invisible* variants will be evaluated in a simulation study described in section 10.

7.1 Participants

Twenty participants, 15 females and 5 males, took part in the user study. They were recruited from several local social institutions. Their age ranged from 61 to 92 years old, though the most prevalent age group was from 71 to 90 years old (75% of participants). Although all participants were right handed, 2 of them used only the left hand to type, 5 used both hands, and the rest used only their right hand to type.

None of the participants had severe visual impairments; all reported that they were able to read the screen content without difficulties. Users' capabilities regarding task-specific tremor were also assessed, by asking them to draw a Archimedes Spiral with each hand without leaning hand or arm on the table [5]. None of the participants presented accentuated hand tremor. No participant had used a touchscreen device before. Regarding QWERTY familiarity, 3 participants had used typewriters in the past, 9 had used personal computers, 5 had used both while 3 were completely inexperienced.

7.2 Procedure

The procedure was similar to the baseline study procedure described in section 5.2. The differences were: on the training phase, instead of allowing each user to experiment each keyboard variant during 2 minutes, each participant was allowed to type two sentences with each variant. If by the end of the two trial sentences participants did not fully understand the keyboard variant, we would let them type another sentence.

Also, in the end, after asking participants to answer a survey with demographic data and a satisfaction questionnaire, the users' capabilities regarding task-specific tremor were also assessed, by asking them to draw a spiral with each hand without leaning hand or arm on the table [5]. The whole process took approximately 1 hour per participant.

7.3 Apparatus

The apparatus is the same used in the baseline study described in section 5.3.

7.4 Dependent Measures

Performance during the text-entry task was measured by several quantitative variables: *Words Per Minute*

(*WPM*), *Minimum String Distance* (*MSD*) error rate, and character-level errors (substitutions - incorrect characters, insertions - added characters, and omissions - omitted characters) [29]. Qualitative measures were also gathered in the end of the experiment by debriefing each participant. One tremor-related measure was also gathered from each participant following the text-entry task: Archimedes Spiral test (action tremor) [5].

7.5 Design and Analysis

A within-subjects design where each user evaluated all conditions. For each keyboard condition each user entered 5 sentences (1 practice + 4 test), resulting in a total of 15 sentences per user. In sum the study design consisted of: 20 users × 5 sentences × 3 keyboards.

Shapiro-Wilkinson tests of the observed values for *WPM*, *MSD* error rate, types of errors and tremor measures were performed to assess if they were normally distributed. If they were, parametric statistical tests, such as repeated measures ANOVA, t-test, and Pearson correlations were applied. On the other hand, if measures were not normally distributed, non-parametric tests were used: Friedman, Wilcoxon, and Spearman correlations. Bonferroni corrections were used for post-hoc tests.

8 Elderly Study Results

In this section each user's tremor profile is described and characterized and related with text-entry performance. Moreover, input speed and accuracy for the three conditions (*QWERTY* keyboard, *Color* and *Predict Words*) were analyzed, focusing on type of errors and main causes.

8.1 Tremor Profile

Regarding tremor, task-specific tremor (a type of action tremor) was measured in both hands, using the Archimedes spiral test [5]. Since it was not possible to find an expert who could classify the drawings, it was decided to ask three different observers to classify the drawings, by visually comparing the drawings performed by the participants with examples of spiral drawings from other study [34]. For instance, if to the eyes of the classifier, the drawn spiral was similar to a spiral classified as "slight" on Nicolau's [34] study, the same classification should be attributed by the classifier. While this is not the best way to assess user tremor, we are confident that by having different classifiers corroborating each other's scores we have reached trustworthy results.

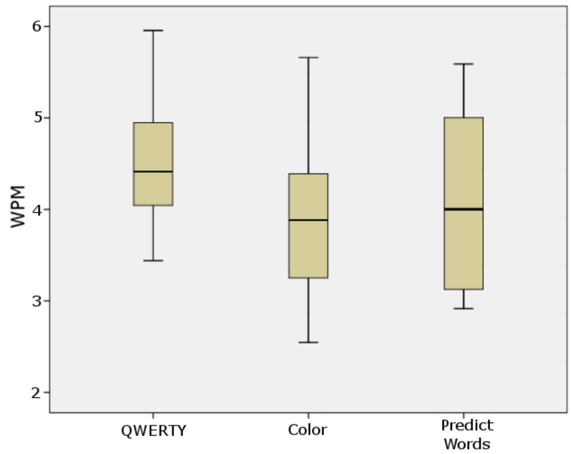

Fig. 10 Typing speed for each variant without outliers.

Classifications could be one of five: *Absent* (0), *Slight* (1), *Moderate* (2), *Severe* (3) and *Marked* (4). Following the classification, a *Cronbach's alpha* test was carried out to verify if the scores from the different classifiers were consistent. The *Cronbach's alpha* for the right and left hand is 0.890 and 0.901, respectively, which indicates a high level of internal consistency (the highest value of internal consistency is 1.0). To obtain the final score the average of the three observations was calculated. For the right-hand drawings, 45% of the participants had a score in the [0, 1[interval, 35% in the [1, 2[, 20% in the [2, 3[, and 0% in both [3, 4[and 4 ([4]). Regarding the left hand drawings, 5% of the participants had a score in the [0, 1[interval, 60% in the [1, 2[, 30% in the [2, 3[, 5% in [3, 4[and 0% in the 4 ([4]).

8.2 Input speed

To assess typing speed, the Words Per Minute (WPM) [31] text input measure was calculated as described in section 6.1 (equation 1).

Figure 10 illustrates WPM for each variant without outliers. Outliers were found using the *labeling rule* [19]. They were removed to perform the ANOVA analysis. As expected, a positive correlation between input rate, QWERTY experience and number of hands used to type was found. Furthermore, in general, correlations between input rate and task-specific tremor for both hands were not found. This means that QWERTY experience and number of hands used can explain input rate, while task-specific tremor can not.

A repeated measures ANOVA test revealed significant differences between keyboard variants on text-entry speed (F(2,30)=3.84, p<.033). Bonferroni post-hoc tests showed significant differences between QWERTY and *Color* variant, meaning that users type significantly slower with the latter. This result contradicted the hypothesis that inexperienced users, who are not acquainted with the QWERTY layout, would benefit from the *Color* variant. It is thus argued that the main reason for the lower input rate in the *Color* variant is because the highlighting of the keys was distracting. However, no user reported this. It was also noted that, in some cases, despite the correct letter being the only one highlighted by the *Color* variant, some older adults took a long time to find that letter on the keyboard. This means that some older adults were not paying enough attention to the highlighted keys, excluding them from the benefits of letter suggestion.

Regarding the *Predict Words* variant there was no significant difference when compared with the QWERTY keyboard. Still, one must take into account that only 7 of the 20 participants accepted at least one suggested word from the list during evaluation; the remaining 13 participants used the *Predict Words* variant as a normal QWERTY keyboard. Still, it was not possible to find a correlation between text-entry speed on *Predict Words* variant and interaction methodology, i.e., if the participant accepted suggested words or typed as a normal QWERTY keyboard.

8.3 Quality of transcribed sentences

The quality of the transcribed sentences was measured using the Minimum String Distance (MSD), calculated as described in subsection 6.2 (equation 2).

Figure 11 illustrates the MSD by variant without outliers. Outliers were found using the *labeling rule* [19]. They were removed to perform the ANOVA analysis, which did not reveal significant differences between keyboard variants (F(2,32)=1.044, p=.364). In contrast to the results obtained on input speed, no correlation was found between quality of transcribed sentences and previous experience with QWERTY keyboards, number of hands used and task-specific tremor. This means that neither of these dimensions can explain error rate.

It was expected both *Color* and *Predict Words* variants to outperform the QWERTY keyboard regarding MSD. Although it is not clear why the *Color* variant did not outperform the QWERTY keyboard, several situations occurred that are important to report. For instance, one

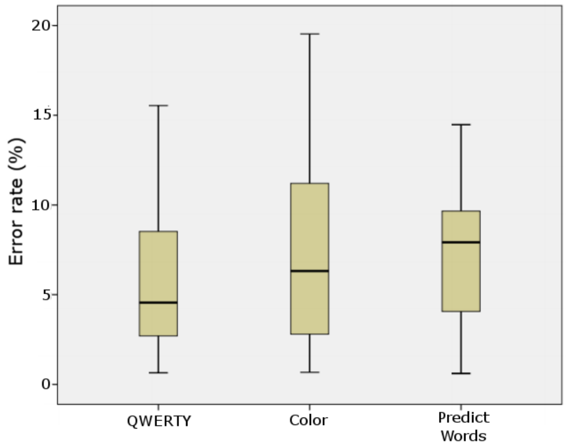

Fig. 11 Error rate for each variant without outliers.

participant ended up typing a word similar to the expected one because the *Color* variant suggested it, and the user tapped the suggested letters without thinking too much. This is an issue related with the prediction algorithm. Since the system does not always suggest the right letter, the user still has to pay attention to the suggested letters. Also related to this, is a trust issue. Do users think that the system is correct most of the times? Are they completely capable of ignoring the suggested letters and select one of their own? Sometimes it seemed as if participants were afraid of tapping a certain key if the system was not suggesting it, especially after tapping a sequence of keys correctly suggested by the system. The performance of the *Color* variant was also affected by the fact that older users committed many errors. This means that the *Color* variant cannot make good suggestions, because once there is an error in the current word, the system is not able to correctly predict the sequence of letters intended by the user.

The *Predict Words* variant also had a MSD similar to QWERTY, mainly because most participants (13) did not accept any suggestion. From the remaining 7, only 3 accepted a high number of suggested words (between 9 and 11 suggestions). From these, 2 participants had worst results in the *Predict Words* variant when compared with QWERTY. This happened because sometimes, when accepting a suggested word (located at the top of the keyboard), users tapped below the intended area, selecting a key from the top row of the keyboard instead. Another common error is to tap the space bar after accepting a suggested word. This counts as an insertion error because after accepting the suggested word a space is automatically inserted. Therefore, the use of the *Predict Words* backfired because participants ended up making mistakes they would not make in other situations.

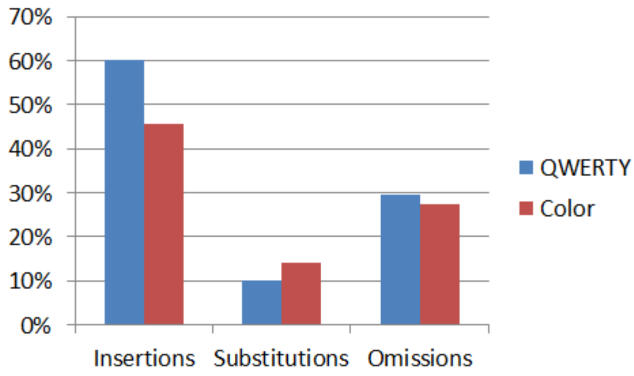

Fig. 12 Contribution of each type of error for the total amount of errors.

8.4 Typing errors

The types of input errors were classified using MacKenzie et al.'s categorization [29] (substitutions – incorrect characters, insertions – added characters, and omissions – omitted characters). In some cases, a more specific categorization is assigned to errors, with a clear indication whenever this occurs.

In Figure 12, one can verify that insertion errors are the most common type of error committed by older adults. This kind of errors is unevenly distributed through all the participants. For instance, in the QWERTY keyboard, participants #2 and #17 are responsible for 62% (139 errors) of all insertion errors. Omissions were the second most common error type among older adults, followed by substitutions. The *Predict Words* variant was not analyzed thoroughly regarding typing errors, because most of the participants used it as a QWERTY keyboard.

8.4.1 Insertion errors

When analyzing the touch data from the QWERTY keyboard, it was found that most of insertions (65.18% of all insertions) occurred due to multiple interleaved points of contact, i.e., the second point of contact occurs before the first one is released (from now on denominated *extra-finger* insertion). Since *extra-finger* insertions exist because the keyboard is multi-touch, it is relevant to assess if this kind of error is mostly committed by participants who used both hands to interact with the keyboard. However, no correlation was found between number of hands used and *extra-finger insertions*. In fact, it can be verified that from the total 5 participants that used both hands to interact with the keyboard, only one committed this type of insertion. This participant only accounted for 4.79% (7 errors) of *extra-finger insertion* errors.

It was found that most of extra-finger insertions were committed by participants #2 and #17, which interacted with just one hand. Participant #2 interacted with the index finger of its left hand (intentionally), and with the thumb of the same hand (unintentionally). This means that, every time the user would tap a key with their index finger, they would also tap the space bar key, or a key in the Z's row, unintentionally. This is the main reason why this user performed so many *extra-finger insertions*. Participant #17 was a completely different case; in fact, as strange as it may sound, this participant only used their index finger to interact with the device, and still committed *extra-finger insertions*. During the test, it was thought the user was committing *double insertion* errors, since the participant was only interacting with one finger, and the output would always double the pressed keys; if the participant pressed the "a" key, a double "a" would appear in the output. The analysis of the logs following the test revealed that something completely different had happened. While the actual cause is still unclear, the logs showed that at least two contact areas were recognized, since touches were interleaved; meaning that, a second press was recognized, before releasing the first press. The only way to correct this kind of error is to disable interleaved touches, i.e., transforming the multi-touch keyboard into a single touch keyboard. This will be further discussed in Section 10 (Simulation Study).

The second most common type of insertion error, was cognitive insertions, accounting for 19.20% (43 errors) of total insertions. This kind of error occurs when the user inserts a character other than the expected. When further analyzing this kind of error, it came out that it had several causes. Some participants repeated syllables, others seemed to misread the word and then obviously typed a wrong word or character. Some participants inserted more than one space between words. It seemed as if participants inserted a space character after typing a word, read the next word from the sentence to transcribe, and re-inserted another space, probably because they were not sure if they had already done it.

The third most common type of insertion error, was double insertions, accounting for 12.05% (27 errors) of total insertions. This kind of error occurs when the user inserts a repeated character with a reduced time interval between the release of the first key and the press of the second key. Analysis of the logs showed that, even though the second inserted character was classified as being the *double insertion* error ("aa"), the error itself can be either of the two inserted characters. The user can (1) insert the first character and accidentally insert a second one; (2) before inserting the intentional character, accidentally touch the keyboard and insert a character unintentionally before inserting the desired one. This classification, although similar to the *accidental insertion* error, is different because the *accidental insertion* error is an isolated error, with only one interaction.

The least common *insertion* error is the *accidental insertion* error, accounting for 3.57% (8 errors) of *insertion* errors. This kind of error occurs when the user presses a key accidentally and is characterized by a reduced time interval between the press and release of the key.

The proportion of each type of insertion error is similar across variants. This was expected since, in general, the *Color* variant does not aim to correct insertion errors. It could somehow avoid cognitive insertions since it draws the attention of the user to the four most probable keys, but no significant difference was found between variants.

8.4.2 Substitution errors

Two types of substitution errors are considered: neighbor and cognitive substitution errors. The former occurs when users touch an adjacent key instead of touching the intended key. The latter occurs when users insert an incorrect character instead of the expected one, which most of the times is related with similar representations of the letter (e.g: p and q).

After analyzing the touch data, it was found that touch points were skewed to the bottom and slightly to the right for users who only used their right hand to type (Figure 13 (a)). Other studies have also reported this result [34, 18]. It was also found that the horizontal direction of the shift was related to the hand being used to type, as hypothesized by Nicolau [34], although it was not possible to verify the pattern for users who only used their left hand to type (Figure 13 (b)). Still, it has to be taken into account that these data might not be enough, since there were only two participants typing only with their left hand. Regarding users that used both hands to type (Figure 13 (c)), it was verified that the left side of the keyboard has its touch points skewed towards the bottom-left, while the right side of the keyboard has its touch points skewed towards the bottom-right. These results were obtained for both QWERTY and *Color* variant, which means that highlighting keys does not influence aiming.

Fig. 13 New center of keys for participants who only used their (a) right, (b) left and (c) both hands to interact in the QWERTY condition.

It was also verified that shifts have a bigger vertical deviation (Mean = 13px; SD = 11.5px) when compared to the horizontal deviation (Mean=4.5px; SD=14.7px), for all typing methods. Moreover, it was also found that, the vertical shift increases gradually, from the top to the bottom row (average vertical deviations: row1 = 11px; row2 = 14px; row3 = 18px; row4 = 20px). For users that typed only with their right hand, something similar seems to happen with the horizontal shift; that is, the horizontal shift seems to increase gradually as we move from the left to the right side of the keyboard (although this pattern is not as strong as the vertical). For users that typed with both hands there seems to be an increasing shift to the right, from the middle of the keyboard to the right edge of the keyboard, and an increasing shift to the left, from the middle of the keyboard to the left edge of the keyboard.

In the QWERTY keyboard, users committed 29 neighbor substitution errors and 9 cognitive substitution errors. However, users significantly committed more cognitive substitution errors on *Color* variant (Z=-1.845, p=.065); they committed 30 neighbor and 30 cognitive substitution errors. Detailed analysis, verified that in 65.5% of cognitive substitutions the user inserted a character that was highlighted by the *Color* variant. And, in the remaining 34.5%, the expected key was highlighted, though it did not prevent the user from inserting an erroneous character which was not highlighted. It was also noticed that in 20.7% of the cognitive substitution errors both expected and inserted keys were highlighted. Despite acknowledging this result, no logical reason to justify it could be identified.

8.4.3 Omission errors

Omissions were subdivided into 3 sub-categories: failed (the user presses an empty space instead of the intended key – only applicable to the keys at the edges), slide (the press action was in a different key when compared to the release action) and cognitive (user forgets to insert an expected character) omissions.

Omission errors had approximately the same proportion across variants, being the cognitive the most frequent (52% - 117 errors), followed by slide (27% - 62 errors) and failed (21% - 47 errors) omissions. Moreover, it was found that forgetting to enter a blank space between words was a common issue among older adults (44.8% of the total cognitive omissions), most likely due to a lack of practice in typing on computers [34]. Since the *Color* variant highlights the next most probable keys, it would be expected that, when correct, the suggestion could minimize omissions. Still, cognitive omissions were as frequent as on the QWERTY keyboard. When further analyzing this type of error, it appeared that during 52.5% of the cognitive omission errors, the expected key was highlighted. This in turn means that in more than half of the cases, the *Color* variant was helping the participant, and still they forgot to type the intended character.

Slide omissions differ from the previous, because the user presents the intention to type a character, but fails in the execution. It occurs when the user presses and lifts its finger on different keys and therefore no output is generated. Furthermore, slide omissions were classified in three subclasses: (1) correct land-on, characterized by the finger landing on the intended key, and then sliding to another key; (2) correct lift-off, characterized by the finger landing on a neighbor key, and then sliding to the intended key; (3) and accidental slide, on which the user has no intention to tap either of the keys. The first type accounted for 36.4% of the slide omission errors, the second for 57.6% and the third for 6%. It was found that all the errors classified as correct land-on, ended always in a key below the intended one; that is, the slide was always performed from the top to the bottom. Contrary to this, 89.5% of the errors classified as correct lift-off, ended in a key above the pressed one. On the remaining cases the slide was performed from the right to the left. This means that, **when taking into account only the vertical slides, if a user performs a slide starting at a key in a given row, and lifts its finger on a key in the row above, we are 100% sure that the user intended to tap the key in the row above. When the opposite happens, in 85.7% of the times, the user also wants the key in the row above (the key were the user landed its finger).** In the remaining 14.3% times, the intentions of the user cannot be clarified, since the slide was completely accidental. This pattern was also verified for the *Color* variant. It was hypothesized that when the user slides down, he/she is moving their hand towards the rest position, below the tablet. When the movement is in the opposite direction, it is a corrective movement, because the user adjusted the touch position in a contrary motion to the resting position. To the authors' knowledge, this pattern has not been reported by any other study, presenting an opportunity for improvement of virtual keyboards.

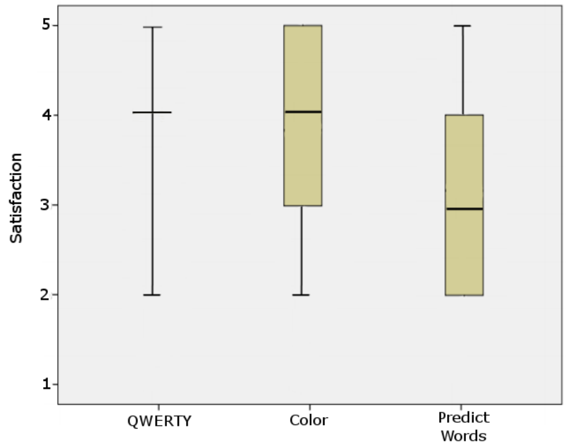

Fig. 14 User satisfaction for each variant (1 represents "completely unsatisfied", while 5 represents "completely satisfied").

8.5 User satisfaction

At the end of the user study participants were debriefed and asked about their preferred keyboard. Comments were also collected during and after the test regarding the participants' opinion about the several keyboard variants. When asked about their satisfaction (5-point Likert scale) using each of the keyboard variants, participants gave a higher rate to the QWERTY keyboard, closely followed by the *Color* variant and finally by the *Predict Words* variant, as we can see in Figure 14. Still, 6 participants rated the *Color* variant with the highest score (5), while only 1 participant rated each of the remaining keyboards with the highest score.

More "complex" questions regarding ease of use, utility, cognitive effort and easiness to locate the intended key were noted. However, during the study it was found that participants did not understand the difference between them; they reported the questions were all the same. Therefore, it was decided to keep things simple and ask only about their satisfaction towards the keyboard.

Some users also reported that the tablet was too sensitive, referring to the fact that it is easy to make typing mistakes when interacting with touch devices. A participant reported that it was faster to type with the *Color* variant, referring to a specific case when the system was able to always suggest the right letter. Some participants reported that the *Color* variant was really helpful but, in order to take full advantage of it, it was necessary to pay attention. When participants were asked about why they did not use the suggestions presented in the *Predict Words* variant, most said it was a feature too complex and they would need more practice in order to correctly use it.

9 New Keyboard Alternatives

The results obtained in the elderly study described in section 8 reveal more about their typing behavior, and therefore created four new variants that aim to correct specific types of errors. These new variants will be used in a simulation study (section 10) along with the *Shifted* and *Size Invisible* variants. In this section these new variants will be described.

9.1 Single Touch variant

The *Single Touch* variant behaves exactly as the baseline keyboard, except it is single touch; that is, instead of allowing more than one touch point at a time, it only allows one. So, if a user presses a second key before releasing the first one, the second touch interaction will be discarded, thus only inserting the first character. Authors that performed studies focusing on touch devices have reported that in general users prefer to interact mainly with one hand and only one finger [26, 49]. This variant will help us clarify if single touch is indeed a more accessible choice for older adults, or if multi-touch is more appropriate.

9.2 Intra-key Timed variant

This variant emerged mainly to correct *accidental insertion* errors. This kind of error is characterized by a reduced time interval between the press and release of a key (intra-key). In order to correct this kind of error, this variant assumes the existence of a threshold that indicates which interactions are considered *accidental insertions* and which are not. Since it is not possible to know what the best threshold for all users is, or if different users will require different thresholds, several simulations will be performed in order to find the best threshold that maximizes the correction of *accidental insertion* errors and minimizes the creation of new errors. A similar approach has been executed by Nicolau [34]. This variant is built upon *Single Touch* variant; i.e., instead of allowing more than one touch point at a time, it only allows one. Regarding all other aspects, this variant behaves just like the *baseline QWERTY* keyboard.

9.3 Inter-key Timed variant

This variant emerged mainly to correct *double insertion* errors. This kind of error is characterized by the insertion of a second character with a reduced time interval between the release of the first key, and the press of the second key. Just like the previous variant, in order to correct this kind of error, this variant assumes the existence of a threshold that indicates which interactions are considered *double insertions* and which are not. Since it is impossible to know what the best threshold for all users is, or if different users will require different thresholds, several simulations will be performed in order to find the best threshold that maximizes the correction of *double insertion* errors and minimizes the creation of new ones. A similar approach has been executed by Nicolau [34]. Just like the previous variant, this variant is built upon *Single Touch* variant; i.e., instead of allowing more than one touch point at a time, it only allows one. Regarding all other aspects, this variant behaves just like the *baseline QWERTY* keyboard.

9.4 Combined Timed variant

The *Combined Timed* variant is the combination of *Intra-key* and *Inter-key Timed* variants. Therefore, its main goal is to correct *accidental* and *double insertion* errors. Both variants keep track of time in order to operate. However, the variants are independent of each other, which mean that each one will have its threshold and operate independently.

10 Simulation Study

In this simulation study the performance of the *Shifted*, *Size Invisible*, *Single Touch* and *Intra-key*, *Inter-key* and *Combined Timed* variants is evaluated. In order to do this the simulation is fed with the log data obtained from the *QWERTY* keyboard condition. The log data contains information about the sentence to transcribe, the pressed and released point of each interaction and the time elapsed between each interaction.

Although a simulation is not as good as running real user tests, it can offer some insights on the impact these variants would have if users were typing on them. It is important to note that simulations were only performed with variants that were visually similar to the QWERTY keyboard; the differences were only regarding the processing of the touch inputs. Therefore, it can be argued that these results are as valid as the results from the user study. Still, different results could be obtained if these variants were tested directly with users, because users' expectations of how the system would respond could change. Users could, for instance, commit less errors or even adapt their interaction technique to the point of committing similar errors (e.g.: since the *Shift* variant compensates the fact that users touch on the bottom-right of the buttons when using the right hand, users could adapt their technique and touch further down and to the right of the button). However, since there is no visual difference to the baseline QWERTY variant, it is argued that, those changes, if they existed, would be small. Thus, the simulated study will be important to study the validity of the possible solutions (even if the most promising ones should be directly evaluated with users in the future).

10.1 Shifted variant

Since participants mainly used their right hand to interact, and the researchers were aiming at a solution that would fit all users, it was decided to shift the touch points towards the top and the left, in order to compensate the global bottom-right skewness verified on the QWERTY data.

In order to find the shift that minimizes the MSD, several simulations were performed. These simulations combined shifting the touch points to the top between 0% and 15% of the height of the key (increments of 1%), and to the left between 0% and 15% of the width of the key (increments of 1%). The best result was obtained with a horizontal shift of 2% (or 4-6%) and a vertical shift of 10%. Figure 15 shows a *Heatmap* with the results for all the evaluated shifts. This result proves that shifts are indeed more intense on the vertical direction (y axis) than on the horizontal direction (x axis), because the vertical compensation is higher than the horizontal one. A Wilcoxon test revealed that the *Shifted* variant significantly improved the MSD in comparison

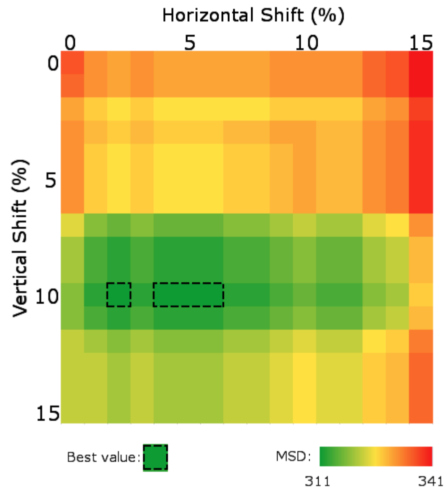

Fig. 15 Heatmap for the Shifted variant.

to QWERTY (Z=-2.746, p=.006).

Considering the generic solution described, the *Shifted* variant corrected 58.6% of the neighbor substitution, 100% of the failed omission and 39.4% of the slide omission errors. Indeed, there is a significant difference between QWERTY's and *Shifted*'s neighbor substitution errors (Z=-2.456; p=.014), as well as for failed omissions (Z=-2.207; p=.027) and slide omissions (Z=-2.232; p=.026).

However, the simulation ran is more suited for users who only use their right hand to type. It was also found that the deviation of the touch points to the key's center is user-dependent, even for users that used the same hand posture. Therefore, an adaptive model that constantly updates the center of each individual key would be the best solution. One must also take into account that the same user can present different touch typing patterns, depending on the hand posture used [50]. Therefore, an adaptive model that recognizes different hand postures will even perform better.

10.2 Size Invisible variant

The *Size Invisible* variant uses the *letter prediction* algorithm described in Section 3 to increase the underlying area of the most likely keys. In order to find the optimal size increase that fits all users, several simulations were performed. These simulations combined increasing the height of the key between 5% and 95% of its height (increments of 5%), and the width of the key between 5% and 95% of its width. The best result was obtained with an increased size of 5% for the width and height of the key. Figure 16 shows a *heatmap* with the results

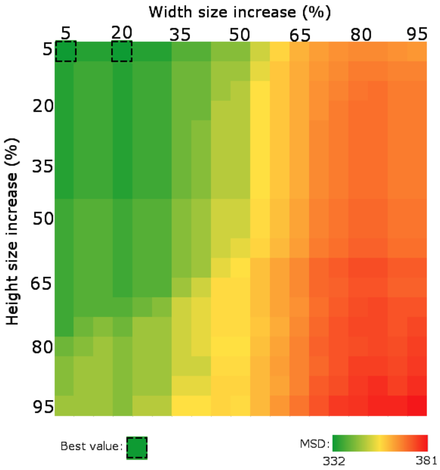

Fig. 16 Heatmap for the Size Invisible variant.

for all the evaluated sizes. A Wilcoxon test revealed significant improvements in the MSD between QWERTY and *Size Invisible* variant (Z=-2.073, p=.038).

The *Size Invisible* variant was not able to perform as well as the *Shifted* variant. This happened because sometimes the prediction system was not able to operate for three reasons: (1) the user had already typed a mistake, thus the prediction system cannot match words; (2) the error was at the beginning of a word and the prediction only works after inserting the first letter; (3) because the key increase was not big enough, but if it was bigger, it would create more errors than corrections. The generic solution (H=5%; V=5%) corrected 17.2% of the neighbor substitution, 21.1% of the failed omission and 12.1% of the slide omission errors. Even though it did not perform as well as the *Shifted* variant, it was able to reduce significantly neighbor substitutions (Z=-2.236; p=.025) and failed omissions (Z=-2.000; p=.046). Results also suggest that the *Shifted* variant is able to reduce slide omissions (Z=-1.633; p=.102) when compared to the QWERTY keyboard (89.8% of confidence level).

10.3 Single Touch variant

The *Single Touch* variant emerged mainly to correct extra-finger insertion errors. Extra-finger insertions are characterized by interleaved presses/releases of different touch points, i.e., when the press of a second key occurs without the release of the first key. This kind of error occurs when (1) users interact with more than one finger unintentionally and when (2) users interact with two fingers intentionally, but do not realize that it means that both touches are recognized, leading to an extra character.

When downgrading the virtual keyboard from multi-touch to single touch, there were two possible outcomes regarding the *extra-finger insertion* errors. For instance, assume the user wants to tap the "*E*" key, but accidentally also touches the space bar. Depending on the order of the presses, a *good* or a *bad* outcome can occur. If the user presses the "*E*" key first, and before releasing it, presses the space bar, the space bar is omitted (*good* outcome). Contrarily, if the user presses the space bar first, and before releasing it, presses the "*E*" key, the "*E*" character is omitted (*bad* outcome).

The results of the simulation revealed that the *Single Touch* variant performed very well for some participants, but also hindered others slightly. It is hypothesized that this explains why the results were not statistically significant. Results suggest that the *Single Touch* variant is able to reduce slide omissions (Z=-1.223, p=.221) when compared to the QWERTY keyboard (77.9% of confidence level).
The *Single Touch* simulation was able to correct 63.84% of the *insertion* errors, 38.46% of the *empty* errors and introduced 26 new errors. The *Single Touch* variant main goal is to deal with *extra-finger insertions*. When taking into account only this type of error, this variant was able to correct 78.77% of them (counting already with the 26 new errors that emerged).

A Wilcoxon test between *QWERTY's* and *Single Touch's extra-finger insertion* errors, revealed significant differences (Z=-2.670; p=0.008) when not taking into account the new errors created by this variant. When taking the new errors into account, the results also suggest that the variant was able to reduce *extra-finger insertions* (Z=-1.584; p=0.113; 88.7% of confidence level).

10.4 Intra-key Timed variant

This variant emerged mainly to correct *accidental insertions*. This kind of error is characterized by a reduced time interval between the press and release of a key. The problem is that this time interval is highly user-dependent, and even within a specific user, it can vary a lot. Therefore, several simulations were performed in order to find the right threshold for each user, which would allow maximizing the rejection of *accidental insertions*, and minimizing the rejection of false positives; i.e., correct characters that happen to have a time interval below the threshold. To do this, a varied threshold between 25ms and 750ms with increments of 25ms.

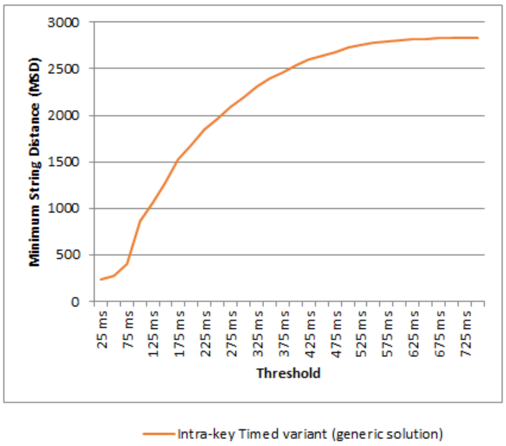

Fig. 17 Minimum String Distance (MSD) for each of the simulated thresholds for the Intra-key Timed variant.

Figure 17 shows that the threshold that yields better results (MSD = 240) for the generic *Intra-key* solution is 25ms. This means that the generic solution is not able to reduce MSD. In fact, the best threshold, worsened the MSD in 5.3% when compared with the *Single Touch* variant.

Even when considering a solution adapted to each user (best threshold for each user), a Wilcoxon test did not reveal significant differences between *Intra-key* and QWERTY, and *Intra-key* and *Single Touch*, regarding MSD.

Regarding typing errors, the adaptive solution of this variant was able to correct 75% of the *accidental insertion* errors and 55.56% of the *double insertion* errors. Before looking at the results, *double insertions* were not expected to be corrected by the *Intra-key* variant. In order to better understand why *double insertions* were corrected by the *Intra-key* variant, this kind of error was analyzed in detail. In terms of output, a word that contains a *double insertion* error will have an extra character repeated (e.g.: "woord"). While classifying the errors, the repeated character was always the one considered an error. For instance, in one of the sentences, one of the participants typed "aaos" instead of "aos". The second "a" was considered a *double insertion* error because it was a repeated character and because the time elapsed between the release of the first "a" and the press of the second "a" was 64ms. The results of the *Intra-key* show that the first "a" of "aaos" was omitted. This indicated that there are different patterns when committing *double insertions*: (1) the user types the first "a" intentionally, and then accidentally types a second "a"; (2) the user intends to tap the "a" key, but first taps it accidentally, and then intentionally; (3) finally, a mix of both, where, although there is the intention to tap the "a" key, both touches are fast, therefore could be considered *accidental*. Results show that 55.56% of the times, *double insertions* were caused by an accidental first touch.

This variant also created 32 new errors. This is the main reason why this variant did not perform well. It is related with the fact that finding the optimum threshold is not an easy task; most of the times it is impossible to set a threshold that rejects all the accidental insertion errors, without rejecting correct characters.

It is important to note that these results are different to the ones obtained by Matero et al. [32]. Some of the users have intentional touches as low as 26ms, and go up to 250ms. On the other hand, Matero et al. [32] reported that the majority of intentional samples of their users have a touch time between 70ms and 400ms.

10.5 Inter-key Timed variant

The *Inter-key Timed* variant emerged mainly to correct *double insertions*. This kind of error is characterized by the insertion of a second character with a reduced time interval between the release of the first key, and the press of the second key. The problem is that, this time interval is highly user-dependent, and even within a specific user, it can vary a lot. Therefore, several simulations were performed for each user in order to find the right threshold for each user, which would allow maximizing the rejection of *double insertions*, and minimizing the rejection of false positives; i.e., correct characters that happen to have a time interval below the threshold. To do this, a varied threshold between 25ms and 1000ms (1 second) with increments of 25ms was adopted.

It was found that, considering a generic solution for all users, the inter-key threshold that yields better results (MSD = 215) is 125ms (Figure 18). With this threshold, the MSD improves 5.7% when compared to the *Single Touch* variant.

When using an adaptive solution, i.e., the best threshold for each user, the MSD improved in 11.4% when compared to the *Single Touch* variant. Indeed, significant differences in MSD were found between the *Inter-key Timed* adaptive solution and the QWERTY keyboard($Z = -2.373$, $p = .018$) and the*Single Touch* variant ($Z = -3.006$, $p = .003$).

Regarding typing errors, the adaptive solution of this variant was able to correct 85.19% of the double in-

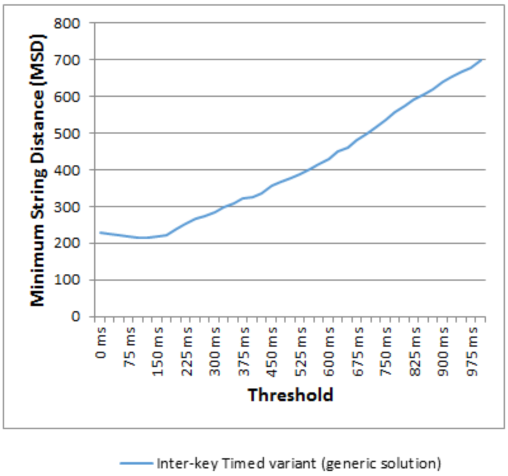

Fig. 18 Minimum String Distance (MSD) for each of the simulated thresholds for the Inter-key Timed variant.

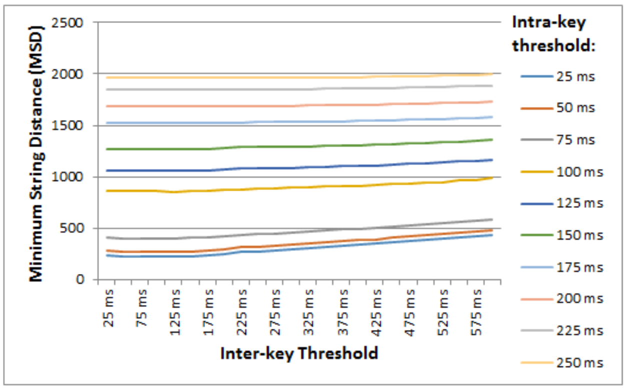

Fig. 19 Minimum String Distance (MSD) for combined the thresholds for the Combined Timed variant.

sertions and also created 8 new errors. This happened because finding the optimum threshold is not an easy task; most of the times it is impossible to set a threshold that rejects all the double insertions, without rejecting correct characters. Still, a Wilcoxon test between *QWERTY's* and *Inter-key's double insertion* errors, revealed significant differences (Z = -3.372, p = 0.001), when not taking into account the new errors created by this variant. A Wilcoxon test also showed that this variant had significantly fewer typing errors than QWERTY (Z=-2.737; p=.006) and Single Touch (Z=-2.874; p=.004).

10.6 Combined Timed variant

The *Combined Timed* variant is the combination of *Intra-key* and *Inter-key Timed* variants. Therefore, it will be able to correct *accidental* and *double insertion* errors, mainly. Both parts keep track of time in order to operate. However, the two parts are independent of each other, meaning that each one will have its threshold and operate independently. Still, it is important to acknowledge that, since the *Intra-key* part operates on the same interaction (i.e., compares the time elapsed between the press and the release of the same interaction), it can prevent *Inter-key* part from operating on certain cases. This means that, in a case where both *Intra and Inter-key Timed* variants would operate on their own, when combined, only the *Intra-key Timed* variant will operate.

This is the reason why the *Combined Timed* variant was not able to perform better than *Inter-key* in general, and regarding *double insertions* in particular. Because the *double insertions* corrected by the *Intra-key* variant independently, were also corrected by the *Inter-key* variant independently. Therefore, when combined, only one of the parts corrected those errors. Thus the *Combined* variant corrected exactly 85.19% of the *double insertions*, just as *Inter-key*. Regarding *accidental insertions*, it corrected the same as *Intra-key's* (75%). In the end, the Combined variant performed worse than Inter-key as it created more new errors (35).

Similar to the *Intra-key* and *Inter-key* variants, the adaptive solution is better than the generic solution. The best generic thresholds for the *Combined* variant are the same found for each of the independent variants; that is, the thresholds that yield better results (MSD = 226) for the *Combined* variant are 125ms and 25ms, for the *Intra-key* and *Inter-key* threshold, respectively (Figure 19).

A Wilcoxon test between *QWERTY's* and *Combined's accidental* and double insertion errors (the two types of error that could be solved by *Combined* variant), revealed significant differences (Z=-3.342; p=0.001), when not taking into account the new errors created by this variant. When taking the new errors into account significant differences were also found (Z=-1.947; p=0.052) at a lower confidence level (94.8%). On the first case it means that *Combined* variant was able to significantly reduce *accidental* and *double insertions*, while on the second one, it means that *Combined* variant actually created more new errors than corrected *accidental* and *double insertions*.

11 Design Implications

From the results of the baseline study, it can be derived the following design implications for text-entry solutions in tablet devices for younger adults:

Allow personalization. Since touch typing behavior is completely user-dependent, the optimal approach is to have an adaptive model that constantly updates the center of each key.

Use a language model to increase the underlying area of the most probable keys. Since no touch typing pattern emerged, a model that increases the underlying area of the four most probable keys will help to decrease neighbor substitution errors. The model will be able to ensure that, independently of the position of the touch point (top left, top right, bottom left or bottom right), it will be assigned to the intended key, as long as the intended key is highlighted.

Include an option that allows the user to enable/disable the highlighting of the four most probable keys (color). If the user is more concerned with accuracy than with typing speed, better results will be obtained with this option enabled. Otherwise, better results will be obtained with this option disabled.

From the results of the study with the older adults, the following design implications for text-entry solutions in tablet devices for older adults can be derived:
Keep visual changes to a minimum. As verified in the younger and older adults user studies, visual changes that aim to focus the user attention on the most probable keys, have a negative impact in text-input speed. This result is comparable to the results obtained by Henze et al. [18], on which users were slower when visual feedback regarding the tapped area was shown. Also, unlike what was found for younger users, the *Color* variant had twice the *cognitive substitution* errors, when compared with the *QWERTY* condition. When further analyzing this kind of error, it was found that 44.83% of the times, the substitution character inserted by the user was highlighted, while the expected one was not. Although one cannot be absolute about this, it is possible that users were influenced by the highlighting of the keys. Therefore, visual changes should only occur to give feedback about the pressed and released key.

Shift the touch points to the top and to the opposite side of the hand the user is using to type. As verified in the user study, different touch patterns emerge from different interaction techniques. For instance, users who only used their *right hand* to interact with the virtual keyboard had a tendency to touch on the bottom-right of targets (also verified by several authors [34, 17, 18]). This means that such users benefit with a top-left shift of their touch points to compensate the tendency. Conversely, users who only used their *left hand* benefit with a top-right shift of their touch points. Users who interacted with *both hands* will, hypothetically, benefit with a top-left shift of the touch points performed on the right side of the keyboard (keys: y, u, i, o, p, h, j, k, l, b, n and m), and a top-right shift of the touch points performed on the left side of the keyboard (keys: q, w, e, r, t, a, s, d, f, z, x, c and v). These results differ from the ones found for the younger users, which were more user-dependent.

When a vertical slide occurs between two keys of subsequent rows, assign the press and release to the key in the row above. It was verified in the user study that, when users perform a vertical slide from one row to a subsequent row (up or down), most of the times the user intends to select the key from the row above. This was verified across variants. When the slide was performed from the bottom to the top, the user intended to tap the key in the row above 100% of the times, for both the *QWERTY* and *Color* conditions. When the slide was performed from the top to the bottom, the user also intended to tap the key in the row above 85.71% and 100% of the times, for the *QWERTY* and *Color* conditions, respectively. In the remaining 14.29% times (*QWERTY* keyboard), it is not clear exactly what were the intentions of the user, since the slide was completely accidental. Evidence of this phenomenon for the younger adults has not been found.

Choose single touch over multi-touch. Older users are all different with different necessities and capabilities. If there is a need of including all types of older users, *Single Touch* is the right choice. The quality of the sentences of two of the most problematic participants (#2 and #17) increased drastically, with few new errors for other participants. Younger users do not benefit from this approach, since they are generally faster and are more likely to intertwine the release of a key and the press of another. This would exclude correct touch interactions.

Omit touch interactions that are below a certain threshold (between press and releases of different interactions). During the simulation study, it was verified that the *Inter-key* variant was able to increase the quality of the transcribed sentences, when compared with the *Single Touch* variant. This means that, having a keyboard that omits interactions based on the time elapsed between the release of a first key and the press of a second key can enhance older adults' accuracy. The threshold value must be defined carefully since this value is highly user-dependent. A similar design implication was proposed by Nicolau [34].

Allow personalization. Similarly to younger users, patterns were found that emerge from the different interaction methods (only *right hand*, only *left hand* or *both*) adopted by the participants. Still, when looking at each participant of the same group, one can verify that some of them have different typing behaviors, particularly hit point locations and inter-key time interval (*Inter-key* variant). Therefore an adaptive model that constantly updates the center of each individual key seems the best solution. A similar design implication was proposed by Nicolau [34].

12 Conclusion

The goal of this work was to better understand how older users interact with touch devices, in order to obtain the needed knowledge to design text entry solutions more appropriate for older adults.

To fulfill this goal, two different studies were conducted. On the first one, the baseline study, a traditional QWERTY (virtual) keyboard and five different variants (*Color*, *Width*, *Predict Words*, *Shifted* and *Size Invisible*) that were based on state-of-the-art solutions were evaluated. The goal of this study was to assess which were the most promising variants and also assess if our application was robust enough to be used in the elderly user test. On the subsequent study with the elderly the most promising variants (*Color*, *Predict Words*, *Shifted* and *Size Invisible*) were maintained and the least promising one (*Width* variant) was removed. The goal of this study was to test only the variants that had visual changes (*Color* and *Predict Words*) and perform a simulation of the remaining variants by using the input data of the QWERTY condition. Four new variants (*Single Touch* and *Intra-key*, *Inter-key* and *Combined Timed* variants) were also created based on Nicolau's [34] studies and the authors' own observations during the tests. Since these variants differed from the others only with respect to the processing of touch inputs, it was also possible to use them as simulations. The goal in this study was to better understand older users typing patterns and find the most fitting variant(s).

In the first study (**Baseline Study**) the goal was to assess which of the variants were most promising. The ***Color* variant** was able to reduce the error rate significantly, specially *cognitive omissions*. This was possible because on the *QWERTY* condition, most of the cognitive omissions were on the space bar. Users sometimes completely missed the touch area captured by the tablet, hitting its bevel instead. On the *Color* condition they were able to acknowledge it because the space bar remained highlighted. Still, this result was obtained at the cost of reducing input rate. Even though the ***Width* variant** is very similar to the *Color* variant (in concept), the *Width* variant was not able to reduce error rate significantly, and it even performed worse than the *Color* variant regarding input rate. The ***Predict Words* variant** performed even worse than the *Width* variant, regarding both error and input rates. Still, users were satisfied and found it easy to use. The ***Shifted*** and ***Size Invisible* variants** were able to reduce neighbor substitution errors. The *Size Invisible* variant outperformed the *Shifted* variant regarding the correction of *neighbor substitutions*, because in general the bottom-right pattern reported by other authors [34, 17, 18], on which the *Shifted* variant was based, could not be found.

In the second study (**Study With The Older Adults**) the goal was to better understand older users typing patterns and find the most fitting variant(s). Older users in general, as well as inexperienced older users specifically, performed worse with the ***Color* variant**, regarding input speed (just like younger users). It is hypothesized that the highlighting of the keys distracted the users (similar to Henze et al. [18] results) and also lead to more cognitive substitutions. Still, users were satisfied with the *Color* variant, and emphasized it was a big help, since it helped them to locate the desired key. The ***Predict Words* variant** was as good as the traditional QWERTY keyboard in terms of input speed and error rate. Still, thirteen of the participants used the *Predict Words* variant as the *traditional QWERTY* keyboard; that is, ignoring the suggested words. The ***Shifted* simulation** was able to reduce significantly the error rate, specifically *neighbor substitution*, *failed omission* and *slide omission* errors, even more than on the younger adults case. This was possible because the touch typing pattern of the users was similar to the one described by several authors [34, 17, 18] and the compensating shift applied was based on the results of those authors. Furthermore, it was found that **vertical shifts increase gradually from row to row** until the space bar is reached. It was also found that the **horizontal shift pattern is closely related with the hand the user is using to type** as hypothesized by Nicolau [34]; that is, the **shift is more intense towards the side of the keyboard of the hand the user is using to type**. For instance, when users interact only with their right hand, the horizontal shift is more intense to the right on the right side of the keyboard. When users interact with both hands, the horizontal shift is more intense to the right on the right side of the keyboard, when compared with the center of the keyboard. Conversely, the horizontal shift is more intense to the left on the left side of the keyboard. Even though the ***Size Invisible* simulation** did not perform as well as the *Shifted* variant, it was able to significantly reduce the error rate (similar to the results obtained by Gunawardana [15]), specifically *neighbor substitution*, *failed omission* and *slide omission* errors. This variant had better results for the younger adults, mainly because, since no touch typing pattern emerged, increasing the underlying area of the four most probable keys helped to decrease neighbor substitution errors. This ensures that, independently of the position of the touch point (top left, top right, bottom left or bottom right), it will be assigned to the intended key, as long as the intended key is highlighted.

We also found that, in general, a single touch keyboard is more advantageous for older users than a multi-touch one. The ***Single Touch*** simulation was able to reduce significantly *extra-finger insertions*. Regarding the Timed simulations, the ***Intra-key*** did not perform as well as we expected, because it is hard to define a threshold that allows maximizing the rejection of *accidental insertion* errors, and minimizing the rejection of false positives. Although it is also hard to define a threshold for the ***Inter-key Timed* variant**, it was able to reduce significantly ***double insertion* errors**. The ***Combined Timed* variant** did not perform as well as we expected, mainly because of the Intra-key threshold. Finally, a result that, to the authors' knowledge, has **never** been reported by other authors, is that **when users perform a vertical slide from one row to a subsequent row (up or down), 96.4% of the times the user intends to select the key from the row above.**

12.1 Future Work

Re-Evaluate the *Color* and *Predict Words* variants. In the user study performed, each older user was only allowed to type two sentences with each variant before beginning the evaluation. A new study should be performed that allows users to experiment each of these variants for a longer period of time (e.g.: one week). A study should then be performed, to assess if the users were able to improve their performance on these two variants.

Evaluate the *Shifted* variant with more participants that only use their left hand and/or both hands to type. The majority of participants used only their right hand to type. This allowed to investigate the touch typing behavior of these users. Since less users used their left hand and both hands, the results obtained were not as strong. Therefore, a new study should be performed in order to verify the touch typing patterns reported by the authors.

A *conditional Single Touch* variant, that rejects interactions based on a prediction algorithm should be developed and evaluated. The proposed Single Touch approach was limited, because it rejected interactions based only in the order of touch interactions (it rejected a second press, if the first press was not yet released). A new approach should reject touch interactions, based on the probability of each of the pressed keys. The same could also be applied to the *Inter-key* variant.

The *Single Touch* and *Inter-key Timed* variants should be evaluated in a user study (instead of a simulation study). The simulation presented in this paper indicates that these variants are promising, still it is important to assess how would users react to them. In the case of the Inter-key variant, it would also be important to develop a mechanism that adapts the threshold to the user, depending on its past interactions.

Even though the approach presented allowed to learn more about the similarities and differences between younger and older adults regarding virtual keyboards, future studies should focus on older adults, so that design decisions are only made based on the population who will use the system.

Acknowledgements This work was supported by FCT (INESC-ID multiannual funding) under the project PEst-OE/ EEI/LA0021/2013 and the project PAELife, reference AAL/ 0014/2009.

References

1. Al Faraj, K., Mojahid, M., Vigouroux, N.: Bigkey: A virtual keyboard for mobile devices. In: Proceedings of the 13th International Conference on Human-Computer Interaction. Part III: Ubiquitous and Intelligent Interaction, pp. 3–10. Springer-Verlag, Berlin, Heidelberg (2009)
2. Annett, M., Anderson, F., Goertzen, D., Halton, J., Ranson, Q., Bischof, W.F., Boulanger, P.: Using a multi-touch tabletop for upper extremity motor rehabilitation. In: Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7, OZCHI '09, pp. 261–264. ACM (2009)

3. Apted, T., Kay, J., Quigley, A.: Tabletop sharing of digital photographs for the elderly. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '06, pp. 781–790. ACM (2006)
4. Azenkot, S., Zhai, S.: Touch behavior with different postures on soft smartphone keyboards. In: Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services, MobileHCI '12, pp. 251–260. ACM, New York, NY, USA (2012). DOI 10.1145/2371574.2371612
5. Bain, P., Mally, J., Gresty, M., Findley, L.: Assessing the impact of essential tremor on upper limb function. Journal of Neurology **241**(1) (1993). DOI 10.1007/BF00870673. URL http://dx.doi.org/10.1007/BF00870673
6. Brewster, S., Chohan, F., Brown, L.: Tactile feedback for mobile interactions. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07, pp. 159–162. ACM, New York, NY, USA (2007). DOI 10.1145/1240624.1240649. URL http://doi.acm.org/10.1145/1240624.1240649
7. Carstensen, L.L., Hartel, C.R.: When I'm 64, 1st edition edn. The National Academies Press (2008)
8. Cheng, L.P., Liang, H.S., Wu, C.Y., Chen, M.Y.: igrasp: grasp-based adaptive keyboard for mobile devices. In: CHI '13 Extended Abstracts on Human Factors in Computing Systems, CHI EA '13, pp. 2791–2792. ACM, New York, NY, USA (2013). DOI 10.1145/2468356.2479514
9. Dunlop, M., Levine, J.: Multidimensional pareto optimization of touchscreen keyboards for speed, familiarity and improved spell checking. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, pp. 2669–2678. ACM, New York, NY, USA (2012). DOI 10.1145/2207676.2208659. URL http://doi.acm.org/10.1145/2207676.2208659
10. Findlater, L., Wobbrock, J.: Personalized input: improving ten-finger touchscreen typing through automatic adaptation. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, pp. 815–824. ACM, New York, NY, USA (2012). DOI 10.1145/2207676.2208520
11. Fisk, A.D., Rogers, W.A., Charness, N., Czaja, S.J., Sharit, J.: Designing for Older Adults: Principles and Creative Human Factors, 2nd edition edn. Human Factors and Aging Series. CRC Press (2009)
12. Garay, N., Abascal, J.: Using statistical and syntactic information in word prediction for input speed enhancement. Information Systems Design and Hypermedia pp. 223–230 (1994)
13. Garay-Vitoria, N.: Sistemas de predicción lingüística. aplicación a idiomas con alto y bajo grado de flexión, en el ámbito de la comunicación alternativa y aumentativa. Servicio Editorial de la UPV-EHU, Leioa (Bizkaia) (2001)
14. Garay-Vitoria, N., Abascal, J.: Text prediction systems: a survey. Univers. Access Inf. Soc. **4**(3), 188–203 (2006). DOI 10.1007/s10209-005-0005-9
15. Gunawardana, A., Paek, T., Meek, C.: Usability guided key-target resizing for soft keyboards. In: Proceedings of the 15th international conference on Intelligent user interfaces, IUI '10, pp. 111–118. ACM, New York, NY, USA (2010). DOI 10.1145/1719970.1719986
16. Heckathorne, C., Childress, D.S.: Applying anticipatory text selection in a writing aid for people with severe motor impairment. Micro, IEEE **3**(3), 17–23 (1983). DOI 10.1109/MM.1983.291116
17. Henze, N., Rukzio, E., Boll, S.: 100,000,000 taps: analysis and improvement of touch performance in the large. In: Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI '11, pp. 133–142. ACM, New York, NY, USA (2011). DOI 10.1145/2037373.2037395
18. Henze, N., Rukzio, E., Boll, S.: Observational and experimental investigation of typing behaviour using virtual keyboards for mobile devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '12, pp. 2659–2668. ACM, New York, NY, USA (2012). DOI 10.1145/2207676.2208658
19. Hoaglin, D.C., Iglewicz, B.: Fine-tuning some resistant rules for outlier labeling. Journal of the American Statistical Association **82**(400), 1147–1149 (1987)
20. Hoggan, E.E., Brewster, S.A., Johnston, J.: Investigating the effectiveness of tactile feedback for mobile touchscreens. In: Computer Human Interaction, pp. 1573–1582 (2008). DOI 10.1145/1357054.1357300
21. Hunnicutt, S.: Input and output alternatives in word prediction. STL/QPRS **2**(3), 15–29 (1987)
22. Hwangbo, H., Yoon, S.H., Jin, B.S., Han, Y.S., Ji, Y.G.: A study of pointing performance of elderly users on smartphones. International Journal of Human-Computer Interaction **29**(9), 604–618 (2013)
23. Jin, Z.X., Plocher, T., Kiff, L.: Touch screen user interfaces for older adults: button size and spacing. In: Proceedings of the 4th international conference on Universal access in human computer interaction: coping with diversity, UAHCI'07, pp. 933–941. Springer-Verlag, Berlin, Heidelberg (2007). URL http://dl.acm.org/citation.cfm?id=1766311.1766419
24. Ketcham, C.J., Seidler, R.D., Van Gemmert, A.W., Stelmach, G.E.: Age-related kinematic differences as influenced by task difficulty, target size, and movement amplitude. The Journals of Gerontology Series B: Psychological Sciences and Social Sciences (2002)
25. Kobayashi, M., Hiyama, A., Miura, T., Asakawa, C., Hirose, M., Ifukube, T.: Elderly user evaluation of mobile touchscreen interactions. In: Proceedings of the 13th IFIP TC 13 international conference on Human-computer interaction - Volume Part I, INTERACT'11, pp. 83–99. Springer-Verlag, Berlin, Heidelberg (2011). URL http://dl.acm.org/citation.cfm?id=2042053.2042065
26. Lepicard, G., Vigouroux, N.: Touch screen user interfaces for older subjects: effect of the targets number and the two hands use. In: Proceedings of the 12th international conference on Computers helping people with special needs, ICCHP'10, pp. 592–599. Springer-Verlag, Berlin, Heidelberg (2010). URL http://dl.acm.org/citation.cfm?id=1880751.1880851
27. Loureiro, B., Rodrigues, R.: Multi-touch as a natural user interface for elders: A survey. In: Information Systems and Technologies (CISTI), 2011 6th Iberian Conference on (2011)
28. Lucas, D.: Disabled 'r' all. Master's thesis, Instituto Superior Técnico (2012)
29. MacKenzie, I.S., Soukoreff, R.W.: A character-level error analysis technique for evaluating text entry methods. In: Proceedings of the second Nordic conference on Human-computer interaction, NordiCHI '02, pp. 243–246. ACM, New York, NY, USA (2002). DOI 10.1145/572020.572056. URL http://doi.acm.org/10.1145/572020.572056
30. MacKenzie, I.S., Zhang, X.: Eye typing using word and letter prediction and a fixation algorithm. In: Proceed-

ings of the 2008 symposium on Eye tracking research & applications, ETRA '08, pp. 55–58. ACM, New York, NY, USA (2008). DOI 10.1145/1344471.1344484
31. Mackenzie, S.I., Soukoreff, W.R.: Text entry for mobile computing: Models and methods,theory and practice. Human-Computer Interaction **17** (2002)
32. Matero, J., Colley, A.: Identifying unintentional touches on handheld touch screen devices. In: Proceedings of the Designing Interactive Systems Conference, DIS '12, pp. 506–509. ACM, New York, NY, USA (2012). DOI 10.1145/2317956.2318031. URL http://doi.acm.org/10.1145/2317956.2318031
33. Medline Plus: Aging changes in the bones - muscles - joints. http://www.nlm.nih.gov/medlineplus/ency/article/004015.htm (2012)
34. Nicolau, H.: Disabled 'r' all: Bridging the gap between health- and situationally-induced impairments and disabilities. Ph.D. thesis, Instituto Superior Técnico (2013)
35. Nicolau, H., Jorge, J.: Elderly text-entry performance on touchscreens. In: Proc. of the 14th international ACM SIGACCESS conference on Computers and accessibility, ASSETS '12. ACM (2012). DOI 10.1145/2384916.2384939. URL http://doi.acm.org/10.1145/2384916.2384939
36. Oulasvirta, A., Reichel, A., Li, W., Zhang, Y., Bachynskyi, M., Vertanen, K., Kristensson, P.O.: Improving two-thumb text entry on touchscreen devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, pp. 2765–2774. ACM, New York, NY, USA (2013). DOI 10.1145/2470654.2481383
37. Pak, R., McLaughlin, A.: Designing Displays for Older Adults. Human Factors & Aging Series. CRC PressINC (2010)
38. Piper, A.M., Campbell, R., Hollan, J.D.: Exploring the accessibility and appeal of surface computing for older adult health care support. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '10, pp. 907–916. ACM (2010)
39. Poupyrev, I., Maruyama, S.: Tactile interfaces for small touch screens. In: Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology, UIST '03, pp. 217–220. ACM, New York, NY, USA (2003). DOI 10.1145/964696.964721. URL http://doi.acm.org/10.1145/964696.964721
40. Soukoreff, R.W., MacKenzie, I.S.: Recent developments in text-entry error rate measurement. In: CHI '04 Extended Abstracts on Human Factors in Computing Systems, CHI EA '04, pp. 1425–1428. ACM, New York, NY, USA (2004). DOI 10.1145/985921.986081. URL http://doi.acm.org/10.1145/985921.986081
41. Stone, R.G.: Mobile touch interfaces for the elderly. In: Proceedings of ICT, Society and Human Beings. International Association for Development of the Information Society (IADIS) (2008)
42. Swiffin, A., Arnott, J., Pickering, J.A., Newell, A.: Adaptive and predictive techniques in a communication prosthesis. Augmentative and Alternative Communication **3**(4), 181–191 (1987)
43. Trewin, S., Pain, H.: A model of keyboard configuration requirements. In: Proceedings of the Third International ACM Conference on Assistive Technologies, Assets '98, pp. 173–181. ACM, New York, NY, USA (1998). DOI 10.1145/274497.274530. URL http://doi.acm.org/10.1145/274497.274530
44. Tsai, T.H., Chang, H.T., Chang, Y.M., Huang, G.S.: Sharetouch: A system to enrich social network experiences for the elderly. J. Syst. Softw. **85**(6), 1363–1369 (2012)
45. Venkatagiri, H.: Efficiency of lexical prediction as a communication acceleration technique. Augmentative and Alternative Communication **9**(3), 161–167 (1993)
46. Victoria Meza-Kubo Alberto L Morán, M.R.: Intouchfun, a pervasive collaborative system to cope with elder's isolation and cognitive decline. Proceedings of the UbiComp (2010)
47. Werner, F., Werner, K., Oberzaucher, J.: Tablets for seniors - an evaluation of a current model (ipad). In: R. Wichert, B. Eberhardt (eds.) Ambient Assisted Living, Advanced Technologies and Societal Change, pp. 177–184. Springer Berlin Heidelberg (2012)
48. WHO: Global health and aging. http://www.who.int/ageing/publications/global_health.pdf (2012)
49. Wobbrock, J.O., Morris, M.R., Wilson, A.D.: Understanding users' preferences for surface gestures. In: Proceedings of Graphics Interface 2010, GI '10, pp. 261–268. Canadian Information Processing Society, Toronto, Ont., Canada, Canada (2010). URL http://dl.acm.org/citation.cfm?id=1839214.1839260
50. Yin, Y., Ouyang, T.Y., Partridge, K., Zhai, S.: Making touchscreen keyboards adaptive to keys, hand postures, and individuals: a hierarchical spatial backoff model approach. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13, pp. 2775–2784. ACM, New York, NY, USA (2013). DOI 10.1145/2470654.2481384