

A new concept of assistive virtual keyboards based on a systematic review of text entry optimization techniques

Renato de Sousa Gomide*, Luiz Fernando Batista Loja, Rodrigo Pinto Lemos, Edna Lúcia Flôres, Francisco Ramos Melo, Ricardo Antonio Gonçalves Teixeira

Abstract Introduction: Due to the increasing popularization of computers and the internet expansion, Alternative and Augmentative Communication technologies have been employed to restore the ability to communicate of people with aphasia and tetraplegia. Virtual keyboards are one of the most primitive mechanisms for alternatively entering text and play a very important role in accomplishing this task. However, the text entry for this kind of keyboard is much slower than entering information through their physical counterparts. Many techniques and layouts have been proposed to improve the typing performance of virtual keyboards, each one concerning a different issue or solving a specific problem. However, not all of them are suitable to assist seriously people with motor impairment. **Methods:** In order to develop an assistive virtual keyboard with improved typing performance, we performed a systematic review on scientific databases. **Results:** We found 250 related papers and 52 of them were selected to compose. After that, we identified eight essentials virtual keyboard features, five methods to optimize data entry performance and five metrics to assess typing performance. **Conclusion:** Based on this review, we introduce a concept of an assistive, optimized, compact and adaptive virtual keyboard that gathers a set of suitable techniques such as: a new ambiguous keyboard layout, disambiguation algorithms, dynamic scan techniques, static text prediction of letters and words and, finally, the use of phonetic and similarity algorithms to reduce the user's typing error rate.

Keywords: Assistive technology, Keyboard layout, Motor impairment, Text prediction, Typing performance, Virtual keyboard.

Introduction

In recent times, much effort has been done in developing technologies and techniques in order to help social inclusion of people with disabilities (Galvão and Garcia, 2012). This trend led to a new field called Assistive Technology (AT). Cook and Polgar (2014) defines AT according to the concept created by Colker (1999): "a wide range of equipment, services, strategies and practices designed and implemented to reduce the functional problems encountered by individuals with disabilities".

A sub-area of Assistive Technology is Augmentative and Alternative Communication (AAC). This area comprises the methods and technologies designed to assist or replace communication of people with speech limitation (Wilkinson and Hennig, 2007). According to Park et al. (2012), due to the increasing popularity of computers and the expansion of the Internet, several studies were developed to assist the communication of patients with aphasia and tetraplegia. Computer systems implemented to aid communication of individuals with such characteristics

can be segmented into two distinct components: input devices and communication software.

Input devices are equipment used to capture any type of patient's voluntary intent. Park et al. (2012) and Cipresso et al. (2011) presented researches using eye movements as input, while Mele and Federici (2012) presents a systematic review over this subject. The works of Al-Abdullatif et al. (2013), Blain et al. (2008), Schalk et al. (2008) and Usakli and Gurkan (2010) used patient's brain activity to allow communication. The area responsible for creating a communication channel between the brain and the computer is called brain computer interface (BCI).

The communication software is the program developed to analyze the data captured by the input devices and turn them into information. These programs are diverse and range from virtual keyboards (Doval et al., 2010; Fu and Ho, 2009; Orhan et al., 2012) to complex communication spreadsheets (Biswas and Samanta, 2008; Mason and Chinn, 2010).

According to Molina et al. (2009b) a virtual keyboard is a kind of software that shows a keyboard layout

*e-mail: renato@doutorado.ufu.br

on the computer screen. This sort of keyboard is one of the most primitive mechanisms for alternatively entering text (Ghosh, 2011). Studies like (Arif and Stuerzlinger, 2013; Kim et al., 2014; Kwon et al., 2009) indicate that virtual keyboards shows lower typing performance than physical keyboards, even when users have no disability. The main reasons that make virtual keyboard slower than their physical counterpart are the small size of the virtual keys, absence of tactile feedback and occlusion of virtual keys by fingers (Kim et al., 2014; Kwon et al., 2009).

Entering performance is even slower for users with impairments, due to patient's motor restrictions that limit their ability to interact with the keyboard software. There is a wide range of disabilities ranging from simple motor limitations, such as loss of arm movement, to even more serious limitations in which only eye movements are allowed. In order to grant patients communicate using the virtual keyboard, various interaction strategies are used.

Users with classic LIS produce only a single type of stimulus, so they are limited to communicate objectively (*activate, deactivate*) (Steriadis and Constantinou, 2003). For these patients, the computer must scan the keys of the virtual keyboard in such a way that, once the user identifies the desired key, he sends the stimulus to stop scanning and choose the key (Rivera et al., 2009). Then, the software recognizes the stimulus and selects the desired key (Miró-Borrás and Bernabeu-Soler, 2009). According to Rivera et al. (2009) and Miró-Borrás and Bernabeu-Soler (2009), this sort of interaction technique limits the user writing performance. In order to maximize the number of words per minute that virtual keyboard users can type, several techniques can be applied.

This systematic review identified methods and techniques ranging from optimization of keyboard performance structure, like keyboard layouts and sequences of letters, to complex text prediction techniques. However, even using a huge variety of methods and techniques to optimize virtual keyboards, their typing performance is still low.

According to Simathamamand and Piromsopa (2011), Varcholik et al. (2012) and Hoste and Signer (2013), a physical keyboard can produce more than 30 words per minute if operated by experienced typists. Assistive virtual keyboards operated by a person with motor impairment typically produce only 4 to 7 words per minute (Miró-Borrás and Bernabeu-Soler, 2009). The physical keyboard overcomes the assistive virtual keyboard in approximately 400%, in the best case.

There is still no definitive solution to solve the problem of low typing performance of virtual keyboards used by people with disability (Poláček et al. 2012).

This study intends to identify the virtual keyboard characteristics that influence the entry performance, the main techniques used to optimize it and the main methods responsible for measuring the typing rate performance.

The results of our Systematic Review (SR) were used to design a new assistive virtual keyboard to the Brazilian Portuguese language proposed in this paper. This keyboard will be set with the most appropriate features brought up in the SR. It will also implement some optimization techniques gathered from the review in order to improve the data entry for people with motor disability. Then, the measures of text entry performance collected by this research will be applied to the new virtual keyboard in order to evaluate the performance of this new proposal.

This paper is structured as follows: Section *Planning* describes the planning of this review, the parameters adopted in the search engines and the criteria of selection. The systematic review and a preliminary selection are presented in Section *Protocol Implementation*. Section *Data Analysis* shows the final selection and also answers the research questions. Then, a new assistive virtual keyboard is derived from this analysis in Section *Proposal of an Assistive, Optimized, Compact and Adaptive Virtual Keyboard*. Finally, Section *Conclusion* brings the concluding remarks and suggestions of future works.

Planning

This SR was planned according to the protocol presented by Biolchini et al. (2007), and we state its main aspects in this section.

Research objectives

The aim of this work is to identify and analyze the characteristics of virtual keyboards that influence the typing performance, as well as the main computational techniques that have been applied to optimize data entry. Finally metrics and parameters for assessing the performance of virtual keyboards will be identified. Specifically we intend to:

- Objective 1: Identify a composition of virtual keyboard characteristics suitable for users with motor impairment.
- Objective 2: Gather the optimization methods and techniques applicable for assistive virtual keyboards.
- Objective 3: Elect two or more measures to assess the virtual keyboard performance.

Formulation and research question: Scope and specificities

The main goal of the research questions is to identify the works that state about virtual keyboard:

- Research question 1: Which are the essential characteristics of a virtual keyboard?
- Research question 2: Which are the techniques and methods used to increase data entry performance of virtual keyboards?
- Research question 3: Which are the metrics used to evaluate the performance of data entry of virtual keyboards?

The first question was used to find the essential characteristics of a virtual keyboard. The second examines the techniques applied to improve the typing performance and, finally, the third intends to identify how to measure the input speed of a virtual keyboard. The specificities of this study are described below:

- Intervention: Virtual keyboards characteristics, optimization techniques and methods. Methods used to quantify the performance data input by virtual keyboards.
- Control: This research initially began with the work of Garay-Vitoria and Abascal (2006), MacKenzie and Tanaka-Ishii (2007), Miró-Borrás and Bernabeu-Soler (2009), Prabhu and Prasad (2011) and Yang et al. (2013).
- Population: Virtual keyboards used by patients with physical disabilities.
- Results: Identify which virtual keyboard characteristics influence the text input and which optimization techniques have been, applied to improve the performance of data entry. Finally, we intend to discover which are the metrics used to quantify the typing performance.
- Application: This work will provide theoretical and practical resources for developers and researchers who want to implement an optimized virtual keyboard for patients with simultaneous motor impairment of the upper limbs and speech.

Search strategy for selecting studies

Initially, we defined the selection criteria and which searching methods (manual, electronic search engines, etc.) would be considered. Then, we chose the

languages for which the search would be restricted. Finally, we set the keywords and search strings.

The keywords were defined according to the objectives and issues presented in this work. In order to identify more papers related to virtual keyboards, we used only the term “virtual keyboard” to perform the search. The choice of a single term had the goal to find any work related to virtual keyboards.

This strategy allowed us to identify valuable text entry optimization techniques, despite those works were not meant to assist people with impairment. Additionally, the search was carried out using additional terms like: Augmentative and Alternative Communication, Assistive Technology, Communication and Locked in Syndrome. Nevertheless only a few works were returned for this reason, only one keyword was used.

The search strategy and all its features are described as follows:

- Criteria for sources selection: only indexed databases and internet-based electronic search engines were selected.
- Search methods of sources: in databases and search engines, we used filters for data along with the keywords. The search was performed only on the titles and abstracts of the papers.
- Keywords: only a single entry - virtual keyboard.
- Sources: according to Kitchenham and Brereton (2013), we had better seek in specific search databases and use at least one general search engine. Then, we chosen IEEE Xplore as the specific database and both Science Direct Portal and Scopus database aggregator as general search engines.
- Study types: journal and conferences papers, patents, reviews, theses and dissertations.
- Language of the studies: English has been chosen due to its international acceptance for publishing scientific papers.

Criteria and procedures for selecting studies

With the aim of picking the most relevant works, we refined the search by applying inclusion and exclusion criteria to the title and abstract of each selected document. These criteria were suited to the search strategies lately described in this work. Then, we removed both the irrelevant papers for this review and those with incomplete electronic version. These criteria are described in the following:

Inclusion criteria

The purpose of the inclusion criteria is to qualify the relevance of each work according to this systematic review.

- Inclusion Criterion 1 (IC1): papers must be digitally available for free on the internet or through institution agreements.
- Inclusion Criterion 2 (IC2): only full papers written in English must be considered.
- Inclusion Criterion 3 (IC3): Studies related to communication using virtual keyboards.
- Inclusion Criterion 4 (IC4): Present a method or technique that improves or has the intention to optimize virtual keyboard typing performance.
- Inclusion Criterion 5 (IC5): Present metrics to measure the typing performance of virtual keyboards.

Exclusion criteria

The aim of the exclusion criteria is discard papers that do not fit on this systematic review.

- Exclusion Criterion 1 (EC1): Use of virtual keyboards for different purposes than communication.
- Exclusion Criterion 2 (EC2): Techniques or methods that do not have the goal of improving the typing performance.
- Exclusion Criterion 3 (EC3): Optimization techniques that can not be adapted and applied to patients with motor disabilities.

Search string

The purpose of drafting the search string was to identify works related to virtual keyboards in general. Manual searches were carried out including several terms like “characteristics of virtual keyboards” and “optimized virtual keyboards”, however it was noticed that this search would miss many papers in this subject due to the specificity of the search string.

Therefore, we adopted a broader quest. To modify the string search we considered the control articles (Miró-Borrás and Bernabeu-Soler, 2009; Prabhu and Prasad, 2011; Yang et al., 2013). We modified the search string and checked if the new search retrieved these control articles. After various searches, the best approach was using a single term. Thus, we simply chose the term “virtual keyboard” as the search string.

Process for selecting studies

The search string was used for searching on indexed sources in the preliminary selection of the studies. After the selection process, the works were catalogued to ensure that each document was selected only once. Thus, the documents were distributed to two researchers who read their abstracts and conclusions.

Each researcher used the inclusion and exclusion criteria outlined in Section *Criteria and procedures for selecting studies* to decide whether the work was appropriate or not for this systematic review (SR). In case of disagreement, the different opinions were discussed until a consensus was achieved.

After evaluating all works, each researcher recorded his reasons of including or excluding each work. In the final selection, those documents included in the preliminary phase were integrally read and evaluated according to the two questions stated in Section *Formulation and research question: Scope and specificities*. Finally, this evaluation determined whether or not the work would be included in this SR.

Protocol implementation

Only the search engines and digital libraries accessible through CAPES portal we considered. Initially, we performed the search on the IEEE Xplore database. Then, we used the search string firstly in Science Direct and next in Scopus aggregator. We searched for works that were published between the years 2009 until 2014.

We retrieved a total of 250 papers: 58 on IEEE xplore; 12 on Science Direct and 180 on Scopus. Then, we used the reference manager JabRef 2.9.2/2013 to organize the documents retrieved.

We found 63 duplicate papers among the bases. After removing the duplicates, 187 papers remained. Reading their titles and abstracts allowed us to eliminate irrelevant references, leaving 52 papers to be fully read. Figure 1 illustrates the steps performed in implementing the search protocol.

After reading all papers, both researchers have developed a categorization sheet to classify the works according to the questions presented in this review. Three main categories were drawn up, they were: “Virtual keyboards characteristics”, “Methods and data entry optimization techniques” and “Data entry performance metrics”.

Papers in the “Virtual keyboards characteristics” category usually define, modify or enhance virtual keyboards features. Therefore, this category gathered works that state about: distribution of symbols,

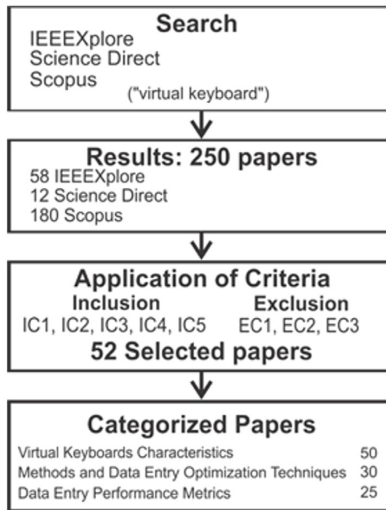


Figure 1. Steps performed in implementing the search protocol.

position of keys, keys size, amount of symbols per key, navigation style and typing feedback.

The category “Methods and techniques to data entry optimization” comprised articles that describe techniques to maximize virtual keyboards data entry. They address the following issues: the definition of the best virtual keyboard layout, text prediction techniques, methods to minimize typing errors, error correction methods and software customization.

Finally, “Data entry performance metrics” category includes all papers that describe methods and formulas to measure the virtual keyboards input rate. Thus, this category bring together: papers that compare or analyze text entry speed in words or gestures and characters per minute, and works that measure the amount of errors inserted by the use of such keyboards.

Notice that one work can be classified into multiple categories. For instance, a paper that both describes a Text Input Optimization Method and evaluates its performance was assigned to the second and third category.

Out of the 52 fully read papers, 50 described or shown by figures the Characteristics of Virtual Keyboards, 30 commented about one or more Methods and Data Entry Optimization Techniques and 25 stated at least one method of Data Entry Performance Metrics. As shown in Tables 1-3, each category was split into subcategories.

Data analysis

This section aims to answer the key questions of this SR while summarizing the information collected in all works selected, as presented in Table 1.

Question 1: Virtual keyboards characteristics

Originally this systematic review concerns to elucidate the main features of a virtual keyboard regarding the typing performance. First, we define *virtual keyboard characteristics* as any attributes belonging to a virtual keyboard, essential for its use and display that can vary according to the set up of each keyboard.

Based on this concept, this review identified the following eight characteristics inherent to all virtual keyboards: positioning of keys, amount of keys, key size, distribution of symbols over the keyboard, special character presentation, number of symbols in each key, keyboard feedback and navigation style.

Several of these characteristics are related directly to the virtual keyboard layout. This layout involves the positioning of keys, amount of keys, key size, distribution of symbols over the keyboard and number of symbols in each key.

The *positioning of keys* is the characteristic that defines how the keys are visually positioned on the keyboard. We identified two ways of organizing the keys on the virtual keyboard: matrix or circular shape. Most of the works arrange the keys on matrix shape, while only few authors use the circular shape. The Figure 2 shows two distinct forms of circular keyboard.

Besides the geometry of the virtual keyboard, several studies are concerned with the *key size*, such that virtual keyboards can be set up using fixed size or variable size keys.

The *distribution of symbols* over the keyboard determines the order in which symbols are presented in the virtual keyboard. According to Joshi et al. (2011) this distribution occurs in two ways: by frequency or logically. However, the work of Bhattacharya and Laha (2012) adds another form of distribution called adaptive.

The distribution of symbols based on their frequency arranges the symbols by taking on account the frequency each symbol is used in a given language corpus (Joshi et al., 2011). According to Bhattacharya and Laha (2012), the language corpus is a representative collection of texts of different sizes and styles. This composition of texts aims to represent the language which it belongs. The frequency distribution makes easier to the virtual keyboard user to access the most frequent symbols (Bhattacharya and Laha, 2012).

The logical distribution of the symbols is based on the logic of the language used on the keyboard (Joshi et al., 2011). One instance of this approach is the arrangement in alphabetical order. The keyboards

Table 1. Papers related to characteristics categories.

Characteristics	Options	Works
Positioning of keys	Matrix	(An et al., 2013), (Arif and Stuerzlinger, 2013), (Bhattacharya and Laha, 2012), (Burgbacher and Hinrichs, 2014), (Rivera et al., 2009), (Faraj et al., 2009a), (Faraj et al., 2009a), (Faraj et al., 2009b), (Francis and Johnson, 2011), (Gelormini and Bishop, 2013), (Ghosh et al., 2010), (Ghosh, 2011), (Gizatdinova et al., 2012a), (Gizatdinova et al., 2012b), (Guerrier et al., 2011), (Guerrier et al., 2013), (Guerrier et al., 2013), (Jain and Bhattacharya, 2010), (Janpinijrut et al., 2011b), (Kim et al., 2014), (Klima and Slovacek, 2009), (Kwon et al., 2009), (Merino et al., 2010), (Merino et al., 2012), (Miró-Borrás and Bernabeu-Soler, 2009), (Nicolau et al., 2013), (Panwar et al., 2012), (Phanchaipetch and Nattee, 2012), (Poláček et al., 2012), (Sarafis and Markoulidis, 2010), (Sharma et al., 2010), (Sharma et al., 2012), (Simathamanand and Piromsopa, 2011), (Truong et al., 2013), (Usakli and Gurkan, 2010), (Varcholik et al., 2012), (Vella and Vigouroux, 2013), (Wu et al., 2014), (Yang et al., 2013)
	Circular	(Prabhu and Prasad, 2011), (Topal et al., 2012)
Distribution of symbols	Frequency	(Bhattacharya and Laha, 2012), (Faraj et al., 2009c), (Francis and Johnson, 2011), (Ghosh et al., 2010), (Ghosh, 2011), (Guerrier et al., 2013), (Jain and Bhattacharya, 2010), (Joshi et al., 2011), (Merino et al., 2010), (Merino et al., 2012), (Millet et al., 2009), (Panwar et al., 2012), (Prabhu and Prasad, 2011), (Sarafis and Markoulidis, 2010), (Simathamanand and Piromsopa, 2011), (Topal et al., 2012), (Vella and Vigouroux, 2013)
	Logically	(Arif and Stuerzlinger, 2013), (Bhattacharya and Laha, 2012), (Burgbacher and Hinrichs, 2014), (Eklund et al., 2010), (Rivera et al., 2009), (Faraj et al., 2009a), (Gelormini and Bishop, 2013), (Gizatdinova et al., 2012a), (Gizatdinova et al., 2012b), (Guerrier et al., 2011), (Guerrier et al., 2013), (Joshi et al., 2011), (Kim et al., 2014), (Klima and Slovacek, 2009), (Krejcar, 2011), (Kwon et al., 2009), (Merino et al., 2010), (Millet et al., 2009), (Nicolau et al., 2013), (Phanchaipetch and Nattee, 2012), (Poláček et al., 2012), (Sarafis and Markoulidis, 2010), (Sharma et al., 2010), (Truong et al., 2013), (Usakli and Gurkan, 2010), (Varcholik et al., 2012), (Vella and Vigouroux, 2013), (Wu et al., 2014), (Yang et al., 2013)
	Adaptative	(Bhattacharya and Laha, 2012), (Faraj et al., 2009b), (Millet et al., 2009), (Poláček et al., 2012), (Sharma et al., 2012)
Amount of symbols per key	Ambiguous	(Burgbacher and Hinrichs, 2014), (Guerrier et al., 2011), (Janpinijrut et al., 2011a), (Joshi et al., 2011), (Klima and Slovacek, 2009), (Merino et al., 2010), (Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009b), (Prabhu and Prasad, 2011)
	Unambiguous	(An et al., 2013), (Arif and Stuerzlinger, 2013), (Bhattacharya and Laha, 2012), (Corley et al., 2012), (Eklund et al., 2010), (Faraj et al., 2009c), (Faraj et al., 2009a), (Faraj et al., 2009b), (Gelormini and Bishop, 2013), (Ghosh, 2011), (Gizatdinova et al., 2012a), (Gizatdinova et al., 2012b), (Guerrier et al., 2011), (Guerrier et al., 2013), (Janpinijrut et al., 2011b), (Joshi et al., 2011), (Krejcar, 2011), (Kwon et al., 2009), (Merino et al., 2010), (Merino et al., 2012), (Molina et al., 2009a), (Nicolau et al., 2013), (Panwar et al., 2012), (Phanchaipetch and Nattee, 2012), (Poláček et al., 2012), (Sarafis and Markoulidis, 2010), (Sharma et al., 2012), (Simathamanand and Piromsopa, 2011), (Truong et al., 2013), (Varcholik et al., 2012), (Vella and Vigouroux, 2013), (Wu et al., 2014), (Yang et al., 2013)
Feedback	Audible	(Topal et al., 2012)
	Visual	(Nicolau et al., 2013), (Topal et al., 2012)
	Line	(Gizatdinova et al., 2012a; 2012b), (Merino et al., 2010), (Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009a)
Navigation style	Row and column	(Rivera et al., 2009), (Francis and Johnson, 2011), (Merino et al., 2010), (Merino et al., 2012), (Millet et al., 2009), (Molina et al., 2009a), (Poláček et al., 2012)
	Three dimensional	(Corley et al., 2012)
Amount of keys		(Molina et al., 2009b), (Miró-Borrás and Bernabeu-Soler, 2009), (Bhattacharya and Laha, 2012)

Table 1. Continued...

Characteristics	Options	Works
Display special characters	Yes	(An et al., 2013), (Arif and Stuerzlinger, 2013), (Corley et al., 2012), (Francis and Johnson, 2011), (Gelormini and Bishop, 2013), (Ghosh, 2011), (Guerrier et al., 2013), (Janpinijrut et al., 2011b), (Kim et al., 2014), (Klima and Slovacek, 2009), (Merino et al., 2010), (Millet et al., 2009), (Nicolau et al., 2013), (Prabhu and Prasad, 2011), (Sharma et al., 2012), (Simathamanand and Piromsopa, 2011), (Usakli and Gurkan, 2010), (Vella and Vigouroux, 2013), (Wu et al., 2014)
	No	(An et al., 2013), (Arif and Stuerzlinger, 2013), (Arif and Stuerzlinger, 2013), (Bhattacharya and Laha, 2012), (Burgbacher and Hinrichs, 2014), (Corley et al., 2012), (Eklund et al., 2010), (Faraj et al., 2009c), (Faraj et al., 2009a), (Faraj et al., 2009b), (Francis and Johnson, 2011), (Gelormini and Bishop, 2013), (Ghosh, 2011), (Gizatdinova et al., 2012a), (Gizatdinova et al., 2012b), (Guerrier et al., 2013), (Jain and Bhattacharya, 2010), (Janpinijrut et al., 2011a), (Kim et al., 2014), (Klima and Slovacek, 2009), (Merino et al., 2010), (Millet et al., 2009), (Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009b), (Nicolau et al., 2013), (Panwar et al., 2012), (Prabhu and Prasad, 2011), (Sarafis and Markoulidis, 2010), (Sharma et al., 2012), (Simathamanand and Piromsopa, 2011), (Usakli and Gurkan, 2010), (Vella and Vigouroux, 2013), (Vella and Vigouroux, 2013), (Wu et al., 2014)
Key size		(Faraj et al., 2009c), (Faraj et al., 2009a), (Gelormini and Bishop, 2013), (Ghosh et al., 2010), (Nicolau et al., 2013), (Varcholik et al., 2012), (Wu et al., 2014),,

Table 2. Papers related to optimization categories.

Methods and techniques for data entry optimization	Implementation	Works
Letter prediction	n-gram	(Arif and Stuerzlinger, 2013), (Janpinijrut et al., 2011b), (Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009a), (Phanchaipetch and Nattee, 2012), (Sarafis and Markoulidis, 2010), (Sharma et al., 2012)
	k-gram	(Molina et al., 2009a), (Miró-Borrás and Bernabeu-Soler, 2009)
	Prediction not specified	(Janpinijrut et al., 2011b)
Word prediction	Frequency of Occurrence	(Merino et al., 2012), (Sharma et al., 2010)
	N-gram	(Sharma et al., 2012), (Ghosh, 2011)
	Regency of Use	(Bhattacharya and Laha, 2012), (Sharma et al., 2010)
	Words table probability	(Guerrier et al., 2011), (Merino et al., 2010), (Truong et al., 2013)
	Syntactic Probability Table	(Sharma et al., 2010)
	Prediction not specified	(Bhattacharya and Laha, 2012), (Guerrier et al., 2011), (Guerrier et al., 2013), (Guerrier et al., 2013), (Janpinijrut et al., 2011b), (Molina et al., 2009b), (Prabhu and Prasad, 2011), (Topal et al., 2012), (Topal et al., 2012)
Organization of keys and symbols	Evolutionary Algorithm	(Guerrier et al., 2011), (Francis and Johnson, 2011)
	N-gram	(Sarcar et al., 2010), (Ghosh, 2011)
	Digraph Frequency	(Gelormini and Bishop, 2013), (Millet et al., 2009), (Sarcar et al., 2010)
	Single character Frequency	(Gelormini and Bishop, 2013), (Ghosh et al., 2010), (Guerrier et al., 2013), (Jain and Bhattacharya, 2010), (Merino et al., 2012), (Millet et al., 2009), (Panwar et al., 2012), (Prabhu and Prasad, 2011), (Topal et al., 2012)
	Fitts' Law	(Faraj et al., 2009a, b), (Ghosh et al., 2010), (Ghosh, 2011)
	Fitts' Digraph Model	(Ghosh et al., 2010), (Ghosh, 2011), (Jain and Bhattacharya, 2010)
Disambiguation methods	Disambiguation algorithm	(Merino et al., 2010), (Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009b), (Sarafis and Markoulidis, 2010)
	Multi-tap	(Burgbacher and Hinrichs, 2014), (Guerrier et al., 2011), (Molina et al., 2009b), (Prabhu and Prasad, 2011), (Sarafis and Markoulidis, 2010)
Keyboard user adaptation		(Bhattacharya and Laha, 2012), (Ghosh et al., 2010), (Merino et al., 2010)

Table 3. Papers related to data entry performance metrics.

Data entry performance metrics	Works
Words per minute (WPM)	(Arif and Stuerzlinger, 2013), (Rivera et al., 2009), (Faraj et al., 2009c), (Faraj et al., 2009a), (Faraj et al., 2009b), (Ghosh, 2011), (Gizatdinova et al., 2012a), (Guerrier et al., 2011), (Hoste and Signer, 2013), (Jain and Bhattacharya, 2010), (Joshi et al., 2011), (Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009a), (Molina et al., 2009b), (Nicolau et al., 2013), (Panwar et al., 2012), (Sarafis and Markoulidis, 2010), (Sarcar et al., 2010), (Sharma et al., 2012), (Topal et al., 2012), (Vella and Vigouroux, 2013)
Characters per minute (CPM)	(Bhattacharya and Laha, 2012), (Burgbacher and Hinrichs, 2014), (Poláček et al., 2012)
Gestures per character (GPC)	(Poláček et al., 2012)
Keystrokes per character (KSPC)	(Miró-Borrás and Bernabeu-Soler, 2009), (Molina et al., 2009a), (Sarcar et al., 2010), (Wu et al., 2014)
Total error rate (TER)	(Arif and Stuerzlinger, 2013), (Biolchini et al., 2007)
Minimum string distance (MSD)	(Gizatdinova et al., 2012a), (Nicolau et al., 2013)

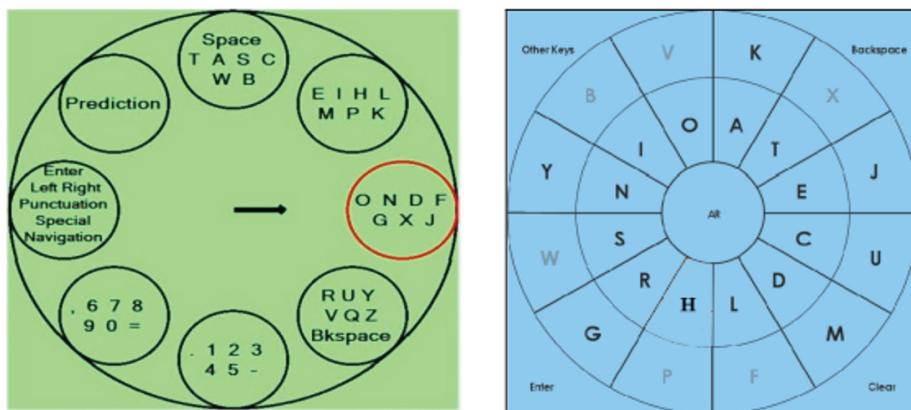


Figure 2. Circular virtual keyboard.

that adopt the alphabetical distribution present letters “a” to “z” sequentially distributed over the keyboard. Other example is QWERTY distribution. Keyboards that use the QWERTY distribution organize the letters in the sequence of letters adopted in their physical correlates. This approach facilitates learning and is easier to be remembered by those who already have experience in physical keyboards (Bhattacharya and Laha, 2012).

The adaptive distribution changes the sequence of letters according to the user behavior, such that the symbols order is dynamically adjusted (Bhattacharya and Laha, 2012). The idea is creating a keyboard that is suited to the way the user types.

The *amount of symbols per key* is also an important feature in the construction of virtual keyboards. Keyboards with a single symbol per key are classified as

unambiguous (Molina et al., 2009a). On the other hand, those with multiple symbols per key are categorized as ambiguous keyboards (Molina et al., 2009a). Besides ambiguity, we noted that the amount of symbols in each key can be homogeneously or heterogeneously distributed. Homogeneously distributed keyboards have the same number of symbols per key, while heterogeneous arrangements have a different amount of symbols per key.

Another feature of ambiguous virtual keyboards is the variety in the *amount of keys*. In unambiguous keyboards, the amount of keys is equal to the number of symbols. However, in ambiguous keyboards, the number of keys may vary. Miró and Bernabeu (2008) made a keyboard using only two keys for text entry, while other works such as Miró-Borrás and Bernabeu-Soler (2009) and Tanaka-Ishii et al. (2002)

use four keys. The number of keys may increase, but is limited to the number of symbols in the alphabet.

The *keyboard feedback* is the way the keyboard communicate to the user that the key has been selected. In physical keyboards, the feedback is tactile, that is, the users can feel the keys being pressed by their fingers. Nevertheless, virtual keyboards are not able to provide this kind of feedback in a natural way and require alternative type of feedback. Some options include providing an audible or visual feedback. Audible feedback is usually provided by playing a sound as soon as a key is selected. Visual feedback is provided by highlighting the selected key in the virtual keyboard.

On the long run, effort reduction is much more important for many people with disabilities because it allows them to continue typing for a longer period with less effort and thus effectively produces more output (Sharma et al., 2010). In addition, there are patients who are able to provide only a single type of signal with active or inactive states. In these cases, the keyboard essentially requires a scanning system (Poláček et al., 2012). For this reason, the scanning method was considered an essential attribute of virtual keyboards.

Poláček et al. (2012) identified four *scanning styles*: linear, row and column, three dimensional and containment hierarchy. The linear scanning moves successively the cursor focus over the keyboard. At each focus movement, the focused key stays highlighted during a certain amount of time. The user selects the key by emitting a signal when the desired key is

highlighted (Schadle, 2004). The Figure 3 shows this scanning style in four steps.

The row and column scan first moves the cursor focus along aligned symbol groups arranged in rows. Then, the selection of the desired key is done in two steps. The first step is to select the desired row. Once the row is selected, the software starts a linear scan along the columns to allow selecting the desired key (Schadle, 2004). This scanning style is shown in Figure 4.

According to Felzer and Rinderknecht (2009), scanning in three dimensions or in block is made by dividing the keyboard into n distinct quadrants. The selection procedure first highlights each quadrant and once the user sends a signal, the highlighted quadrant is selected. After block selection, row and column scan method is employed. The Figure 5 presents this scanning approach.

The containment hierarchy method arranges the keys in a graph-shaped tree. Each node of the graph is associated with a key group and leaf nodes represent only one key. As shown in Figure 6, this scanning method starts by highlighting the top node and allows the user to perform a drill-down operation on the tree graph, highlighting each branch node until reaching a leaf and choosing the desired symbol (Baljko and Tam, 2006).

The scanning method can also be characterized as automatic or manual (Molina et al., 2009b). In automatic scanning, the user must select the keys while the software performs the navigation. In manual

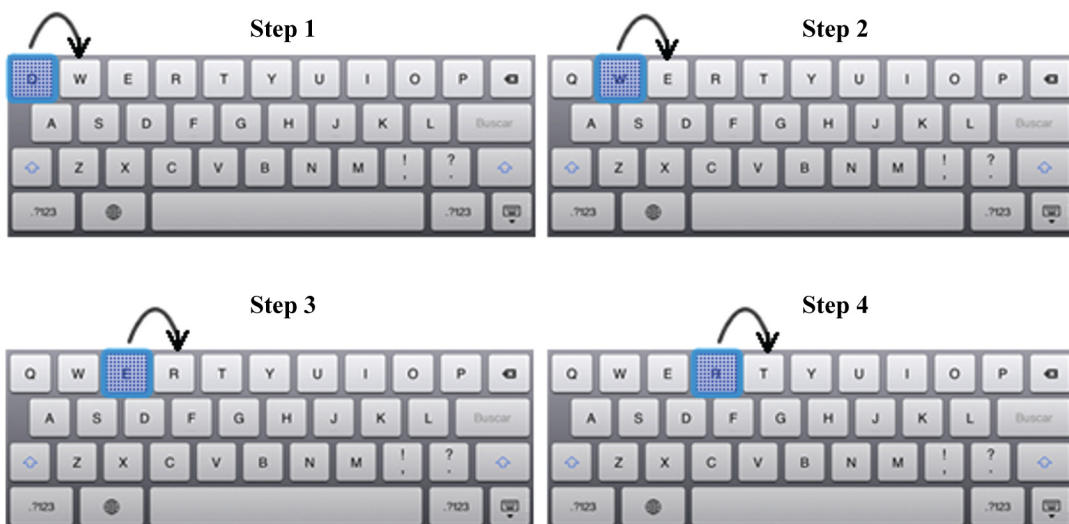


Figure 3. Linear scanning method in four steps. The scanning method passes the keys in sequence from letter q until letter r.

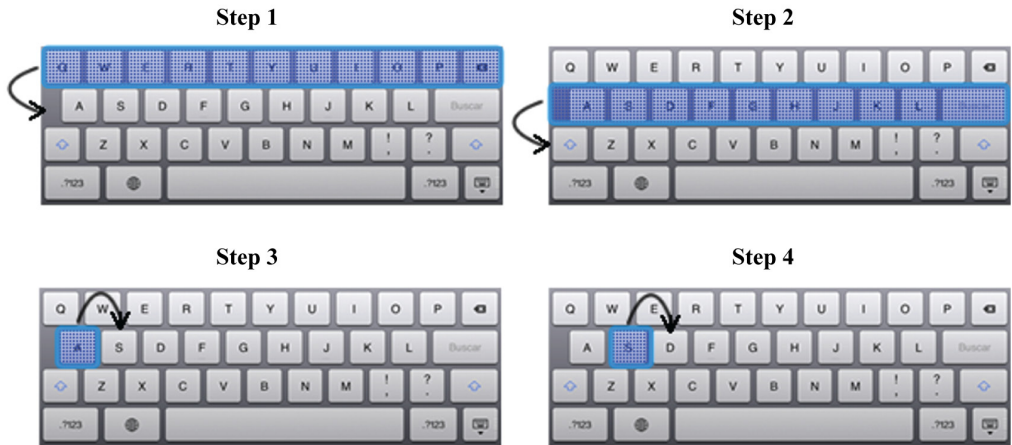


Figure 4. Line and column scanning method in 4 steps. This method highlights the group of keys in line then the user chooses the line in which the letter is. After the first selection the system runs the linear scanning method on the selected group.



Figure 5. Three-dimensional scanning method in 4 steps. This method highlights blocks of keys. Once the user selects a block system performs line and column scanning method.



Figure 6. Containment hierarchy scanning method in 4 steps. This method starts by highlighting the group of characters according to the letter hierarchy. Once a group is selected the system starts the linear scanning method.

scanning, the user is allowed to change the scanning direction, stop its progress or go backwards.

Another variable scan feature is the action after selecting a particular key. This attribute can be snap-to-home or persistent (Millet et al., 2009). Snap-to-home kicks the selection to the first key on the keyboard as soon as a selection is made. Persistent methods continue to scan from the selected key position.

Another important feature of the scanning methods is delay time spent in each key. This time is known as scanning delay and is related directly to the typing performance (Miró-Borrás and Bernabeu-Soler, 2009). Small scanning delays are intuitively associated to a higher typing speed, however very short delays usually lead to typing errors (Francis and Johnson, 2011).

Special characters are of significant importance for keyboard design and efficiency (Sarafis and Markoulidis, 2010). Aiming to improve the typing performance, virtual keyboards may or may not *display special characters*. Some studies remove special characters such as punctuation, navigation and formatting, allowing the scanning method to be optimized. However this approach avoid the user to enter formal texts.

Question 2: Methods and techniques for data entry optimization

Various methods have been investigated in order to optimize the data entry performance and minimize user interaction. This study mainly identified algorithms for predicting letters and words, methods for optimizing keyboard layouts, disambiguation methods, as well as dynamic and adaptive keyboards.

According to Sharma et al. (2010, 2012), prediction can be categorized into two types: syntactic and statistical. Statistical prediction is based on language models like n-gram, such that a list of suggested terms is generated based on the frequency of the words in the language corpus and/or the most recently used word is displayed (Sharma et al., 2012).

The purpose of syntactic prediction is to ensure that the system does not suggest a grammatically inappropriate word to the user (Sharma et al., 2010). Letter prediction methods use only statistical prediction, while word prediction methods use both types of prediction.

The *letter prediction* algorithms help the user by suggesting the characters most likely to occur after the choice of a particular letter sequence. This study identified two language models underlying the prediction algorithms. These models are: n-grams and k-grams.

N-gram model is a widely used technique in language processing (Janpinijrut et al., 2011b). This model can

be used to predict words or letters (Goodman et al., 2002). N-grams suggest the following term of a given sentence based on its previous terms. The amount of terms used to calculate the suggestion may vary. Algorithms that use only one term are called unigram, two terms are called bigram, and trigram in case of three terms. The larger n, the more correct the statistic model produced and more system resources are used to process (Janpinijrut et al., 2011b).

The probability of a term is calculated using the corpus of the desired language and generates a probability table. The algorithm uses this table to analyze the previous terms and suggest which term is the most likely to appear after a certain sequence of terms.

Another language model used for text prediction is the k-gram. According to Miró-Borrás and Bernabeu-Soler (2009) n-grams and k-grams are similar models. The difference between them is that k-grams make the suggestion based on the word being inserted, while n-grams consider the entire previous sentence. Therefore, the n-gram characters used for prediction can begin in any part of the word or even belong to another word.

According to Truong et al. (2013), *word prediction* algorithms suggest a set of words with high probability of occurrence based on a sequence of typed characters. This work identified 5 methods that are the basis for this type of algorithm. These methods are: prediction based on the frequency of occurrence, regency-of-use, word probability table, n-gram and syntactic probability table. Figure 7 shows the prediction algorithms according to their classifications.

Prediction algorithms consider the words and their frequency on a certain corpus of language to predict the word that is being writing (Garay-Vitoria and Abascal, 2006). This kind of prediction is used in conjunction with the method of the regency-of-use. Regency-of-use specify how recently the word has been used. If a word has been used recently, therefore it is highly probable that it will be needed very soon, so this word will have higher priority over the others (Sharma et al., 2010).

The word probability table consists of a list of two words and the likelihood of these terms occur in sequence Garay-Vitoria and Abascal (2006). For every word inserted using the keyboard the system searches which words have the highest probabilities of occurrence after the word entered and build a suggestion list with them. This method is the same as n-gram considering n equals to two.

The syntactic prediction is performed using a probability table using two types of statistics: the probability of each word and the relative probability

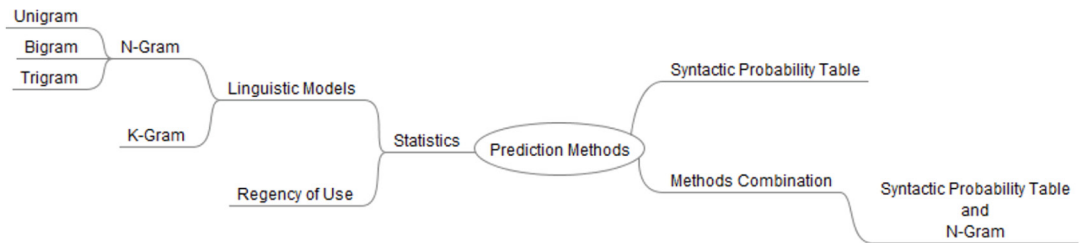


Figure 7. Prediction methods.

of occurrence of each syntactic category after another particular syntactic category (Garay-Vitoria and Abascal, 2006). When the user enters a word, the system calculates the next term using the probability of occurrence of the next word according to its category.

In an efficient virtual keyboard, the *organization of keys and symbols* should be done in order to maximize performance and minimize the typing effort (Ghosh, 2011). In order to achieve this goal, the letters and keys must be optimally reorganized. We found 6 methods devoted to developing an optimized layout: Fitts' law (Fitts, 1954), Fitts' digraph (MacKenzie and Zhang, 1999), frequency of a single character, frequency of digraphs, n-gram and evolutionary algorithms.

According to Simathamanand and Piromsopa (2011), Fitts law is a movement model for human computer interaction (HCI). It measures the movement time based on the distance and size of the keys. Many studies use this ratio to calculate the size and the distance between keys in order to optimize the typing performance. This review noted that Fitts' law is generally used in conjunction with the techniques of frequency of both a single character and digraphs.

The frequency method for a single character calculates the probability of occurrence of a character on a language corpus. This calculation results in a probability table containing each letter and the corresponding frequency of occurrence. From the frequency table, the keys and letters are repositioned and/or resized in order to facilitate access.

The optimization method using digraphs is similar to the frequency method for a single character (Gelormini and Bishop, 2013). The digraphs method calculates the probability of occurrence of a digraph on a corpus of language. This calculation produces a table containing the digraphs and their corresponding frequency of occurrence. According to probability table, both keys and letters are repositioned and/or resized in order to optimally position the digraphs.

The Fitts' digraph method is based both on Fitts' law and digraphs frequency. This model, proposed by MacKenzie and Zhang (1999), estimates the typing

performance based on the sum of the number of movements required by digraphs using the Fitts' law weighted by the frequency of occurrence of digraphs.

The n-gram method calculates the likelihood of a letter based on the n previous terms. Based on the corresponding probability table, the keyboard layout is adjusted such that the letters most likely to occur in sequence are positioned close to each other.

Evolutionary algorithms are also used to optimize the keyboard layout, such as genetic algorithm and Hill Climbing algorithm. However, this type of algorithm requires objective functions to determine whether the proposed layout is a good solution. The evolutionary algorithm identified in this paper uses the Fitts' law (Simathamanand and Piromsopa, 2011) and a particular function (Francis and Johnson, 2011) to determine the optimization of the keyboard layout.

Ambiguous keyboards, by their turn, require resolving the ambiguity of keys. Two *disambiguation methods* were identified: multi-tap and disambiguation algorithms.

In multi-tap method the user chooses a symbol by successively selecting one or more times the same key until the desired symbol is chosen (Kwon et al., 2009). For instance, if a key corresponds to letters a, b and c, and someone wants to choose the character "b", he will have to select the key twice.

Disambiguation algorithms are more complex. This approach uses a database of terms to find candidate words that fit to sequence of symbol groups attributed to the keys selected by the user (Molina et al., 2009b). In this case, users select only once the key containing the desired letter, while the algorithm provides a list of possible words using the chosen sequence of keys.

T9 disambiguation method (Grover et al., 1998) is used to perform disambiguation of words in phones with 9 keys. The TNK algorithm (acronym for Text in N Keys) is a generalization of the T9 method for any number of keys. For both algorithms, a word dictionary is required to provide the most likely word associated to the selected keys sequence (Molina et al., 2009b).

Mackenzie proposed a method named LetterWise to perform the disambiguation process. This method uses a prefix table to perform the disambiguation. The prefix is composed by the letters that precede the next key to be pressed. The algorithm stores the table prefixes and their occurrence probability. For instance, if the user presses the group of letters “d, e, f” after entering the letters “th” the most likely next letter is “e” because “the” in English is far more probable than either “thd” or “thf” (MacKenzie et al., 2001).

This work has identified two ways of adapting the user’s keyboard. The first kind of adaptation is performed before using the communication software. Some approaches allow keyboard customization, be it in the number of keys, the sequence of letters, as well as the kind of word or letter prediction. Another way is to adapt the keyboard during text entry. Studies were identified that adjust the sequence of letters or key size according to the text entered by the user.

Keyboards that adapt the keys and vary the position of the letters according to the user’s writing are known as dynamic. For example, when the user writes a particular word the keyboard repositions letters more likely to be chosen in order to facilitate their selection. Language models as n-gram and k-gram can be used to relocate the keys and or rearrange the sequence of letters.

Question 3: Data entry performance metrics

We identified six types of metrics to evaluate virtual keyboard performance. They are: words per minute (WPM), characters per minute (CPM), gestures per character (GPC), keystrokes per character (KSPC), total error rate (TER), mistake string distance (MSD).

According to Wobbrock (2007), the number of words per minute is perhaps the most used method to measure the performance of typing. WPM is calculated by dividing the amount of transcribed symbols by the time spent to transcribe them in seconds, multiplied by 60 that is the number of seconds in one minute, and then divided by average word length (Nicolau et al., 2013). The average word length is considered to be 5 symbols (Yamada, 1980). This measure considers the elapsed time between entering the first and the last symbol of a sentence.

$$WPM = \frac{|T|-1}{S} * 60 * \frac{1}{5} \quad (1)$$

In Equation 1, T represents the number of transcribed symbols, S references the time elapsed between entering the first and the last symbol of a sequence in seconds.

Characters per minute measures the number of input characters per minute. Bhattacharya and Laha

(2012) uses the same form of the WPM calculation, only suppressing the division by 5, the average number of characters per word. In Equation 2 T and S represent the same quantities they represent in WPM equation.

$$CPM = \frac{|T|-1}{S} * 60 \quad (2)$$

The metrics WPM and CPM do not take into account the errors produced by users (Wobbrock, 2007). Bhattacharya et al. (2008) points the importance of considering users’ mistakes to measure the typing performance of a virtual keyboard.

The metric KSPC (MacKenzie, 2002) proposes a ratio between the number of character transcribed and the amount of keystrokes to produce the text. This measure assesses the number of mistakes, since the calculation considers the number of times that the ‘delete’ key has been pressed along with the other necessary keys to produce the text. Minimizing KSPC means reducing user effort to enter text (Wobbrock, 2007).

$$KSPC = \frac{|IS|}{|T|} \quad (3)$$

In Equation 3 IS is equal to the amount of selected characters, including non-printable keys as the delete key. T represents the amount of transcribed characters.

An extension of KSPC is the metric Gestures by Character (Wobbrock, 2007). This metric considers the amount of shares or actions needed to input a text. The gesture is considered as an atomic action of interaction between the data entry system and the user. In order to use this metric, one must inform what exactly a gesture is to the system. For instance, a gesture can be an eye blink, a breath or a sign issued by the brain. This measure is intended to capture the system accuracy in relation to its communication mechanism (Wobbrock, 2007).

$$GPC = \frac{|IS|}{T} \quad (4)$$

In Equation 4 IS equals the total of gestures performed while typing text and T is equal the amount of transcribed characters.

The user error rate is measured using Equation 5 (Bhattacharya and Laha, 2012), where INF represents the amount of incorrectly entered characters in the transcribed text. IF represents the amount of keystrokes related to non-printable characters, like delete and backspace. Finally, C is the amount of correctly transcribed characters.

$$TER = \frac{INF + IF}{C + INF + IF} * 100 \quad (5)$$

The Minimum String Distance (MSD) is a measure that assesses how accurate is the transcribed string when compared to the original text. Therefore, MSD provides the statistical distance between two strings determining the smallest amount correction operations needed to convert the transcribed string into the original one (Wobbrock, 2007). This distance is calculated using the algorithm proposed by (Soukoreff and MacKenzie, 2001).

Proposal of an assistive, optimized, compact and adaptive virtual keyboard

In this systematic review we identified several data entry optimization techniques and methods, which are the essential characteristics of virtual keyboards. In order to build a new assistive, optimized, compact and adaptive virtual keyboard for Portuguese language, we first define which are the best virtual keyboard features. Then we determine which input optimization techniques must be used in the keyboard. Finally, we discuss which techniques must be used to measure the new virtual keyboard performance.

Figure 8 shows the essential features of the virtual keyboard marked in boldface and what are the available options for each feature. Those options that we have chosen to be in our design proposal are marked with a V. In the following sections we will discuss every option that we chosen.

Amount of symbols per key

According to Molina et al. (2009b) and Topal et al. (2012), ambiguous virtual keyboards are more productive than unambiguous keyboards. Also, Bhattacharya

and Laha (2012) states that one of the main goals of the assistive keyboard is to reduce the typing effort. Then, considering that the ambiguous approach may reduce the interaction between the user and the keyboard (Guerrier et al., 2011), we have chosen to adopt the ambiguous approach.

Disambiguation method

In order to minimize the interaction between the user and the virtual keyboard, we decided to use disambiguation algorithms, once, according to Molina et al. (2009b), the keyboards using this method show better performance than keyboards using multi-tap disambiguation.

Unlike the English language, Portuguese language has vowels that are often accentuated. The disambiguation method will solve this problem by identifying and accentuating words correctly. Thus, using the ambiguous keyboard will reduce the set of possible letters.

Letter sequence

Zhai et al. (2000) reports that using the letter frequencies to distribute them on the keyboard layout is a promising technique to improve typing performance. However, this review has not identified a method to optimally distribute the letters on the keys of an ambiguous virtual keyboard. Distributing the letters in alphabetical order is more used approach to assist disabled users. However, according to Topal et al. (2012), this limits the typing performance.

In order to define which way the letters will be distributed over the keys, we conducted an analysis using digrams on a dictionary of 65,000 words. We identified that the letters distribution in Portuguese words follow a pattern. 85% of the dictionary words

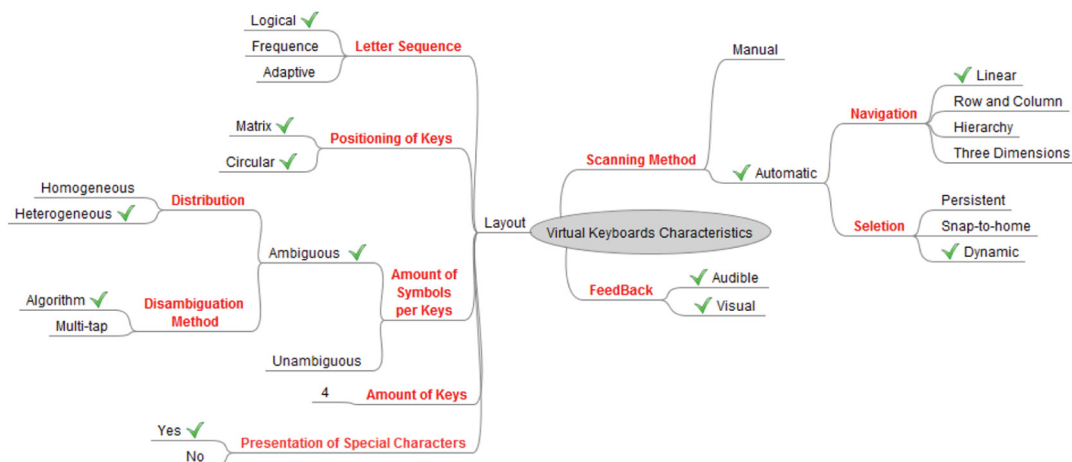


Figure 8. Available features of the virtual keyboard.

in this language are composed of a sequence of interspersed vowels and consonants. Therefore, we decided that we will group the vowels in a single key and to alphabetically distribute the consonants over the remaining keys.

Scanning method

This work proposes a new line scanning method: linear, automatic and dynamic. Automatic Scan tends to minimize the amount of interactions between the user and the virtual keyboard, as this kind of scan requires only the Selection stimulus from the user (Molina et al., 2009b).

On the other hand, some authors like Poláček et al. (2012) and Topal et al. (2012) report that dynamic keyboards do not have a good acceptance among users. Changing the letter sequence leaves the user confused and makes it hard to find the desired character (Poláček et al., 2012). Poláček et al. (2012) suggests using dynamic scanning with static methods and layouts. Therefore, we chose using a method of dynamic scan, instead of snap-to-home or persistent selection.

Scanning techniques are usually very slow as they reach only about 5-20 CPM (Poláček et al., 2012). In order to optimize the scanning method, Miró-Borrás and Bernabeu-Soler (2009) used a linguistic model to dynamically navigate over the keyboard. Based on Miró-Borrás and Bernabeu-Soler (2009), the new virtual keyboard will adopt a dynamic scanning method using the linguistic structure of Portuguese words proposed in this work. Therefore, whenever a consonant is selected, the keyboard focus will automatically move to the vowels key. Also, wherever a vowel is selected, the keyboard focus will move to the consonant keys.

Originally, the scanner delay is set by the caregiver according to the user capacity of interaction. However, we propose to dynamically adjust this delay according to the user's choice speed. Therefore, the faster the keys are selected, the smaller the delay scanner. According to Francis and Johnson (2011), a good delay can be implemented considering a trade off between speed and the amount of user mistakes. Thus, the larger the amount of mistakes, the larger the delay scanner will be.

To measure the number of user's miss type we will consider the amount of times that he will select the backspace button. Thus, every time the user chose this button the scanner delay will be increased. However, if the user types consciously without using this button the scanner time will be decreased.

We are also considering use the algorithm proposed by Ghedira et al. (2009) along with our scanning

technic in future implementations. This algorithm increase and decrease the scanner delay based on the number of times that user make his choice in less then 100 mili seconds. The results obtained by Ghedira et al. (2009) confirms that their algorithms are effective in dynamically adapting a scan speed.

Keys size

Fitts' law determines that the key size and the distance is very important for typing performance. Nonetheless, this rule applies to manual navigation keyboards. The distance between keys or their dimensions are not relevant when building keyboards with scanning methods, as the navigation is not done by the user.

In this case, instead of trying to minimize the movement of fingers according to Fitts' Law, we will try to minimize the scanning cycles applying our scanning method. Because the number of scan cycles are the most time-consuming action in scanning keyboards (Miró-Borrás and Bernabeu-Soler, 2009).

Amount of keys

This work did not identify an ideal amount of keys for ambiguous keyboards. Nevertheless, it is known that lower scanning delays improve typing performance (Molina et al., 2009b). Therefore, to reduce the scanning time, we developed a modular selection such that the keyboard is divided into two distinct sets of keys: one for letters and another set for special characters, punctuation and correction.

The letter set contains 8 keys, being a key to the vowels, three keys for consonants, a key for special characters, a correction key, and two for navigation. A reduced number of keys should moderately reduce the search time by one character (Sharma et al., 2012).

Presentation of special characters

Punctuation and special characters will be displayed by selecting a button labeled "Caracteres Especiais". Once this button is pushed, a matrix will be displayed with special characters as punctuation, numbers and tab characters. The row and column scanning method will allow user to select the desired characters.

If the users want to insert a new word they can select the button labeled "Nova palavra". This button will allow them to insert new words using a row column scanner on a unambiguous keyboard. To correct the text, the user can use the "Correções" key. The navigation button "Salto" will lead the users to the letter set again.

Positioning of keys

According to Hick-Hyman law (Hick, 1952; Hyman, 1953), user reaction time is directly proportional to the number of objects in the interface. To reduce the reaction time, we must reduce the number of keys (Sharma et al., 2012). However, interface designers claim that a keyboard should contain all the necessary keys to compose a text (Sharma et al., 2012). In order to overcome this challenge, this paper proposes a layout that blends the matrix and circular shape keyboards (Figure 9).

The circular layout will show only the key group focused by the scanning. A circle formed by buttons will always be around the letter under insertion. This approach aims to minimize the time spent to view the text and choose the nearby keys.

The matrix arrangement of keys will show all the keyboard keys. This approach was adopted in order to decrease the learning curve for using the keyboard. In addition, it will serve as a guide to show the user all the keyboard options.

The user will be able to configure which arrangement style of keys will be displayed by the keyboard. Thus, patients may select circular or matrix layout, he also can choose both layouts at the same time. This option will be displayed at main menu in “Opções Gerais”.

One detail that must be noticed in Figure 9 is the “Salto” button. This keyboard uses linear scanning for shifting between keys. This scanning style take a

long time to pass through all eight keys. To overcome this issue we add the “Salto” button. This button will allow the user to switch between the two group of keys. The first group of keys contains the letter keys and the other group is composed by special characters and correction buttons.

Feedback

According to Majoranta et al. (2003), the users type faster in the presence of both auditory and visual feedback (compared to their absence). Therefore, the keyboard feedback method will carry out both feedbacks auditory and visual. Every time a user presses a key, a button-pressing sound will be played by the software and the selected key will be highlighted.

Word prediction

Text prediction is one of the most widely used techniques to enhance the text entry rate in message composition systems using virtual keyboards (Sharma et al., 2010). A dictionary containing the words frequency will be used to word prediction. We chose to use the dictionary often, because this technique is easy to implement, allows the inclusion of new words and assists in the implementation of disambiguation methods.

To calculate the frequency of each word, we analyzed the corpus of the Portuguese language in

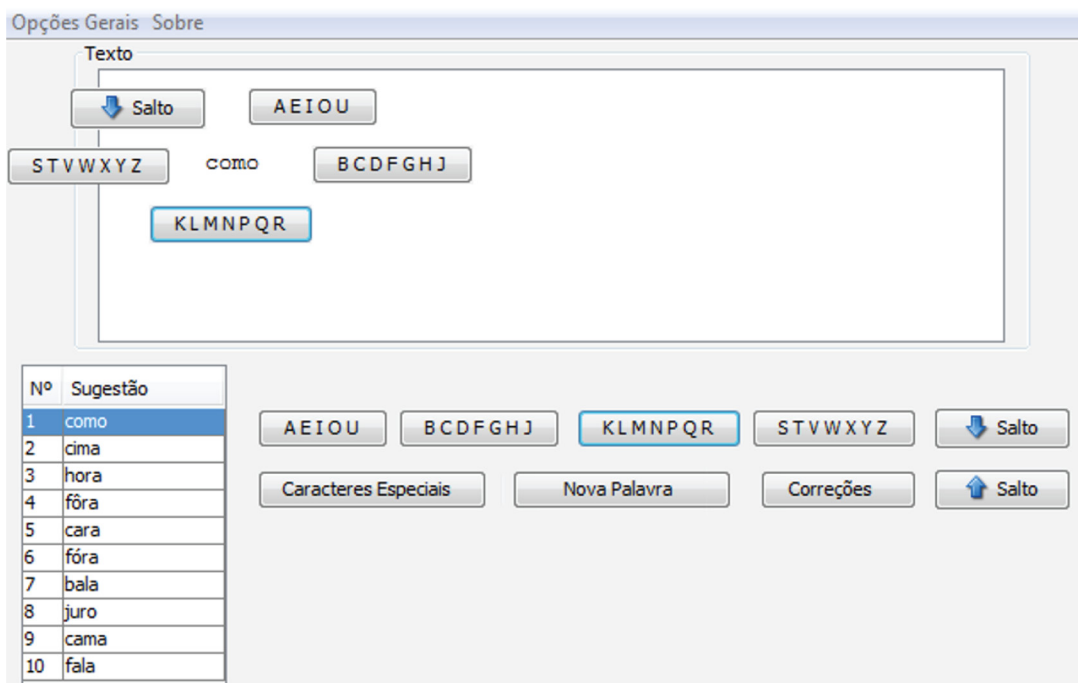


Figure 9. Mixed layout assistive virtual keyboard developed in this work. The circular keyboard involves the word being written. The matrix keyboard is below suggested word lists.

Brazil. This corpus of over one billion words with 3 million distinct words were taken from (Sardinha et al., 2014). The dictionary was loaded with over 70,000 distinct words and their corresponding frequencies.

In the word-level disambiguation mode, the dictionary could be reduced and dynamically adjusted to the user vocabulary in order to minimize the prediction list of words and reduce the position of the desired word in the list (Miró-Borrás and Bernabeu-Soler, 2009).

In order to optimize prediction and to make it adaptive to user while he writes, the keyboard will display three words suggestion lists. The first list will show the words most used by the user. The second list will provide the last word typed by the user. The last list will show the words with highest occurrence in the language corpus. The word frequencies will be dynamically updated during the use of the software. The number of words available in each list may vary from 5 to 15, according to the keyboard configuration.

Character prediction

This review identified some studies that used trigram as linguistic model for character prediction (Janpinijrut et al., 2011b; Phanchaipetch and Nattee, 2012; Sarafis and Markoulidis, 2010) and had got significant results using this model. Therefore, we chosen to use the trigram for predicting the next characters to be entered.

As the keyboard proposed in this paper is ambiguous, the suggestion of characters will be held after the selection of a button. Once the user has made a choice, the system suggests one character with a high probability of being the next one to be entered. The user can accept the system suggestion or wait for the next character provided by the scanning method.

Disambiguation algorithm

According to Silfverberg et al. (2000), using disambiguation algorithm (T9) allows for larger rate of words per minute than using multi-tap. Besides, it minimizes the text entry effort (Harbusch and Kuhn, 2003). Thus, in order to perform words disambiguation, we propose using the TNK algorithm (Molina et al., 2009b). TNK is easier to implement and shows a good typing performance. Besides, it is better suited to the method of word prediction adopted in this paper.

Error handling

Several techniques increase the typing performance of virtual keyboards at the expense of accuracy. The work of Bhattacharya et al. (2008) reports a great number of works focused on increasing the typing performance but with no concern on reducing the amount of typing

errors of the users. Then, we propose using phonetic and word similarity algorithms in order to minimize the amount of errors entered by users.

Phonological search algorithms index the words according to a phonetic code. Words with similar phonetic are easily identified. This method helps reducing the errors due to the user ignore the correct writing of a certain word. The phonetic algorithm will be based on Philips (2015) and on the work of Vasilévski (2008).

Similarity search algorithms identify the strings on a dictionary that best match a query string according to a given threshold of coincident characters (Okazaki and Tsujii, 2010). This algorithm will be used to reduce errors due to mistyped, excessive or omitted characters.

Adaptation of the virtual keyboard to user

The assistive virtual keyboard will allow adding new words on the dictionary from the user vocabulary, as well as changing the number of keys and the scanning delay. The addition of new words and vocabulary recognition will be made during text entry, while the remaining adjustments must be made offline.

Virtual keyboard measurements

According to Miró-Borrás and Bernabeu-Soler (2009) is difficult to assign differences between keyboard interfaces, because of they employ several scanners, different key types as well as various sizes of prediction dictionaries. Consequently, data available from scanning systems are quite inaccurate and hard to be compared between keyboards (Miró-Borrás and Bernabeu-Soler, 2009).

Despite these difficulties, most studies use WPM to compare virtual keyboard typing performances (Wobbrock, 2007). This can be seen in Table 3. Therefore, we decided to use WPM to measure keyboard performance. However, as WPM does not measure the amount of user mistakes, user's errors should be rated in order to measure the efficacy of a design (Ghosh, 2011). Then, in order to assess the amount of user mistakes, we chose the measures TER and KBPS. The metric shortest distance between strings (MSD) does not suit this keyboard because the user can only enter words that are contained in the dictionary.

Conclusion

This systematic review focused on works on virtual keyboards that were published between the years 2009 and 2014. This study identified more than 250 publications related to data entry using virtual

keyboards. Out of them, 52 publications were analyzed thoroughly in order to answer the search questions prepared according to our protocol.

At first, the search questions were answered in a broad way and, then, detailed in Section *Data Analysis*. We identified 50 papers related to essential features to build a virtual keyboard. We found 30 works with techniques usually employed to optimize the typing performance of those keyboards and 25 out of them were finally detailed in terms of input speed.

Based on the review, this paper proposed a new approach to assistive virtual keyboard that aims to optimize the performance of typing, reduce the typing errors and minimize the effort required to input text by patients with severe motor disabilities.

Entering performance will be optimized through the prediction of letters and words. The phonetic and similarity algorithms will help reducing user's mistakes. Finally, a new dynamic scanning method has been proposed to both decrease the typing effort and increase the typing speed.

The keyboard we proposed in this paper is in the first phase of creating. The current interface is a start prototype and a better ergonomic release suitable for use in a real situation is being designed. For future work we will implement and test this new approach with people without any motor restriction. Then, the resulting virtual keyboard will be validated with real users.

References

- Al-Abdullatif A, Al-Negheimish H, Al-Mofeez L, Al-Khalifa N, AlAndas L, Al-Wabil A. Mind-controlled augmentative and alternative communication for people with severe motor disabilities. In: Serhani MA, Shuaib K, Mansoor W, editors. *Innovations in Information Technology. IIT 2013: Proceedings of the 9th International Conference on Innovations in Information Technology*; 2013 Mar 17-19; Abu Dhabi, Al-Ain. Hoboken: IEEE; 2013. p. 107-112. <http://dx.doi.org/10.1109/Innovations.2013.6544402>.
- An KO, Kim DH, Kim J. Development of a virtual keyboard system using a bio-signal interface and preliminary usability test: human-computer interaction towards intelligent and implicit interaction. New York: Springer; 2013. p. 3-9. http://dx.doi.org/10.1007/978-3-642-39342-6_1.
- Arif AS, Stuerzlinger W. Evaluation of a new error prevention technique for mobile touchscreen text entry. In: Shen H, Smith R, Paay J, Calder P, Wyeld T, editors. *Augmentation, Application, Innovation, Collaboration. OzCHI 2013: Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration*; 2013 Nov 25-29; Adelaide, Australia. New York: ACM; 2013. p. 397-400. <http://dx.doi.org/10.1145/2541016.2541063>.
- Baljko M, Tam A. Indirect text entry using one or two keys. In: Keates S, editor. *Computers and Accessibility. Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility - Assets '06*; 2006 Oct 23-25; Portland, USA. New York: ACM; 2006. p. 18. <http://dx.doi.org/10.1145/1168987.1168992>.
- Bhattacharya S, Basu A, Samanta D. Computational modeling of user errors for the design of virtual scanning keyboards. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2008; 16(4):400-9. <http://dx.doi.org/10.1109/TNSRE.2008.925073>. PMID:18701386.
- Bhattacharya S, Laha S. Bengali text input interface design for mobile devices. *Universal Access in the Information Society*. 2012; 12(4):441-51. <http://dx.doi.org/10.1007/s10209-012-0280-1>.
- Biolchini JC, Mian PG, Natali ACC, Conte TU, Travassos GH. Scientific research ontology to support systematic review in software engineering. *Advanced Engineering Informatics*. 2007; 21(2):133-51. <http://dx.doi.org/10.1016/j.aei.2006.11.006>.
- Biswas P, Samanta D. Friend: a communication aid for persons with disabilities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. 2008; 16(2):205-9. <http://dx.doi.org/10.1109/TNSRE.2008.917305>. PMID:18403290.
- Blain S, Mihailidis A, Chau T. Assessing the potential of electrodermal activity as an alternative access pathway. *Medical engineering & physics*. 2008; 30(4):498-505.
- Burgbacher U, Hinrichs K. An implicit author verification system for text messages based on gesture typing biometrics. In: Jones M, Palanque P, editors. *Conference on Human Factors in Computing Systems. CHI 2014: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*; 2014 Apr 26-May 01; Toronto, Canada. New York: ACM; 2014. p. 2951-4. <http://dx.doi.org/10.1145/2556288.2557346>.
- Cipresso P, Meriggi P, Carelli L, Solca F, Meazzi D, Poletti B, Lulé D, Ludolph AC, Riva G, Silani V. The combined use of Brain Computer Interface and eye-tracking technology for cognitive assessment in Amyotrophic Lateral Sclerosis. In: Chlamtac I, editor. *Pervasive Health Conference. PervasiveHealth 2011: Proceedings of the 5th International Conference on Pervasive Computing Technologies for Healthcare*; 2011 May 23-26; Dublin, Ireland. Hoboken: IEEE; 2011. p. 320-4. <http://dx.doi.org/10.4108/icst.pervasivehealth.2011.246018>.
- Colker R. Americans with disabilities act: a windfall for defendants. *Harvard Civil Rights-Civil Liberties Law Review*. 1999; 34-99.
- Cook AM, Polgar JM. *Assistive technologies: principles and practice*. 4th ed. Missouri: Mosby; 2014.
- Corley J, Heaton D, Gray J, Carver JC, Smith R. Brain-Computer Interface virtual keyboard for accessibility. In: Merrell R, Shin DG, Hamza MH, editors. *Imaging and Signal Processing in Health Care and Technology. Human-Computer*; 2012 May 14-16; Baltimore, USA. Calgary: ACTAPRESS; 2012. <http://dx.doi.org/10.2316/P.2012.772-036>.

- Doval G, Ma F, Pousada Carballo JM, Vez Jeremias JM. TICTAC: Information and communication technologies for augmentative communication boards. In: Castro M, Tovar E, Auer ME, editors. Education Engineering. IEEE EDUCON 2010 Proceedings; 2010 Apr 14-16; Madrid, Spain. Hoboken: IEEE; 2010. p. 1783-7.
- Eklund A, Andersson M, Ohlsson H, Ynnerman A, Knutsson H. A Brain Computer Interface for communication using real-time fMRI. In: Ercil A, editors. International Conference on Pattern Recognition. ICPR 2010: Proceedings of the 20th International Conference on Pattern Recognition; 2010 Aug 23-26; Istanbul, Turkey. Hoboken: IEEE; 2010. p. 3665-9. <http://dx.doi.org/10.1109/ICPR.2010.894>.
- Faraj K, Mojahid M, Vigouroux N. Bigkey: a virtual keyboard for mobile devices. In: Jacko JA, editor. Human-computer interaction ambient, ubiquitous and intelligent interaction. Berlin: Springer-Verlag; 2009a. p. 3-10. http://dx.doi.org/10.1007/978-3-642-02580-8_1.
- Faraj K, Vigouroux N, Mojahid M. SmartKey: a multi purpose target expansion based virtual keyboard. In: Trewin S, editor. The 11th International ACM SIGACCESS Conference on Computers and Accessibility. ASSETS 2009: Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility; 2009 Oct 25-28; Pittsburgh, USA. New York: ACM; 2009b. p. 251-2. <http://dx.doi.org/10.1145/1639642.1639700>.
- Faraj K, Mojahid M, Vigouroux N. 3DKey: an accordion-folding based virtual keyboard for small screen. In: Gross T, Gulliksen J, Kotzé P, Oestreicher L, Palanque P, Prates RO, Winckler M, editros. Human-computer interaction – INTERACT. Berlin: Springer-Verlag; 2009c. p. 634-44. http://dx.doi.org/10.1007/978-3-642-03655-2_71.
- Felzer T, Rinderknecht S. 3dScan: an environment control system supporting persons with severe motor impairments. In: Trewin S, editor. The 11th International ACM SIGACCESS Conference on Computers and Accessibility. ASSETS 2009: Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility; 2009 Oct 25-28; Pittsburgh, USA. New York: ACM; 2009. p. 213-4. <http://dx.doi.org/10.1145/1639642.1639681>.
- Fitts PM. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*. 1954; 47(6):381-91. <http://dx.doi.org/10.1037/h0055392>. PMID:13174710.
- Francis G, Johnson E. Speed - Accuracy tradeoffs in specialized keyboards. *Journal of Human Computer Studies*. 2011; 69(7-8):526-38. <http://dx.doi.org/10.1016/j.ijhcs.2011.04.002>.
- Fu YF, Ho CS. A fast text-based communication system for handicapped aphasiacs. In: Chen YW, Pan JS, editors. Intelligent Information Hiding and Multimedia Signal Processing. IHH-MSP 2009: Proceedings of the Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing; 2009 Sep 12-14; Kyoto, Japan. Hoboken: IEEE; 2009. p. 583-94. <http://dx.doi.org/10.1109/IHH-MSP.2009.251>.
- Galvão TA Fo, Garcia JCD. Pesquisa nacional de tecnologia assistiva. São Paulo: MCTI-SECIS; 2012.
- Garay-Vitoria N, Abascal J. Text prediction systems: a survey. *Universal Access in the Information Society*. 2006; 4(3):188-203. <http://dx.doi.org/10.1007/s10209-005-0005-9>.
- Gelormini D, Bishop B. Optimizing the android virtual keyboard: A study of user experience. In: Multimedia and Expo Workshops. ICMEW 2013: IEEE International Conference on Multimedia and Expo Workshops; 2013 Jul 15-19; San Jose, USA. Hoboken: IEEE; 2013. p. 1-4. <http://dx.doi.org/10.1109/ICMEW.2013.6618310>.
- Ghedira S, Pino P, Bourhis G. Conception and experimentation of a communication device with adaptive scanning. *ACM Transactions on Accessible Computing*. 2009; 1(3):14. <http://dx.doi.org/10.1145/1497302.1497304>.
- Ghosh S, Sarcar S, Sharma MK, Samanta D. Effective virtual keyboard design with size and space adaptation. In: Kumar A, Sheet D, Sarcar S, editors. Students' Technology Symposium. TechSym 2010: Proceedings of the 2010 IEEE Students' Technology Symposium; 2010 Apr 3-4; Kharagpur, India. Hoboken: IEEE; 2010. p. 262-7.
- Ghosh S. Designing an efficient virtual keyboard for text composition in Bengali. In: Tripathi S, editor. Human Computer Interaction. India HCI 2011: Proceedings of the 3rd International Conference on Human Computer Interaction; 2011 Apr 07-10; Bangalore, India. New York: ACM; 2011. p. 84-7. <http://dx.doi.org/10.1145/2407796.2407809>.
- Gizatdinova Y, Spakov O, Surakka V. Comparison of video-based pointing and selection techniques for hands-free text entry. In: Tortora G, Levialdi S, Tucci M. Advanced Visual Interfaces. AVI 2012: Proceedings of the International Working Conference on Advanced Visual Interfaces; 2012 May 22-25, Capri Island, Italy. New York: ACM; 2012a. p. 132. <http://dx.doi.org/10.1145/2254556.2254582>.
- Gizatdinova Y, Spakov O, Surakka V. Face typing: Vision-based perceptual interface for hands-free text entry with a scrollable virtual keyboard. In: Applications of Computer Vision. WACV 2012: IEEE Workshop on the Applications of Computer Vision; 2012 Jan 9-11; Breckenridge, USA. Hoboken: IEEE; 2012b. p. 81-7. <http://dx.doi.org/10.1109/WACV.2012.6162997>.
- Goodman J, Venolia G, Steury K, Parker C. Language modeling for soft keyboards. In: Hammond K, editor. Intelligent User Interfaces. IUI 2002: Proceedings of the 7th International Conference on Intelligent User Interfaces; 2002 Jan 13-16; San Francisco, USA. New York: ACM; 2002. p. 194. <http://dx.doi.org/10.1145/502716.502753>.
- Grover DL, King MT, Kushler CA. Reduced keyboard disambiguating computer. Google Patents; US Patent 5,818,437. 1998.
- Guerrier Y, Baas M, Kolski C, Poirier F. Comparative study between AZERTY-type and K-Hermes virtual keyboards dedicated to users with cerebral palsy. In: Stephanidis C, editor. Universal access in human-computer interaction users diversity. Berlin: Springer-Verlag; 2011. p. 310-9.. http://dx.doi.org/10.1007/978-3-642-21663-3_33.

- Guerrier Y, Kolski C, Poirier F. Towards a communication system for people with athetoid cerebral palsy: human-computer interaction. INTERACT 2013: International Conference on Human-Computer Interaction; 2013 Sep 2-6; Cape Town, South Africa. Berlin: Springer; 2013. p. 681-8. http://dx.doi.org/10.1007/978-3-642-40498-6_61.
- Harbusch K, Kuhn M. Towards an adaptive communication aid with text input from ambiguous keyboards. In: Copestake A, editor. European Chapter of the Association for Computational Linguistics. EACL 2003: Proceedings of the 10th Conference on European chapter of the Association for Computational Linguistics; Stroudsburg: Association for Computational Linguistics; 2003. p. 207-10, vol. 2. <http://dx.doi.org/10.3115/1067737.1067786>.
- Hick WE. On the rate of gain of information. The Quarterly Journal of Experimental Psychology. 1952; 4(1):11-26. <http://dx.doi.org/10.1080/17470215208416600>.
- Hoste L, Signer B. SpeeG2: a speech-and gesture-based interface for efficient controller-free text input. In: Epps J, Chen F, Oviatt S, Mase K, editors. Multimodal Interaction. Proceedings of the 15th ACM on International Conference on Multimodal Interaction; 2013 Dec 9-13; Sydney, Australia. New York: ACM; 2013. p. 213-20. <http://dx.doi.org/10.1145/2522848.2522861>.
- Hyman R. Stimulus information as a determinant of reaction time. Journal of Experimental Psychology. 1953; 45(3):188-96. <http://dx.doi.org/10.1037/h0056940>. PMID:13052851.
- Jain S, Bhattacharya S. Virtual keyboard layout optimization. In: Kumar A, Sheet D, Sarcar S, editors. Students' Technology Symposium. TechSym 2010: Proceedings of the 2010 IEEE Students' Technology Symposium; 2010 Apr 3-4; Kharagpur, India. Hoboken: IEEE; 2010. p. 312-7.
- Janpinijrut S, Kayasith P, Nattee C, Okumura M. Vowel-separated Layout: A Thai touchscreen keyboard for people with hand movement disability. In: Thajchayapong P, editor. Rehabilitation Engineering & Assistive Technology. i-CREAtE 2011: Proceedings of the 5th International Conference on Rehabilitation Engineering & Assistive Technology; 2011 Jul 21-23; Bangkok, Thailand. New York: ACM; 2011a. p. 1-4.
- Janpinijrut S, Nattee C, Kayasith P. An approach for improving Thai text entry on touch screen mobile phones based on distance and statistical language model. In: Kunifuji S, Papadopoulos GA, Skulimowski AMJ, Kacprzyk J, editors. Knowledge, information, and creativity support systems. New York: Springer; 2011b. p. 44-55. http://dx.doi.org/10.1007/978-3-642-24788-0_5.
- Joshi A, Dalvi G, Joshi M, Rashinkar P, Sarangdhar A. Design and evaluation of Devanagari virtual keyboards for touch screen mobile phones. In: Bylund M, editor. Human Computer Interaction with Mobile Devices and Services. MobileHCI 2011: Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services; 2011 Aug 30-Sep 02; Stockholm, Sweden. New York: ACM; 2011. p. 323. <http://dx.doi.org/10.1145/2037373.2037422>.
- Kim JH, Aulck L, Bartha MC, Harper CA, Johnson PW. Differences in typing forces, muscle activity, comfort, and typing performance among virtual, notebook, and desktop keyboards. Applied Ergonomics. 2014; 45(6):1406-13. <http://dx.doi.org/10.1016/j.apergo.2014.04.001>. PMID:24856862.
- Kitchenham B, Brereton P. A systematic review of systematic review process research in software engineering. Information and Software Technology. 2013; 55(12):2049-75. <http://dx.doi.org/10.1016/j.infsof.2013.07.010>.
- Klima M, Slovacek V. Vector keyboard for touch screen devices. In: Karsh BT, editor. Ergonomics and health aspects of work with computers. Berlin: Springer-Verlag; 2009. p. 250-6. http://dx.doi.org/10.1007/978-3-642-02731-4_29.
- Krejcar O. Handicapped people virtual keyboard controlled by head motion detection. In: Ultra Modern Telecommunications and Control Systems. ICUMT 2011: 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops; 2011 Oct 5-7; Budapest, Hungary. Hoboken: IEEE; 2011. p. 1-7.
- Kwon S, Lee D, Chung MK. Effect of key size and activation area on the performance of a regional error correction method in a touchscreen QWERTY keyboard. International Journal of Industrial Ergonomics. 2009; 39(5):888-93. <http://dx.doi.org/10.1016/j.ergon.2009.02.013>.
- MacKenzie IS, Kober H, Smith D, Jones T, Skepner E. LetterWise: prefix-based disambiguation for mobile text input. In: Marks J, editor. User Interface Software and Technology. UIST 2001: Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology; 2001 Nov 11-14; Orlando, USA. New York: ACM; 2001. p. 111-20.
- MacKenzie IS, Tanaka-Ishii K. Text entry systems: mobility, accessibility, universality. San Francisco: Morgan Kaufmann Publishers Inc.; 2007.
- MacKenzie IS, Zhang SX. The design and evaluation of a high performance soft keyboard. In: Altom M, Williams M, editors. Human Factors in Computing Systems. CHI 1999: Proceedings of the ACM CHI 99 Human Factors in Computing Systems Conference; 1999 May 15-20; Pittsburgh, USA. New York: ACM; 1999. p. 25-31. <http://dx.doi.org/10.1145/302979.302983>.
- MacKenzie IS. KSPC (Keystrokes Per Character) as a characteristic of text entry techniques. In: Paternò F, editor. Human Computer Interaction with Mobile Devices. MobileHCI 2002: Proceedings of the 4th International Symposium on Mobile Human-Computer Interaction; 2002 Sep 18-20; Pisa, Italy. London: Springer; 2002. p. 195-210. http://dx.doi.org/10.1007/3-540-45756-9_16.
- Majaranta P, MacKenzie IS, Aula A, Raiha KJ. Auditory and visual feedback during eye typing. In: Cockton G, Korhonen P, editors. Human Factors in Computing Systems. CHI 2003: Extended Abstracts on Human Factors in Computing Systems; 2003 Apr 05-10; Lauderdale, USA. New York: ACM; 2003. p. 766-7. <http://dx.doi.org/10.1145/765891.765979>.
- Mason C, Chinn KM. Augmentative alternative communication. In: Peterson P, Baker E, McGaw B, editors. International

- encyclopedia of education. Philadelphia: Elsevier; 2010. p. 544-9. <http://dx.doi.org/10.1016/B978-0-08-044894-7.01133-7>.
- Mele ML, Federici S. A psychotechnological review on eye-tracking systems: towards user experience. *Disability and Rehabilitation: Assistive Technology*. 2012; 7(4):261-81.
- Merino M, Gómez I, Molina AJ, Guzman K. Assessment of biosignals for managing a virtual keyboard. In: Miesenberger K, Karshmer A, Penaz P, Zagler W, editors. *Computers helping people with special needs*. Heidelberg: Springer; 2012. p. 331-7. http://dx.doi.org/10.1007/978-3-642-31534-3_50.
- Merino M, Gómez I, Rivera O, Molina AJ. Customizable software interface for monitoring applications. In: Miesenberger K, Karshmer A, Penaz P, Zagler W, editors. *Computers helping people with special needs*. Heidelberg: Springer; 2010. p. 147-53. http://dx.doi.org/10.1007/978-3-642-14097-6_25.
- Millet B, Asfour S, Lewis JR. Selection-based virtual keyboard prototypes and data collection application. *Behavior Research Methods*. 2009; 41(3):951-6. <http://dx.doi.org/10.3758/BRM.41.3.951>. PMID:19587212.
- Miró J, Bernabeu P. Text entry system based on a minimal scan matrix for severely physically handicapped people. In: Miesenberger K, Klaus J, Zagler W, Karshmer A, editors. *Computers helping people with special needs*. Heidelberg: Springer; 2008. p. 1216-9. *Lecture Notes in Computer Science*, vol. 5105. http://dx.doi.org/10.1007/978-3-540-70540-6_183.
- Miró-Borrás J, Bernabeu-Soler P. Text entry in the e-commerce age: two proposals for the severely handicapped. *Journal of Theoretical and Applied Electronic Commerce Research*. 2009; 4(1):101-12.
- Molina A, Rivera O, Gómez I, Sánchez G. Evaluation of unambiguous virtual keyboards with character prediction. In: Emiliani PL, Burzagli L, Como A, Gabbanini F, Salminen AL, editors. *Assistive technology from adapted equipment to inclusive environments*. Clifton: IOS Press; 2009a. p. 144-9.
- Molina AJ, Rivera O, Gómez I. Measuring performance of virtual keyboards based on cyclic scanning. In: Calinescu R, Liberal F, Marin M, Herrero LP, Turro C, Popescu M, editors. *Autonomic and Autonomous Systems*. ICAS 2009: Fifth International Conference on Autonomic and Autonomous Systems; 2009 Apr 20-25; Valencia, Spain. Hoboken: IEEE; 2009b. p. 174-8. <http://dx.doi.org/10.1109/ICAS.2009.35>.
- Nicolau H, Guerreiro T, Lucas D, Jorge J. Mobile text-entry and visual demands: reusing and optimizing current solutions. *Universal Access in the Information Society*. 2013; 13(3):291-301. <http://dx.doi.org/10.1007/s10209-013-0319-y>.
- Okazaki N, Tsujii J. Simple and Efficient Algorithm for approximate dictionary matching. In: Joshi AK, editor. *Computational Linguistics*. COLING 2010: Proceedings of the 23rd International Conference on Computational Linguistics; 2010 Aug 23-27; Beijing, China. Stroudsburg: Association for Computational Linguistics; 2010. p. 851-9.
- Orhan U, Hild KE, Erdogmus D, Roark B, Oken B, Fried-Oken M. RSVP keyboard: an EEG based typing interface. In: Arakawa K, Yamada I, Ichige K, editors. *Acoustics, Speech and Signal Processing*. ICASSP 2012: IEEE International Conference on Acoustics, Speech and Signal Processing; 2012 Mar 25-30; Kyoto, Japan. Hoboken: IEEE; 2012. p. 645-8. <http://dx.doi.org/10.1109/ICASSP.2012.6287966>.
- Panwar P, Sarcar S, Samanta D. EyeBoard: a fast and accurate eye gaze based text entry system. In: Samanta D, editor. *Intelligent Human Computer Interaction*. IHCI 2012: 4th International Conference on Intelligent Human Computer Interaction; 2012 Dec 27-29; Kharagpur, India. Hoboken: IEEE; 2012. p. 1-8. <http://dx.doi.org/10.1109/IHCI.2012.6481793>.
- Park SW, Yim YL, Yi SH, Kim HY, Jung SM. Augmentative and alternative communication training using eye blink switch for locked-in syndrome patient. *Annals of Rehabilitation Medicine*. 2012; 36(2):268-72. <http://dx.doi.org/10.5535/arm.2012.36.2.268>. PMID:22639753.
- Phanchaipetch T, Nattee C. An approach for improving Thai text entry on touch screen mobile devices based on bivariate normal distribution and probabilistic language model. In: Pan JS, Chen SM, Nguyen NT, editors. *Intelligent Information and Database Systems*. Berlin: Springer; 2012. p. 187-96. http://dx.doi.org/10.1007/978-3-642-28493-9_21.
- Philips L. Lawrence Philips' metaphone algorithm [internet]. 2015 [cited 2015 Mar 05]. Available from: <http://aspell.net/metaphone/>
- Poláček O, Míkovec Z, Slavík P. Predictive scanning keyboard operated by hissing. In: Hellmich C, Hamza MH, Simsik D, editors. *Assistive Technologies*. IASTED 2012: Proceedings of the 2nd IASTED International Conference Assistive Technologies; 2012 Feb 15-17; Innsbruck, Austria. Calgary: ACTA Press; 2012. p. 862-9.
- Prabhu V, Prasad G. Designing a virtual keyboard with multi-modal access for people with disabilities. In: Abraham A, Agrawal D, Abraham S, Corchado E, Bajaj P, Tomar G, editors. *Information and Communication Technologies*. WICT 2011: World Congress on Information and Communication Technologies; 2011 Dec 11-14; Mumbai, India. Hoboken: IEEE; 2011. p. 1133-8. <http://dx.doi.org/10.1109/WICT.2011.6141407>.
- Rivera O, Molina A, Gómez IM, Merino M. A study of two-inputs scanning methods to enhance the communication rate. In: Emiliani PL, Burzagli L, Como A, Gabbanini F, Salminen AL, editors. *Assistive Technology from Adapted Equipment to Inclusive Environments*: AAATE 2009. Clifton: IOS Press; 2009. p. 132-72.
- Sarafis I, Markoulidis A. Modern greek language frequency counts for text entry devices. In: Blashki K, editor. *Interfaces and Human Computer Interaction*. IHCI 2010: Proceedings of the IADIS International Conference on Interfaces and Human Computer Interaction; 2010 Jul 26-30; Freiburg, Germany. Lisboa: IADIS; 2010. p. 113-120.
- Sarcar S, Ghosh S, Saha PK, Samanta D. Virtual keyboard design: state of the arts and research issues. In: Kumar

- A, Sheet D, Sarcar S, editors. Students' Technology Symposium. TechSym 2010: Proceedings of the 2010 IEEE Students' Technology Symposium; 2010 Apr 3-4; Kharagpur, India. Hoboken: IEEE; 2010. p. 289-99. <http://dx.doi.org/10.1109/TECHSYM.2010.5469165>.
- Sardinha T, Moreira Filho J, Alambert E. Corpus brasileiro [internet]. 2014 [cited 2015 Abr 10] Available from: <http://corpusbrasileiro.pucsp.br/>
- Schadle I. Sibyl: AAC system using NLP techniques. In: Miesenberger K, Klaus J, Zagler W, Burger D, editors. Computers helping people with special needs. Berlin: Springer; 2004. p. 1009-15. http://dx.doi.org/10.1007/978-3-540-27817-7_149.
- Schalk G, Brunner P, Gerhardt LA, Bischof H, Wolpaw JR. Brain-computer interfaces (BCIs): detection instead of classification. *Journal of Neuroscience Methods*. 2008; 167(1):51-62.
- Sharma MK, Dey S, Saha PK, Samanta D, Kharagpur IIT, Bengal W. Parameters effecting the predictive virtual keyboard. In: Kumar A, Sheet D, Sarcar S, editors. Students' Technology Symposium. TechSym 2010: Proceedings of the 2010 IEEE Students' Technology Symposium; 2010 Apr 3-4; Kharagpur, India. Hoboken: IEEE; 2010. p. 268-75. <http://dx.doi.org/10.1109/TECHSYM.2010.5469160>.
- Sharma MK, Sarcar S, Saha PK, Samanta D. Visual clue: an approach to predict and highlight next character. In: Samanta D, editor. Intelligent Human Computer Interaction. IHCI 2012: 4th International Conference on Intelligent Human Computer Interaction; 2012 Dec 27-29; Kharagpur, India. Hoboken: IEEE; 2012. p. 1-7. <http://dx.doi.org/10.1109/IHCI.2012.6481820>.
- Silfverberg M, MacKenzie IS, Korhonen P. Predicting text entry speed on mobile phones. In: Mackay WE, editor. Human Factors in Computing Systems. CHI 2013: Proceedings of the SIGCHI conference on Human factors in computing systems; 2013 Apr 27-May 27; Paris, France. New York: ACM; 2000. p. 9-16. <http://dx.doi.org/10.1145/332040.332044>.
- Simathamand J, Piromsopa K. Performance optimization of Thai virtual keyboard for social networking. In: Jarensri L, editor. Computer Science and Software Engineering. JCSSE 2011: The 8th Joint Conference on Computer Science and Software Engineering; 2011 May 11-13; Nakhon Pathom, Thailand. Hoboken: IEEE; 2011. p. 210-3. <http://dx.doi.org/10.1109/JCSSE.2011.5930122>.
- Soukoreff RW, MacKenzie IS. Measuring errors in text entry tasks: An application of the Levenshtein string distance statistic. In: Tremaine MM, editor. Human Factors in Computing Systems. CHI 2001: Extended Abstracts on Human Factors in Computing Systems; 2001 Mar 31-Apr 05; Seattle, USA. New York: ACM; 2001. p. 319-20. <http://dx.doi.org/10.1145/634067.634256>.
- Steriadis CE, Constantinou P. Designing human-computer interfaces for quadriplegic people. *ACM Transactions on Computer-Human Interaction*. 2003; 10(2):87-118. <http://dx.doi.org/10.1145/772047.772049>.
- Tanaka-Ishii K, Inutsuka Y, Takeichi M. Entering text with a four-button device. In: Computational Linguistics. COLING 2002: Proceedings of the 19th International Conference on Computational Linguistics; 2002 Aug 24-Sep 01; Taipei, Taiwan. New York: ACM; 2002. p. 1-7. <http://dx.doi.org/10.3115/1072228.1072377>.
- Topal C, Benligiray B, Akinlar C. On the efficiency issues of virtual keyboard design. In: Leong JCY, Siegel M, Pena FL, editors. Virtual Environments Human-Computer Interfaces and Measurement Systems. VECIMS 2012: IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems; 2012 Jul 02-04; Tiajin, China. Hoboken: IEEE; 2012. p. 38-42. <http://dx.doi.org/10.1109/VECIMS.2012.6273205>.
- Truong CT, Nguyen-Huynh DH, Tran MT, Duong AD. Collaborative smart virtual keyboard with word predicting function. In: Kurosu M, editor. Human-Computer Interaction. Interaction Modalities and Techniques. Berlin: Springer; 2013. p. 513-22. http://dx.doi.org/10.1007/978-3-642-39330-3_55.
- Usakli AB, Gurkan S. Design of a novel efficient human-computer interface: an electrooculogram based virtual keyboard. *IEEE Transactions on Instrumentation and Measurement*. 2010; 59(8):2099-108. <http://dx.doi.org/10.1109/TIM.2009.2030923>.
- Varcholik PD, LaViola JJ Jr, Hughes CE. Establishing a baseline for text entry for a multi-touch virtual keyboard. *International Journal of Human-Computer Studies*. 2012; 70(10):657-72. <http://dx.doi.org/10.1016/j.ijhcs.2012.05.007>.
- Vasiléviski V. Construção de um sistema computacional para suporte à pesquisa em fonologia do português do Brasil [thesis]. Santa Catarina: Universidade Federal de Santa Catarina; 2008.
- Vella F, Vigouroux N. Validation of Fitts law adapted for upper limb motor impairment on software keyboards. In: Azevedo L, editor. Advancement of Assistive Technology in Europe. AATE 2013: European Conference for the Advancement of Assistive Technology in Europe; 2013 Sep 19-22; Vilamoura, Portugal. Clifton: IOS Press; 2013. p. 500-5.
- Wilkinson KM, Hennig S. The state of research and practice in augmentative and alternative communication for children with developmental/intellectual disabilities. *Mental Retardation and Developmental Disabilities Research Reviews*. 2007; 13(1):58-69. <http://dx.doi.org/10.1002/mrdd.20133>. PMID:17326111.
- Wobbrock JO. Measures of text entry performance. In: MacKenzie IS, Tanaka-Ishii K, editors. Text entry systems. Burlington: Morgan Kaufmann; 2007. p. 47-74. <http://dx.doi.org/10.1016/B978-012373591-1/50003-6>.
- Wu FG, Huang YC, Wu ML. New chording text entry methods combining physical and virtual buttons on a mobile phone. *Applied Ergonomics*. 2014; 45(4):825-32. <http://dx.doi.org/10.1016/j.apergo.2013.10.011>. PMID:24263072.

Yamada H. A historical study of typewriters and typing methods, from the position of planning japanese parallels. *Journal of Information Processing*. 1980; 2(4):175-202.

Yang SW, Lin CS, Lin SK, Lee CH. Design of virtual keyboard using blink control method for the severely disabled. *Computer Methods and Programs in Biomedicine*. 2013; 111(2):410-8. <http://dx.doi.org/10.1016/j.cmpb.2013.04.012>. PMID:23702128.

Zhai S, Hunter M, Smith B. The metropolis keyboard - an exploration of quantitative techniques for virtual keyboard design. In: Ackerman M, Edwards K, editors. *User interface software and technology. UIST 2000: Proceedings of the 13th annual ACM symposium on User interface software and technology*; 2000 Nov 06-08; San Diego, USA. New York: ACM; 2000. p. 119-28. <http://dx.doi.org/10.1145/354401.354424>.

Authors

Renato de Sousa Gomide^{1*}, Luiz Fernando Batista Loja¹, Rodrigo Pinto Lemos², Edna Lúcia Flôres¹, Francisco Ramos Melo³, Ricardo Antonio Gonçalves Teixeira²

¹ Faculty of Electrical Engineering, Universidade Federal de Uberlândia – UFU, Av. João Naves de Avila, 212, Santa Mônica, CEP 38408-100, Uberlândia, MG, Brazil.

² School of Electrical Engineering, Mechatronics and Computing, Universidade Federal de Goiás – UFG, Goiânia, GO, Brazil.

³ Information Systems, Universidade Estadual de Goiás – UEG, Anápolis, GO, Brazil.