



HAL
open science

KeyGlasses: semi-transparent keys on soft keyboard

Mathieu Raynal

► **To cite this version:**

Mathieu Raynal. KeyGlasses: semi-transparent keys on soft keyboard. 16th ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2014), Oct 2014, Rochester, United States. pp.347-349. hal-02089300

HAL Id: hal-02089300

<https://hal.science/hal-02089300>

Submitted on 3 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in:

<http://oatao.univ-toulouse.fr/22719>

Official URL

DOI : <https://doi.org/10.1145/2661334.2661427>

To cite this version: Raynal, Mathieu *KeyGlasses: semi-transparent keys on soft keyboard*. (2014) In: 16th ACM SIGACCESS Conference on Computers and Accessibility (ASSETS 2014), 20 October 2014 - 22 October 2014 (Rochester, United States).

Any correspondence concerning this service should be sent to the repository administrator: tech-oatao@listes-diff.inp-toulouse.fr

KeyGlasses: Semi-transparent Keys on Soft Keyboard

Mathieu Raynal
IRIT - ELIPSE Team
University of Toulouse
31062 Toulouse, cedex 9, FRANCE
mathieu.raynal@irit.fr

ABSTRACT

This paper presents the KeyGlass system: a text entry system with dynamic addition of characters based on the previously entered ones. This system is optimized by the use of a prediction algorithm based on a lexicographic tree and bigrams.

Keywords

soft keyboard, character prediction

1. INTRODUCTION

Currently, the keys and characters of the most commonly used soft keyboards are laid out similarly to the physical AZERTY or QWERTY keyboards (depending on culture). This arrangement has the advantage of being known to everyone. However, the major drawback lies in the distance of the most frequently used characters, which is not an issue for physical keyboards given the use of different fingers. On the other side, the use of a single pointer to select characters implies moving it between each character to enter. The distance of the most frequently used characters causes an increase of the distance with the pointer, which results in a decrease of the text entry speed for the user, but also motor or eye fatigue [7].

To minimize travelled distances, two main approaches were studied:

- First, to match the different characters on the keyboard to bring the characters which frequently follow each other closer. Keyboards have been generated either intuitively (eg Fitaly or OPTI keyboards[3]) or

from optimization algorithms (such as the GAG keyboard[6]);

- Second, the use of a word prediction system. One of the most classic use of a prediction system is to provide a set of the most probable words from the beginning of the user input. If one of the proposed words is the desired one, the user can then validate this word instead of continuing inputting via the keyboard. This reduces the number of characters to be entered. Several systems have been proposed on this principle like [1].

Both methods have advantages, but also disadvantages: in the first case, no matter the layout, some characters will still be distant on the keyboard. For the second proposal, the position of the word list prediction may not be optimal: it may be placed on one side of the keyboard, as for KeyStrokes or cover a part of the keyboard as proposed in the POBox system [4].

In order to reduce the movement of the pointing device, we propose a system between the two main lines of research described above: the KeyGlass system offers the most likely characters near the last entered.

2. KEYGLASS SYSTEM

2.1 Principle

Keyglass is a system that provides the user with additional characters throughout the text input. These characters are dynamically added near the last entered character. Proposed characters after each typed character are those most likely to succeed it. The objective was to minimize the distances traveled by the pointer. The text entry speed is not necessarily improved due to visual search time on new keys. To increase the probability of reducing the distances, the system also works recursively: after each new inputted character, either on a fixed key or dynamically added key, new characters are available around the last typed character (see Figure 1). However minimizing the displacement length reduces motor fatigue.

As the semi-transparent toolbox near the pointer proposed in the *toolglasses* [2], we chose to display additional characters on semi-transparent round keys (see figure 1). We preferred to use semi-transparency to give the user an overview of the keyboard, and so never overlapping a character by displaying another that would be positioned above. The character selection is done by clicking on the key re-

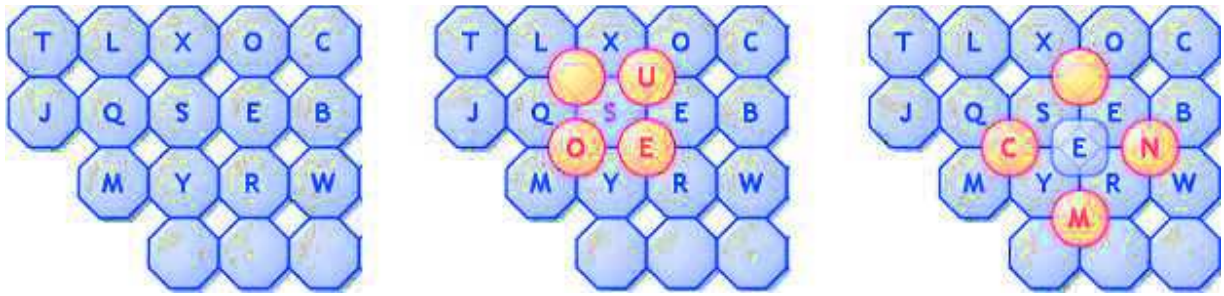


Figure 1: Left: part of the keyboard layout; middle: the KeyGlasses proposed after entering the 's'; Right: the KeyGlasses appearing after entering the 'e' which was placed on a KeyGlass.

Regardless of whether this character is associated with a fixed key or with a KeyGlass.

2.2 System architecture

2.2.1 A modular system

We designed the system in a modular way in order to be able to change each part independently. Our system is split into three parts:

- **The fixed layout of the soft keyboard:** it concerns all the keys that will not move during the text input. Associated characters with these keys are also fixed. This set of keys / characters is described in an XML format. We chose an XML language because it allows to easily change the characteristics of the basic keyboard. We can change the shape, position, color of a key and its associated character. This fixed layout is thus completely independent of KeyGlasses that can be added on top;
- **The prediction system:** it receives the characters entered by the user. From these characters, it returns a ranking of characters that have the highest probability of being typed;
- **The display manager:** it manages the display of the KeyGlasses above the fixed keyboard. It handles the KeyGlasses' positions, their shape and color. For each keyglass, it associates one of the highest ranked character with the additional key.

2.2.2 Operating Mode

Figure 2 shows the operating principle of the KeyGlass system. The numbers on the arrows describe the order of transmission of messages between the different modules.

1. When a character is entered, it is sent to the prediction system. At the same time, the pointer coordinates are sent to the display manager.
2. The prediction system calculates the probability of each character following the last one received. It then ranks these characters in ascending order according to their probability of appearing, before sending this classification to the display manager.
3. Finally, the display manager, after determining the KeyGlasses positions, affects each character on a KeyGlass and returns the information to the soft keyboard that displays them.

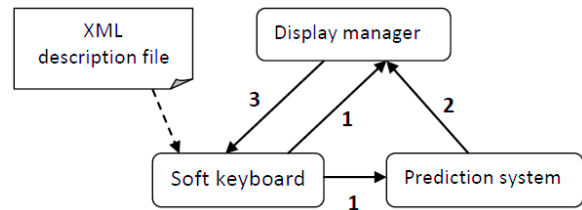


Figure 2: Architecture of KeyGlass system

Once the KeyGlasses are displayed above the soft keyboard, they are considered by the system as belonging to the soft keyboard. Thus, as we mentioned in Section 2.1, this process can operate recursively. Pressing a KeyGlass to enter a character leads exactly to the same process that we have described above.

2.2.3 Communication protocol

We used the IVY bus for communication between the different modules. To facilitate the replacement of a module with another one, we defined a communication protocol. Thus, each module knows exactly what it can receive and / or send. Replacing a module with another is easy: the new module must be connected to the IVY bus and follow the communication protocol.

To be compatible with the E-ASSISTE platform [5], this communication protocol uses the same protocol than E-ASSISTE. The various messages respecting this protocol can also be collected by the E-ASSISTE platform and enable analysis of user input and functioning of KeyGlass system.

3. REFERENCES

- [1] C. Beck, G. Seisenbacher, G. Edelmayer, and W. Zagler. First user test results with the predictive typing system fasty. In *9th ICCHP : Computer Helping People with Special Needs*, pages 813–819, 2004.
- [2] E. Bier, M. Stone, K. Pier, W. Buxton, and T. DeRose. Toolglass and magic lenses: the see-through interface. In *Proc. of SIGGRAPH'93*, pages 73–80. ACM Press, 1993.
- [3] I. MacKenzie and S. Zhang. The design and evaluation of a high-performance soft keyboard. In *Proc. of CHI'99*, pages 25–31. ACM Press, 1999.

- [4] T. Masui. An efficient text input method for pen-based computers. In *Proc. of CHI'98*, pages 328–335. ACM Press/Addison-Wesley Publishing Co., 1998.
- [5] M. Raynal, S. Maubert, N. Vigouroux, F. Vella, and L. Magnien. E-assiste: A platform allowing evaluation of text input system. In *Proc. of UAHCI'05*. Lawrence Erlabaum Associates (LEA), 2005.
- [6] M. Raynal and N. Vigouroux. Genetic algorithm to generate optimized soft keyboard. In *CHI '05 extended abstracts*, pages 1729–1732. ACM Press, 2005.
- [7] F. Vella, N. Vigouroux, and P. Truillet. Sokeyto: a design and simulation environment of software keyboards. In A. Pruski and H. Knops, editors, *8th European conference for the advancement of assistive technology in europe*, pages 723–727. IOS Press, 2005.