

Augmented letters: mnemonic gesture-based shortcuts

Quentin Roy, Sylvain Malacria, Yves Guiard, Éric Lecolinet, James Eagan

▶ **To cite this version:**

Quentin Roy, Sylvain Malacria, Yves Guiard, Éric Lecolinet, James Eagan. Augmented letters: mnemonic gesture-based shortcuts. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Apr 2013, Paris, France. 10.1145/2470654.2481321 . hal-01164207

HAL Id: hal-01164207
https://hal.science/hal-01164207

Submitted on 16 Jun 2015

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.

Augmented Letters: Mnemonic Gesture-Based Shortcuts

Quentin Roy[1,2] **Sylvain Malacria**[1,3] **Yves Guiard**[1] **Éric Lecolinet**[1] **James Eagan**[1]

[1] Telecom ParisTech, CNRS LTCI UMR 5141, Paris, France

[2] GE Healthcare, Buc, France

[3] University of Canterbury, Christchurch, New Zealand

quentin@quentinroy.fr sylvain@malacria.fr {firstname}.{lastname}@telecom-paristech.fr

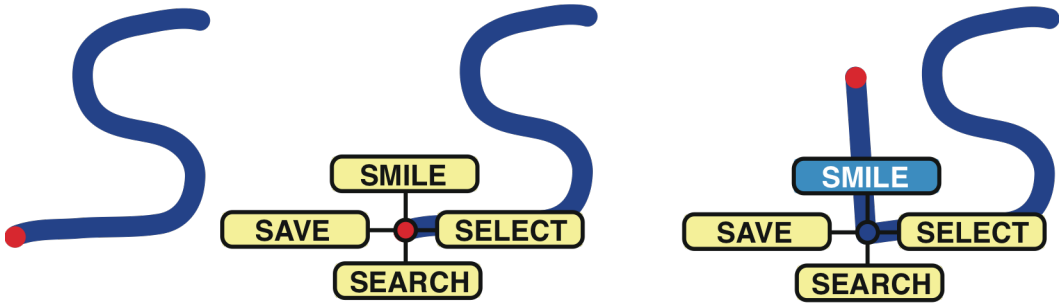


Figure 1: Augmented Letters in novice mode. The menu appears after a 500ms pause.

ABSTRACT

We propose Augmented Letters, a new technique aimed at augmenting gesture-based techniques such as Marking Menus [9] by giving them natural, mnemonic associations. Augmented Letters gestures consist of the initial of command names, sketched by hand in the Unistroke style, and affixed with a straight tail. We designed a tentative touch device interaction technique that supports fast interactions with large sets of commands, is easily discoverable, improves user's recall at no speed cost, and supports fluid transition from novice to expert mode. An experiment suggests that Augmented Letters outperform Marking Menu in terms of user recall.

Author Keywords

Handwriting; Gestures; Semantics; Learning; Recall; Unistroke

ACM Classification Keywords

H.5.2. [Information interfaces and presentation]: User Interfaces - Graphical user interfaces.

General Terms

Human Factors; Design; Experimentation; Measurement

INTRODUCTION

Among the various interaction techniques that have been designed to enhance the expert use of software, keyboard shortcuts and Marking Menus work especially well [6,9]. Keyboard shortcuts are remarkably fast, but are difficult to learn [6] and are not available on many typical mobile devices. In contrast Marking Menus, which support a seamless transition from novice to expert mode, are very well suited to touch-based interaction. However, they rely on an arbitrary mapping between commands and directions, meaning that mappings need to be learned from scratch. Another limitation is that they cannot accommodate more than eight directions or so [10], making a hierarchical organization necessary. Some variants use additional parameters like curvature or shape [1,12] to increase menu width, but the command/gesture mapping remains arbitrary.

We describe Augmented Letters, a novel gesture shortcut that combines unistroke letters with Marking Menus. The aim is to simplify command memorization and to reduce the cognitive load by using the initials of command names as gesture shortcuts. The stroke is augmented with a tail that can be oriented in up to eight directions, so as to handle conflicts amongst commands that share the same initial.

Augmented Letters flatten the command hierarchy, reducing it to a mnemonic plus a direction. Assuming the command vocabulary of the system is already known (*e.g.* Save, Smile, Select, Search, etc., see Figure 1), users only need to learn the tail stroke to append. Using the novice mode of the technique, users can fluidly turn into experts through rehearsal.

In this paper, we:

- Present the design of Augmented Letters and
- Report the results of a user study suggesting that a set of commands is easier to memorize with Augmented Letters than with Marking Menus.

© ACM, 2013. This is the author's version of the work. It is posted here by permission of the ACM for your personal use. Not for redistribution. The definitive version was published in CHI '13: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, April 27 – May 2, 2013, Paris, France. DOI: 10.1145/2470654.2481321

OVER-LEARNING AND GESTURE AUTOMATICITY

Augmented Letters capitalize on the fact that the various skills that constitute the human language are considerably overlearned [13]. In the literate adult, owing to a considerable amount of sustained practice, naming, reading, typing, and handwriting are remarkably automatic, despite the well-known arbitrariness of the linguistic sign [13].

In this regard, the classic psychology of reaction time (RT) teaches some interesting lessons. As expressed by the Hicks-Hyman law [7], choice RT increases linearly with the logarithm of the number *N* of alternatives. But no less important is the fact that the slope of that linear dependency strongly depends on training: for example, Fitts and his colleagues [4] have shown that the slope of the Hicks-Hyman law virtually zeroes out if the task is to utter the name of visually presented characters: the duration of such a reading reaction is not just short, it is hardly affected at all by the size of the stimulus set.

This class of findings is of relevance to HCI when it comes to the design of input vocabularies. In general, the larger the set of possible commands, the more difficult the choice, yet if the memory link has been trained to the point it becomes automatic, the number of alternatives matters no longer. Hence our general argument that the design of gesture shortcuts for touch screens should leverage over-trained linguistic mappings as much as possible and introduce new mapping conventions parsimoniously.

It should take the user of a mobile device little effort, having retrieved from long-term memory the name of a known command, to identify its initial letter, and to sketch its shape on a touch screen, as such a sequence involves no extra arbitrariness whatsoever. Or, shifting the emphasis from speed to contents, it should be relatively easy to master a pretty large vocabulary of commands with a gesture code that relies on old handwriting habits.

HCI researchers, having only recently fully realized the considerable potential, for input on mobile device, of the gestures that consist of cursive fingertip tracing [11], still face a very large design space. The present paper presents a preliminary exploration of the possible advantages of jointly exploiting the merits of marking menus, known to minimize the effort in movement space [9,10], and handwriting, known to rely on over-trained linguistic associations.

AUGMENTED LETTERS

An augmented letter is a *Graffiti*-like unistroke letter [5] which is 'augmented' with a tail that may point in up to eight directions (Figure 1). The letter typically is the initial letter of the associated item, thus allowing a straightforward semantic mapping between the gesture and the command. On the other hand the tail makes it possible to differentiate between items that start with the same letter, hence handling name collisions. Up to eight commands can thus be specified with one handwritten letter. In theory, this design makes it possible to define up to 26x8=208 different commands in expert mode. In fact there are a few conflicts, e.g. a left tailed C is similar to a left tailed G (tail length does not matter). We counted about six of them depending on the recognizer, leaving 208-6=202 different commands.[1]

Two cases are possible. If the user knows the tail, (s)he will trace the complete augmented letter. Otherwise, (s)he will just draw the letter and then wait 500ms (an empirically optimized delay) for a marking menu to appear, showing the possible tails and the corresponding commands (see Figure 1). Thus, just as with Marking Menus, Augmented Letters allow a fluid transition from novice to expert mode [1], with the same gestures serving to invoke the same commands in either mode.

Our implementation relies on the $1 recognizer [15]. In novice mode the recognition, performed before the tail is drawn, uses a specific recognizer of (non-augmented) letters. In expert mode, in which the whole gesture (letter + tail) needs to be recognized, we found that using a different recognizer depending on the direction of the gesture's tail minimized the probability of recognition error.

EXPERIMENT

Augmented letters have a potential to provide quick access to a large set of commands. We conducted an experiment focusing on learning inspired by [2]. The goal was to evaluate the performance of Augmented Letters (AL) relative to Marking Menus (MM), a widely-accepted baseline. Our instructions emphasized memory accuracy over performance speed. Accordingly, the primary dependent measure was the percentage of correct recall.

Apparatus and Participants

Our experiment was conducted using an Archos 80 tablet equipped with a 1,5Ghz processor and running Android 4. Twelve volunteers (5 females, aged 21-48), participated.

Item Set Size and Menu Layout

Being interested in the memorization of large item sets, we wanted to be sure we exceeded users' short-term memory, plausibly estimated at 5-9 items [8]. Non-hierarchical MM can hardly exceed eight angular sectors. Our MM thus had one special sector, labeled "others", that gave access to a second level with eight additional items. This yielded a total of 7+8=15 possible items for the MM. We would have preferred a larger item set, but this would have disadvantaged the MM technique. Item vocabulary were chosen in order to form 5 groups of items starting by the same letter, making 5 different AL with 4 possible tail directions.

[1] In a complementary study, we inventoried the number *N* of commands available in 32 Macintosh applications. Using this real-world corpus, the probability *p* of the initial letter referring to more than eight commands increases linearly with *N*, the best-fitting equation being $p=0.0019N-0.07$, $r^2=.92$. Mobile devices use relatively lightweight software, and so the concern about command conflicts with AL is almost certainly manageable. For example with 50 commands, the predicted probability of the initial letter being shared by more than eight commands is a modest 0.025.

Methods

Two sets of items containing country names and animal names were used. In order to avoid memory interference, the two techniques were not intermixed. The experiment was divided into two successive parts, one dedicated to each technique. Technique order was balanced amongst participants using a Latin square.

Task and Stimuli

The participant was instructed to try to memorize as many gestures as possible. Each block started with the name of the technique involved and the block type (learning or test). Each trial was preceded by a circular countdown. The name of the target was then displayed in the top region of the screen and the participant was to then perform the corresponding gesture. The trial ended when the user released the finger from the tablet, triggering the next trial.

Learning vs. Test Blocks

Our design protocol is schematized in Figure 2. The same 12 items were served in three successive learning blocks, each item being presented three times in a row (Figure 3). In a learning block, the participant was free to wait for the menu or to respond in expert mode. A test block immediately followed each learning block (Figure 2). In this mode no menu was available, the participant having to recall the items from memory. The experiment lasted about 45 minutes, with a total of 2 x (3x (3x12+12) + 3) = 294 trials performed.

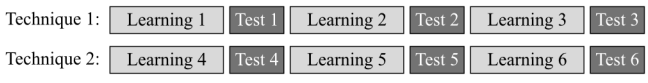


Figure 2: Organization of blocks in the experiment.



Figure 3: Organization of trials within blocks.

RESULTS AND DISCUSSION

Expert-mode performance

On average the recall rate for test blocks (Figure 4) was 24.5% higher with AL (76.6%, 9.2 items) than MM gestures (52.1%, 6.25 items), an effect found to be statistically significant in an analysis of variance ($F_{3,11}$=7.08, p<.01).

The corresponding mean performance speeds are presented in Figure 5. The recall time in expert mode was the same with AL and MM (3.80s vs. 3.79s). Parsing the response time into a reaction time (RT) and a gesture-execution time (ET), there were differences. It took participants more time to execute AL than MM gestures (1.86s vs. 0.83s, $F_{1,11}$=178, p<.01) but that difference was completely offset by a shorter RT (1.92s vs. 2.97s, $F_{1,11}$=19, p<.01). This latter finding is consistent with the view that the initiation of an AL benefitted from the high familiarity of the cognitive path leading from a word, be it a country name or of computer commands, to its handwritten expression.

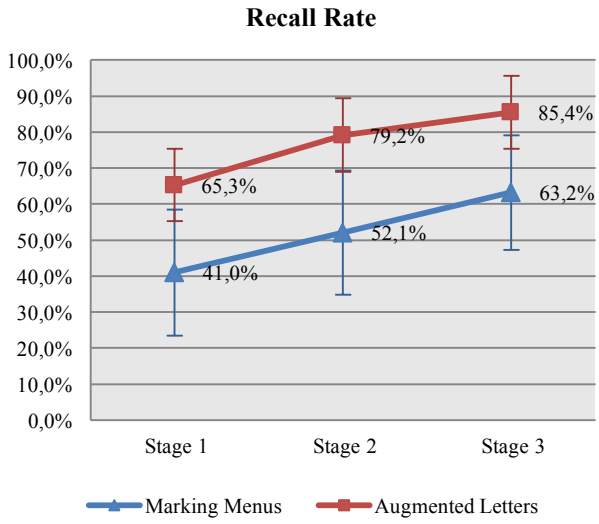


Figure 4: Recall rate for AL and MM, at three learning levels.

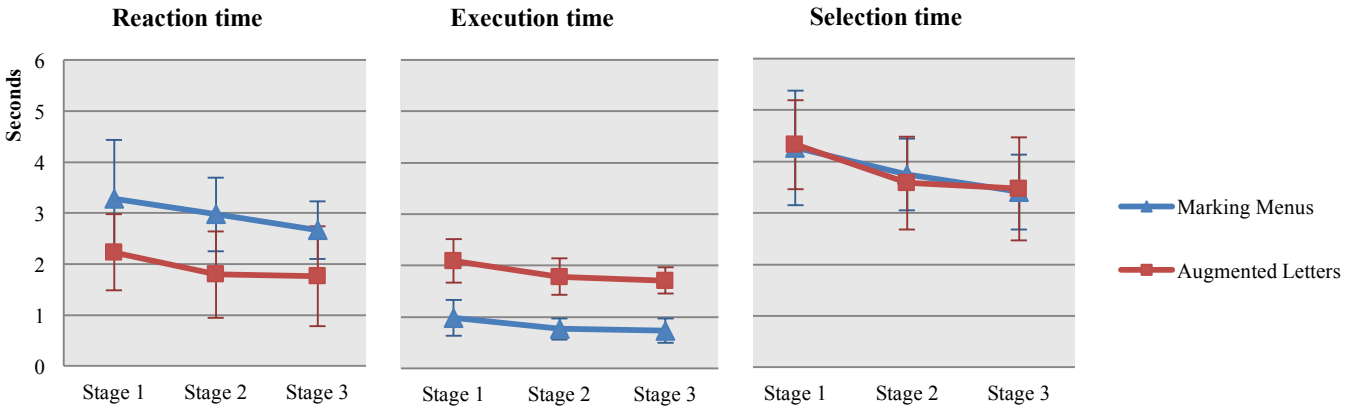


Figure 5: Analysis of performances speed in expert mode for AL and MM, at three learning levels.

Novice-Mode Performance

We examined novice-mode recall time data from the first learning block of the experiment to see how the participants managed with the two techniques when first discovering them. The mean selection time in novice mode was 6.6s for AL and 7.9s for MM (Student's t =1.46, one-tailed p=.09). Although marginally significant, this outcome is certainly not inconsistent with the view that the benefit entailed by recourse to an over-trained code was immediate.

Spontaneous Use of the Expert Mode

In learning blocks the participants were free to either wait for the menu (novice mode) or to perform the hierarchical gesture straightaway (expert mode). On average over the last two learning phases of the experiment, the spontaneous usage of the expert mode was 22.2% more frequent with AL (66.0%) than with MM (43.8%) (t=2.14, one-tailed p=.03). It was easier for our participants to learn letter tails than two freshly defined mappings at both the first and second level of MM.

CONCLUSION

We do not wish to over-interpret the results of this small exploratory experiment, which we just offer as an illustration for a tentative point. The logic of the classic graphical user interface (GUI) [14], with its desktop metaphor, its direct manipulation paradigm, and its menu plus mouse-pointing protocol, is hard to extend to tablets and smartphones. Many basic commands like cut-and-paste are cumbersome on current smartphones.

The remarkable efficiency on the PC of keyboard shortcuts, which rely on language associations, may not have received all the attention it deserves in HCI research. Solutions to the input problem offered by the GUI have been enormously successful, allowing millions of novices to get acquainted with computers. However, expert users still prefer to resort to keyboard shortcuts, and the uninterrupted proliferation of commands in the GUI has become problematic [3]. Increased awareness that language is a major under-exploited input resource in mobile computing might usefully encourage HCI research to further explore the potential of handwriting, along the lines currently suggested by Li and his colleagues [e.g., 11].

The larger the set of commands, the greater the benefit that can be expected from the over-learned skills of language such as drawing letters. Users being able to memorize more AL than MM gestures, at apparently no speed cost, the AL technique seems well-suited to touch-screen interfaces, which badly miss keyboard shortcuts.

AKNOWLEDGEMENTS

We acknowledge the financial support of the Quatro2 project funded by Systematic Paris-Region. We warmly thank L. Launay, F. Peter and J. Knoplioch from GE Healthcare and S. Perrault, D-B Vo and M. Serrano from Telecom ParisTech for their support and suggestions.

REFERENCES

1. Bailly, G., Lecolinet, E., and Nigay, L. Flower menus: a new type of marking menu with large menu breadth, within groups and efficient expert mode memorization. In Proc. of AVI, ACM (2008), 15–22.
2. Bau, O., and Mackay, W. OctoPocus: a dynamic guide for learning gesture-based command sets. In Proc. of UIST '08, ACM, New York, NY, USA, 37-46.
3. Beaudouin-Lafon, M. (2004). Designing interaction, not interfaces. In Proc. of AVI '04, ACM, New York, NY, USA, 15-22.
4. Brainard, R.W., Irby, T.S., Fitts, P.M., and Alluisi, E.A. Some variables influencing the rate of gain of information. Journal of Experimental Psychology 63, 2 (1962), 105–110.
5. Castellucci, S.J. and MacKenzie, I.S. Graffiti vs. unistrokes. In Proc. of CHI '08, ACM Press (2008), 305-308.
6. Grossman, T., Dragicevic, P., and Balakrishnan, R. Strategies for accelerating on-line learning of hotkeys. In Proc. of CHI '07, ACM Press (2007), 1591-1600.
7. Hyman, R. Stimulus information as a determinant of reaction time. Journal of Experimental Psychology 45, 3 (1953), 188–196.
8. Jones, G.V. Tests of the dual-mechanism theory of recall. Acta Psychologica 50, 1 (1982), 61–72.
9. Kurtenbach, G. and Buxton, W. Issues in combining marking and direct manipulation techniques. In Proc. of UIST '91, ACM Press (1991), 137–144.
10. Kurtenbach, G. and Buxton, W. The limits of expert performance using hierarchic marking menus. In Proc. of INTERACT/CHI, ACM (1993), 482–487.
11. Li, Y. Gesture search. In Proc. of UIST '10, ACM Press (2010), 87-96.
12. Roudaut, A., Bailly, G., Lecolinet, E., and Nigay, L. Leaf Menus: Linear Menus with Stroke Shortcuts for Small Handheld Devices. In Proc. of INTERACT, Springer-Verlag (2009), 616–619.
13. Saussure, F. de (1916). Course in General Linguistics.
14. Shneiderman, B. Seven plus or minus two central issues in human-computer interaction. ACM SIGCHI Bulletin 17, 4 (1986), 343–349.
15. Wobbrock, J.O., Wilson, A.D., and Li, Y. Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes. In Proc. of UIST '07, ACM Press (2007), 159-168.