

Glyph: A New Stroke Alphabet for Stylus-Based or Key-Based Text Entry

Franck Poirier

Université de Bretagne-Sud
Laboratoire VALORIA & Laboratoire TAMCIC
Campus de Tohannic – 56000 Vannes - France
Franck.Poirier@univ-ubs.fr

Abstract

Glyph is a new text entry method for handheld devices. This method is based on a restricted set of six shape primitives which allows to decompose letters and symbols into a sequence of one to three primitives according to a principle of analogy with Roman characters.

Glyph is designed for stylus-based entry like other unistroke methods (Graffiti, EdgeWrite, T-Cube...), but also for key-based entry with one key per primitive. So, two types of interfaces are investigated: reduced physical keyboards and soft keyboards. For the latter, a dynamic keyboard is proposed, a kind of movable pie menu which appears to the contact point of the stylus with the screen. After each primitive, the keyboard is automatically re-centred under the stylus, in order to minimize the size of the flick gesture towards the next primitive, and consequently the entry speed in accordance with Fitts' law.

Glyph has two main advantages. First, the analogy principle allows to recall easily in memory the primitives sequences. Secondly, the small number of primitives, combined with the gestural interaction, quickly leads to the construction of effective gestural routines for every character. The first results of evaluation show that users have no problem to learn the Glyph character chart and that they improve rapidly their performances with use.

1 Glyph: a stroke alphabet

This paper describes a new data entry method called Glyph. Many data entry methods have been developed for stylus-based systems these last years. The strong development of mobile systems (PDA, mobile phones) mainly motivated the design of alternative input methods (Myers, 2002). The main characteristic of the approach is to allow the decomposition of any symbol from a reduced set of only six shape primitives.

Many papers have popularized different data entry methods (Sirisena, 2002), (Wobbrock, Myers & Kembel, 2003), (Poirier & Schadle, 2004). Glyph belongs to the family of character-level stroke alphabets. Glyph can be compared with T-Cube (Venolia & Neiberg, 1994) or Quikwriting (Perlin, 1998) for the use of flick gestures, Symbol Creator (Luoma, 2003) for the decomposition into sequences of primitives, or Unistrokes (Goldberg & Richardson, 1993) for the principle of similarity to the Roman alphabet.

1.1 Set of primitives and character chart

In order to decompose characters, Glyph uses the set of six shape primitives given in figure 1.

The default alphabet is the Roman lowercase alphabet. Each letter is associated with a sequence of primitives which maximizes the analogy with the drawing of the letter (manuscript writing or sans serif typefaces). As shown in the figure 2, the analogy was looked for first at the shape level, and next at the temporal level. Indeed, the drawing of the letter “g” starts by the left curve, continues by the vertical stroke above the baseline that closes the bowl, then finishes by the downstroke (defined as the loop in the case of the “g”). According to the complexity of the character, it can be coded by one primitive (case of “i”), two primitives (case of “m”) or three (case of “f”).

Name of the shape primitive (abbreviation, symbol in the text)	Shape
Right curve (R, \supset)	
Left curve (L, \subset)	
Upwards curve (U, \cap)	
Downwards curve (D, \cup)	
Horizontal stroke (H, $—$)	
Vertical stroke (V, $ $)	

Figure 1. The six shape primitives

The decomposition is based on the classical principles of typography: one or two primitives are associated to the body of the character between the baseline and the meanline excluding the ascender and the descender (bowl for “a”, stem for “i”, cross stroke for “t”, leg for “k”...), and possibly only one for the ascender or the descender (exclusive or). Note that the decomposition doesn’t take into account the detail of the characters such as spur, serif or dot of “i”, only the main features are coded. The decomposition is also chosen in order to frequent characters can be entered very rapidly, so “c”, “i”, “n”, “u”, are respectively coded by the primitive L (\subset), V ($|$), U (\cap), D (\cup).

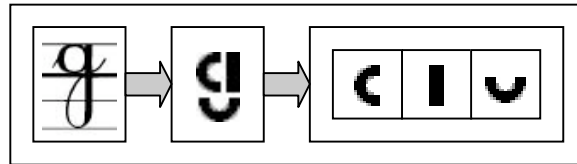


Figure 2. Decomposition of the letter “g”

The Glyph character chart for the letters of the alphabet is shown on figure 3.

a	b	c	d	e	f	g	h	i	j
k	l	m	n	o	p	q	r	s	t
u	v	w	x	y	z				

Figure 3. Glyph character chart

1.2 The segmentation primitive

In order to allow a more complete set of symbols (punctuation, figure, accent...), a seventh primitive is added, called “segmentation primitive” (Segm). This primitive is used for the following actions:

- segmentation between characters for those coded by only one or two shape primitives,
- entry in one stroke the letter “space” which is the most frequently used character,
- entry of accentuated characters,
- entry of punctuation symbols,
- selection of an alphabet (figures, symbols, abbreviations...),
- correction and navigation (delete a character, a word or the selection), undo and redo command,
- access to the system clipboard.

2 Keyboards design

The design principles of the method can be applied to various devices. Two types of interfaces is presented: physical and virtual keyboards. Both Glyph physical and virtual keyboards are well adapted to mobile handheld devices. The virtual keyboard takes little space on the screen and the physical one needs less than 10 keys.

2.1 Physical keyboards

Different types of physical keyboards can be investigated. It's possible to adapt the method to already existing devices or to propose devices specifically designed for Glyph (Uguen & Poirier, 2004). Although Glyph uses a set of seven primitives (six shape primitives plus the segmentation primitive), the method can be adapted to devices with less than seven keys.

It is possible to design keyboards with five, four or three keys: with one key for segmentation (end of character entry), respectively three to one ambiguous keys (for example $\{C/D\}$, $\{N/U\}$ and $\{—/ | \}$ for a version with five keys) and the last one for run (scrolling between C and D for example). The run key can be removed if a timer is used for the validation according to the principle of the multitap method (mobile phone keyboard). Thus, the device has only four, three or two keys.

Finally a version with a single key (or a roller) is also possible. The key allows to scroll between the seven primitives and a timer is used for validation. This approach is especially adapted to motor-impaired persons who can use only a simple button.

So, it is possible to adapt Glyph to the T12 mobile phone keyboard (composed of twelve keys). Figure 4 proposes different layouts of the primitives on the keyboard.



Figure 4. Alternatives for mobile phones

2.2 Virtual keyboards

2.2.1 Dynamic keyboard

Usually, virtual keyboards are static keyboard. For Glyph, a special virtual keyboard has been designed, smaller and more adapted to reduced interfaces (see Figure 5).

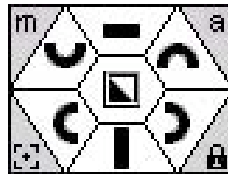


Figure 5. The 7 primitives dynamic keyboard

This keyboard is designed for a continuous entry at the symbol level. Between each symbol the user must hold the stylus up. All input sequences must have a length of three primitives or must be supplemented with one segmentation primitive. To select a primitive, the user has to drag the stylus in one of the small white cells by a short flicking gesture (see Figure 5). The outside rectangular zone is used as a feedback zone.

This keyboard is called “dynamic” for two reasons. First of all, the keyboard proposes different feedbacks during the entry of sequence of primitives. For example, the already seized primitives are displayed near the insertion cursor (see Figure 6). The second justification of the dynamic aspect of the keyboard is more important. It is due to the fact that during the text input, after each primitive, the keyboard is automatically recentered at the point of contact of the stylus with the screen, in order to allow the direct input of the next primitive.

The virtual keyboard is always displayed under the screen pointer (feedback of the stylus). Thus, it doesn’t occupy a specific zone on the screen, it appears to the foreground and the user can move it everywhere.

So, to enter a character, the user presses the stylus in the cell of the first primitive, then moves the stylus from the central cell to one of the six peripheral cells. In other words, for the second and third primitives, the user only drags the stylus in one of the 6 possible directions: North (N), North-East (NE), South-East (SE), South (S), South-West (SW) or North-West (NW). If the character is composed of three primitives, the input requires only two flick gestures in one of the possible directions without transient (or articulatory) stroke between primitives.



Figure 6. Feedback of the previous features in case of “h”

The entry of a sequence of only one primitive is carried out by a simple pressure on one of the seven cells. A sequence of two primitives corresponds to a continuous gesture from the first primitive towards the second one. Between both, the keyboard is automatically recentered on the central hexagon, and so on for three primitives. For example, the letter “y” is entered by pointing on the D primitive (U), then dragging the stylus to the South (primitive I), then to the North-West (primitive U).

Note that the gesture can be composed of the minimal length of primitives by adding a timer between the entry of the second and third primitives.

3 Evaluation

Three input methods were chosen to make a first evaluation: Glyph, Block Recognizer, a method available on Pocket PC, equivalent to Graffiti, the well known single-stroke alphabet developed by Palm Inc. (Blickenstorfer, 1995), and a standard virtual keyboard on Pocket PC.

The test included five sessions proposing each the five following sentences in French :

- “abcdefghijklmnopqrstuvwxyz”
- “maman ne dit rien” (“mom does not say anything”)
- “les trois petits cochons” (“the three small pigs”)
- “isabelle a les yeux bleus” (“isabelle has blue eyes”)
- “un anneau pour les dominer tous “ (“one ring to dominate them all”)

Five sessions take place on five days. The total corpus of test includes 25 sentences and 615 letters.

The general conclusions of the test are the following ones:

- For Block Recognizer and the virtual keyboard, the observed performances are coherent with the already published evaluations.
- The performances vary appreciably according to the experience of the user (knowledge of alternative input methods to the physical keyboard, use of a personal assistant, use of one method in past...).
- The performances are evaluated in characters per second (cpm) (MacKenzie & Zhang, 1999) and take into account entry errors; whatever the method is, these errors punish largely the performances. For Glyph and Block Recognizer, the errors are comparable.

- An experimented user who makes few errors reaches a double cpm of a novice. For Glyph and Block Recognizer, of the order of 50 cpm and with the virtual keyboard of the order of 70 cpm.
- For a given user, not familiarized with the usage of a PDA, the performances are of the same order: 40 cpm for the virtual keyboard, approximately 20 cpm for Glyph and Block Recognizer.
- During the five sessions the users improve their performances. The percentage of improvement is more important for Glyph or Block Recognizer than for the virtual keyboard. For Glyph, the ratio is of the simple to the double for a novice between the first and the fifth session (from 12 to 25 cpm).
- For Glyph, the novice user seizes every character by decomposing it primitive by primitive, for example for "x", he (she) is going to select the primitive R (⊃) then the primitive L (⊂). According to the keyboard of the figure 5, R (⊃) is on the zone SE and L (⊂) on the zone SW. Very quickly, this user is going to learn to associate "x" with a quick and continuous gesture SE + SW, without trying to aim exactly at every primitive.

After this quick evaluation, a more complete evaluation will be led as well as a theoretical study on the basis of the Fitts' Law and the Keystroke Level Model.

4 Conclusion

Glyph is a complete data input method which proposes a simple principle of analogy in order to make data entry easier. This simplicity to find or recall in memory primitives sequences and its ability to be adapted on various devices show that this new method can be used not only by expert users on a single device but also by occasional users of a whole set of devices. Glyph is adapted to stylus-based input as well as to key-based entry.

The dynamic virtual keyboard proposed for Glyph favors the construction of "gestural routines" for every character. For example, the user easily associates the quick gesture SE + SW to "x" or S + S to "l". While the novice user decomposes every character primitive by primitive by aiming slowly at the zone of the keyboard associated to every primitive; the user knows quickly how to enter every character by executing the gesture associated to it (the user has become an expert of Glyph). At this stage, the entry is sharply faster and less tiring, because the entry asks for much less motor and cognitive control of the user.

5 References

- Blickenstorfer, C. (1995). Graffiti: Wow! *Pen Computing Magazine*, p. 30-31.
- Goldberg D., Richardson C. (1993). Touch typing with a stylus. *Proceedings INTERCHI'93*, 80-87.
- Luoma M. (2003). Symbol Creator: Usability Evaluation of the Novel Pen-Based Text Input Technique. *Technical Notes New Interaction Techniques 2003*, universit  de Tampere.
- MacKenzie, I. S., Zhang, S. Z. (1999). The design and evaluation of a high performance soft keyboard. *Proc. of the ACM Conference on Human Factors in Computing Systems - CHI '99*.
- Myers B. (2002). Using Handhelds to Help People with Motor Impairments, *ACM Assets 2002*.
- Perlin, K. (1998). QuikWriting: continuous stylus-based text entry. *Proc. of the 11th ACM symposium on User interface software and technology*.
- Poirier F., Schadle I. (2004). Etat de l'art des m thodes de saisie de donn es sur dispositifs nomades. Une typologie des approches. *Proc. ACM IHM'04, Namur*.
- Sirisena A. (2002). Mobile Text Entry, *internal report*, university of Canterbury (NZ).
- Uguen G., Poirier F. (2004). Variations pour une nouvelle m thode de saisie de donn es sur dispositifs mobiles : Glyph. *RJC IHM 2004*, Lacanau.
- Venolia D., Neiberg F. (1994). T-Cube: a fast, self-disclosing pen-based alphabet. Human Factors in Computing System: celebrating interdependence, *ACM Press*.
- Wobbrock J. O., Myers B. A., Kembel J. A. (2003). EdgeWrite: A Stylus-Based Text Entry Method Designed for High Accuracy and Stability of Motion, *ACM UIST'03*.