IST-2003-511598 (NoE)

COGAIN

Communication by Gaze Interaction

Network of Excellence

Information Society Technologies

D4.3 Report on Asian-language versions of COGAIN communication systems

Due date of deliverable: 30.11.2006
Actual submission date: 30.11.2006

Start date of project: 1.9.2004 Duration: 60 months

Tokyo Institute of Technology & The IT University of Copenhagen.

Project co-funded by the European Commission within the Sixth Framework Programme (2002–2006)

Dissemination Level		
PU	Public	x
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Aoki, H., Hansen, J.P., Itoh, K., and MacKay, D. (2006) ***D4.3 Report on Asian-language versions of COGAIN communication systems.*** Communication by Gaze Interaction (COGAIN), IST-2003-511598: Deliverable 4.3. Available at http://www.cogain.org/results/reports/COGAIN-D4.3.pdf

Contributors: Hirotaka Aoki (TIT and ITU)
John Paulin Hansen (ITU)
Kenji Itoh (TIT)
David MacKay (UCAM)

Thanks to: People and organisations involved in design, development and testing of GazeTalk, see http://www.cogain.org/results/applications/gazetalk/about

People and organisations involved in design, development and testing of Dasher, see http://www.inference.phy.cam.ac.uk/dasher/History.html

Table of Contents

Executive Summary

Gaze interaction has been used considerably in Japan. Japan has well-established technical skills for building gaze-tracking systems, and it has a relatively large user community of late-stage ALS patients who are in need of gaze-based forms of communication. Unfortunately, there are only a few gaze-based typing systems available in Japan, and most of them are not free-ware. In order to stimulate further research and the development of Japanese gaze-typing systems, COGAIN partners have taken the initiative to develop Japanese versions of the GazeTalk system that is partly linked to the Dasher system.

This report describes the challenges that the Japanese language impose on designers of alternative input systems. In sections 2 and 3, we report on experiments with Japanese versions of GazeTalk and Dasher. These experiments revealed:

- Users experience stable performance within having typed 50 sentences using GazeTalk.
- Typing speed for both Dasher and GazeTalk exceeds 20 characters per minute (equivalent to 10 English words per minute) after a few hours of practice.
- Learning models predict that users may attain a typing speed of more than 40 characters per minute by the time they have typed 1500 characters.
- Both GazeTalk and Dasher are quite robust with respect to eye-tracker miscalibration.

On this basis it may be concluded that gaze-based typing is indeed a viable option for Japanese people with special communication needs. Hopefully this finding will inspire the development and experimental validation of other Asian gaze-typing platforms. Dasher is a strong candidate for new Asian gaze-communications systems, given that it is already available in 10 Asian languages in addition to Japanese (see http://www.inference.phy.cam.ac.uk/dasher/download/alphabets/ALPHABETS.html#region7).

1 GazeTalk Japanese Version

1.1 Introduction to GazeTalk Japanese Version

GazeTalk Japanese Version, or GazeTalk-J, was developed in the same style as GazeTalk English and Danish versions. This was motivated by our intention to explore and eventually deploy eye tracking as an input modality for Japanese ALS patients, as well as to explore issues in text input for constrained user interfaces such as mobile devices. The current GazeTalk-J (version 5) comprises the following specifications and characteristics:

1. Functionalities:
    - Typing
    - E-mail
    - Saving and loading documents
    - PDF reader
    - Web browsing
    - Music (MP3) player
    - Video player
2. Multiple modality input can be accepted, such as EEG/EMG switches, head-movement, and so forth.
3. A library of layouts was designed and implemented into the system, with an efficient colour-coding scheme based on cognitive walk-through analysis. The standard four-by-three matrix format was used for all layouts. In keeping with the characteristics of human perception, we selected a black background with white, green, and yellow text for our colour scheme, so as to obtain the best colour contrast. The text colours themselves (white/green/yellow) were given their own semantic roles to indicate the user's current typing process. White indicates that the current typing process is finished by activating it; green indicates that the current typing process is ongoing ("more"), and yellow indicates that the current typing process could be cancelled ("go back"). This colour-coding scheme was derived from clarified analysis of cognitive processes in typing, that is, cognitive walk-through analysis.
4. StateEditor, a layout editor used for designing and building the layouts, can be used in GazeTalk-J.

Figure 1 depicts GazeTalk-J's top typing menu. The menu comprises eight on-screen keys, as well as a blank text area. The user can activate ("push") a key by moving his or her gaze over it and fixating on the key's area for a specified "dwell time". As feedback to the user, the interface causes the key to shrink in size so long as the user gazes at it. The size of the shrinking key indicates the dwell time remaining before the key becomes activated. Activation occurs when the size of the key becomes zero.

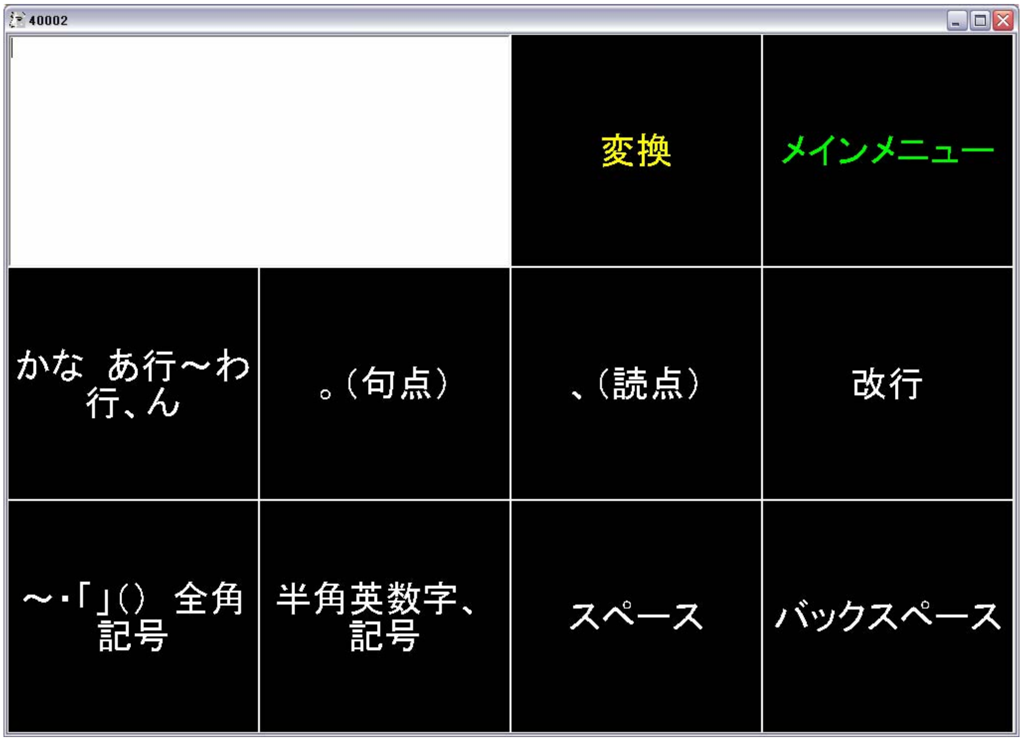


Figure 1. Top typing menu for GazeTalk-J.

1.2 The Japanese language system

The original version of GazeTalk (Hansen et al. 2001), developed for Danish/English-language users, was equipped with a character-prediction function, applying a Markov Chain Model. This allowed users to type with advanced efficiency, as the system could thereby display the statistically most likely characters, given the most recent character typed. The original version therefore employed a dynamic menu system, in which the key for any character should be changed dynamically.

By contrast, GazeTalk-J does not employ character prediction, as this function does not work well with the Japanese language. Instead, the hierarchical, static menu structure was adopted for the Japanese version, in accordance with the Japanese character system. To provide an understanding of the hierarchical menu structure adopted for GazeTalk-J, we first explain briefly how typing is performed in Japanese.

First of all, four distinct character systems are used: Hiragana, Katakana, Kanji (Chinese characters), and alpha-numeric characters. The phonetic character systems Hiragana and Katakana are exactly parallel to each other, and are used to represent fifty Japanese syllabaries. In general, Hiragana characters are used for everyday kinds of text, whereas Katakana characters are reserved for special words, such as foreign words imported into Japanese.

Second, the Hiragana and Katakana character sets are divided into ten *lines*, as follows: the ten consonant sounds—null, *k*, *s*, *t*, *n*, *h*, *m*, *y*, *r*, and *w*—are each combined with the five vowel sounds—*a*, *i*, *u*, *e*, and *o*. Thus, the *line of k* consists of the five syllabaries *ka*, *ki*, *ku*, *ke*, and *ko*, in both Hiragana and Katakana. In this way, each Hiragana and Katakana symbol can be romanised.

A third characteristic of the Japanese language, Japanese sentences need to be represented by the correct combination of Hiragana, Katakana, Kanji, and the alpha-numeric characters. To form a Japanese sentence using a computer-based typing system, we must be able to first input some syllabaries (usually Hiragana characters), and then convert them into the proper combination of Kana/Kanji characters. This conversion can be performed with a Kana-Kanji conversion programme. Perception of the combination of Kana/Kanji characters as being "right" can sometimes vary, depending on the individual reader's age, level of education, language skill, preferences, or other factors.

Finally, it is of vital importance that a typed sentence is grammatically correct and is represented by the correct combination of the four character types. Incorrectness in either case renders the sentence incomprehensible. The above-mentioned constitute the principal differences between Japanese and European languages.

1.3 Text-entry with GazeTalk-J

Taking these unique characteristics of the Japanese language into account, we designed a hierarchical menu structure for the typing interface of GazeTalk-J. In the top typing menu (Figure 1), each of the above-mentioned syllabary groups is allocated a key in the lower and middle rows. When one activates a key, the next-level menu appears, depicting the Hiragana lines, as illustrated in Figure 2. Then, by activating a key in this secondary menu, a tertiary menu appears, depicting the five Hiragana symbols composing this consonant line, i.e., *na*, *ni*, *nu*, *ne*, and *no*. The Hiragana symbol can be typed by fixating one's eyes on the key one wants to input (Figure 3). Thus, a Hiragana symbol can be typed by two "gaze-clicks" for most characters.

As mentioned previously, the Japanese version of GazeTalk also includes a standard Kana-Kanji conversion programme (MS-IME). The typing system includes one or more keys to control this function with each menu. For example, a key for initiating the conversion can be seen to the right of the text areas in Figures 1, 2, and 3 (yellow colour coding). When the user finishes typing a Hiragana character, he or she can switch to conversion mode by pressing the conversion key. Illustrated in Figure 4, a user has just finished typing the Hiragana characters *hi-go-ro,* meaning "always". By pressing the conversion key, the user can activate the conversion programme.

When the conversion programme starts, the conversion functions are displayed (Figure 5): "Show other conversion alternatives", "Finish conversion", "Cursor to the next clause", "Cursor to the previous clause", and "Change to Hiragana characters". In this menu, the text in the text field has already been converted using one of the conversion alternatives. The user can accept the conversion by activating the "Finish conversion" key at the top-right. Subsequently, the top typing menu appears, and the user may resume typing. If the proposed conversion is incorrect, the user must continue the conversion by pressing the "Show other conversion alternatives" key found next to the text field.

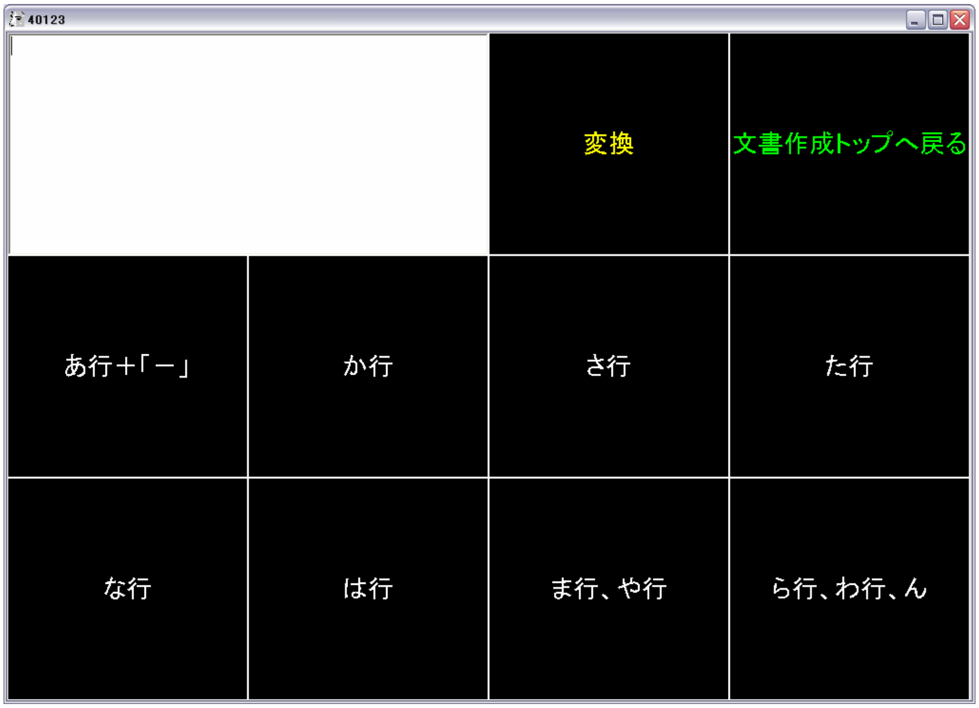


Figure 2. GazeTalk-J, syllabary-groups level.

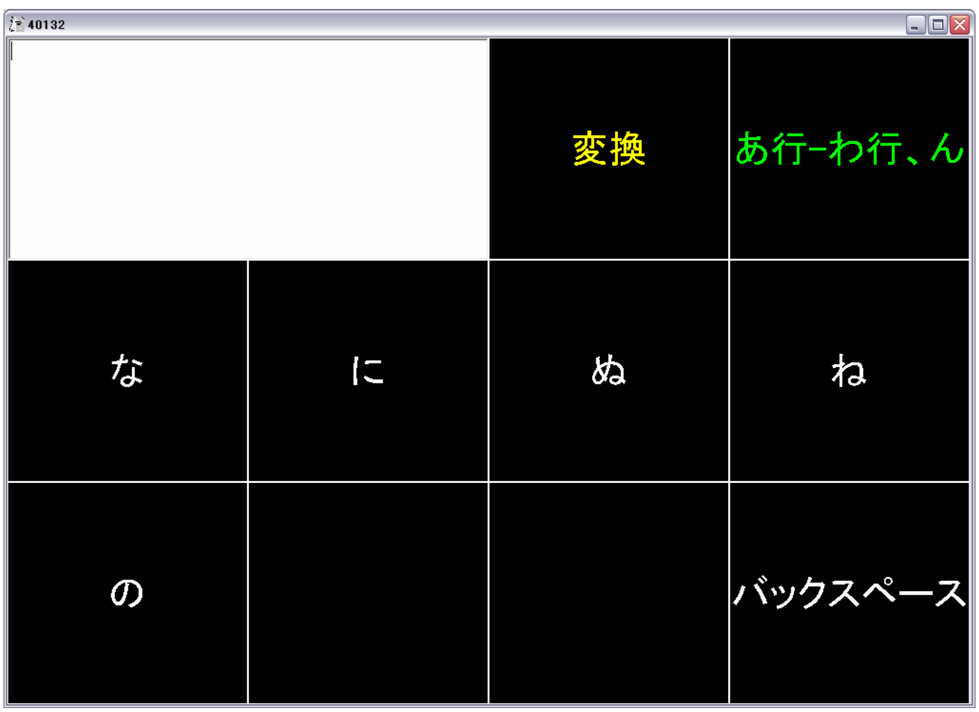


Figure 3. GazeTalk-J, character level.

Figure 4. GazeTalk-J. Typing is finished.

Figure 5. GazeTalk-J. Conversion is started.

Figure 6 illustrates the activation of "Show conversion alternatives". In this menu, the alternatives are numbered beside the text field. The numbers correspond to the numbered keys also displayed. Here, the word the user needs is alternative number 2, which the user can choose by activating the "2" key. If no alternate conversion is needed or desired, the user selects "Finish conversion" instead. After either selecting or accepting a conversion, the top typing menu appears once more (see Fig. 1), and typing can resume as before.

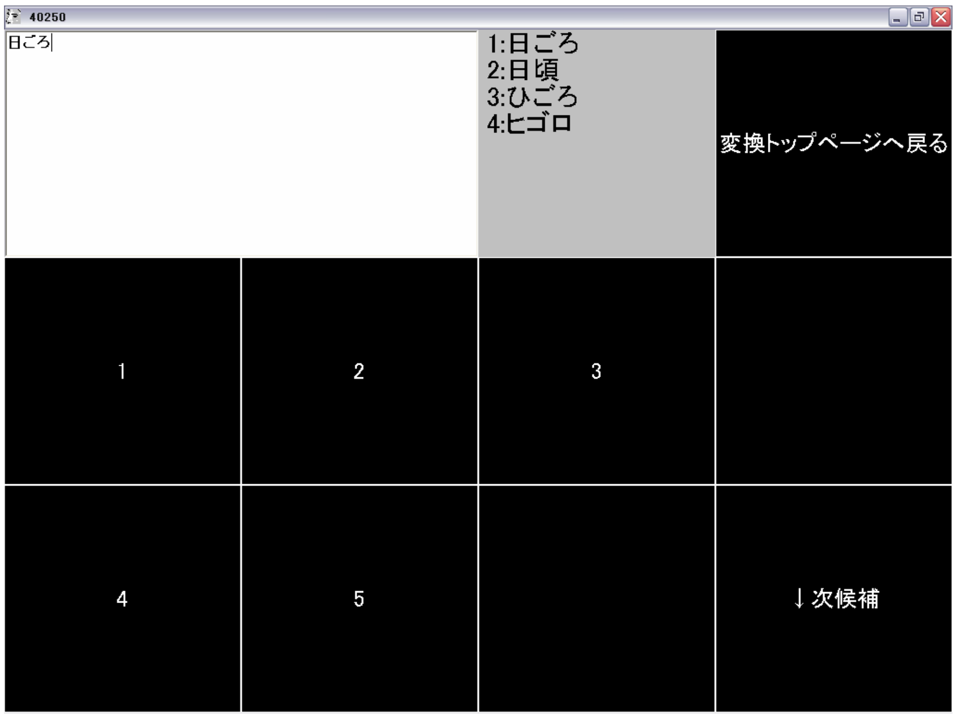


Figure 6. GazeTalk-J with conversion alternatives shown.

1.4 Text-entry procedure in Dasher

Dasher is a text-entry interface controlled by natural, continuous, pointing gestures such as gaze-pointing. Ward and MacKay (2002) reported that the gaze-pointing version of Dasher allowed an experienced user to compose text just as quickly as with normal handwriting, or approximately 29 words per minute (WPM); using a mouse, experienced users could write as quickly as 39 WPM.

Figure 7 presents the Dasher interface, as a user homes in on an intended character. Each Hiragana character is described such that a zoom action corresponds to a character input. When the size of the zoomed area exceeds a given threshold, the character occupying the area becomes typed. A user can input text by choosing where to zoom, and this also can be done continuously.

Dasher may be accessed directly from GazeTalk-J. The link to Dasher from GazeTalk-J is given in the top menu of GazeTalk-J, reminding the user that the option exists whenever GazeTalk-J is opened. Figure 8 (parts **a** and **b**) shows Dasher operating in tandem with GazeTalk-J, with the Dasher area allocated into the left side of the display.

In the example shown, the user is writing *Ha-ji-me-ma-shi-te,* which means "How do you do?" If the user activates "Start input", the middle-left key in Figure 8**a**, he or she will open Dasher. The speed in Dasher can be adjusted with the middle-right and lower-right keys (faster and slower, respectively). The user can stop Dasher text entry with the middle-left key, the same one used to open Dasher. Dasher text can be transferred back to the GazeTalk-J text-field by activating the upper-left key, "Text to GazeTalk-J". Returning to the normal GazeTalk-J interface, the user will then be able to continue writing and editing, and this text can be used for e-mails, documents, or type-to-talk text (Figure 8**c**).

The latest Japanese version of Dasher for Windows partly implements a Kana-Kanji conversion function. Its basic principle is that Dasher uses a training corpus, in which texts are described in Hiragana characters followed by the corresponding Kanji/Katakana/alpha-numeric characters. For example, the training sentence メロスは、村の牧人である。is described in the training corpus as follows:

｜めろす＞メロスは、｜むら＞村の｜ぼくじん＞牧人である。｜

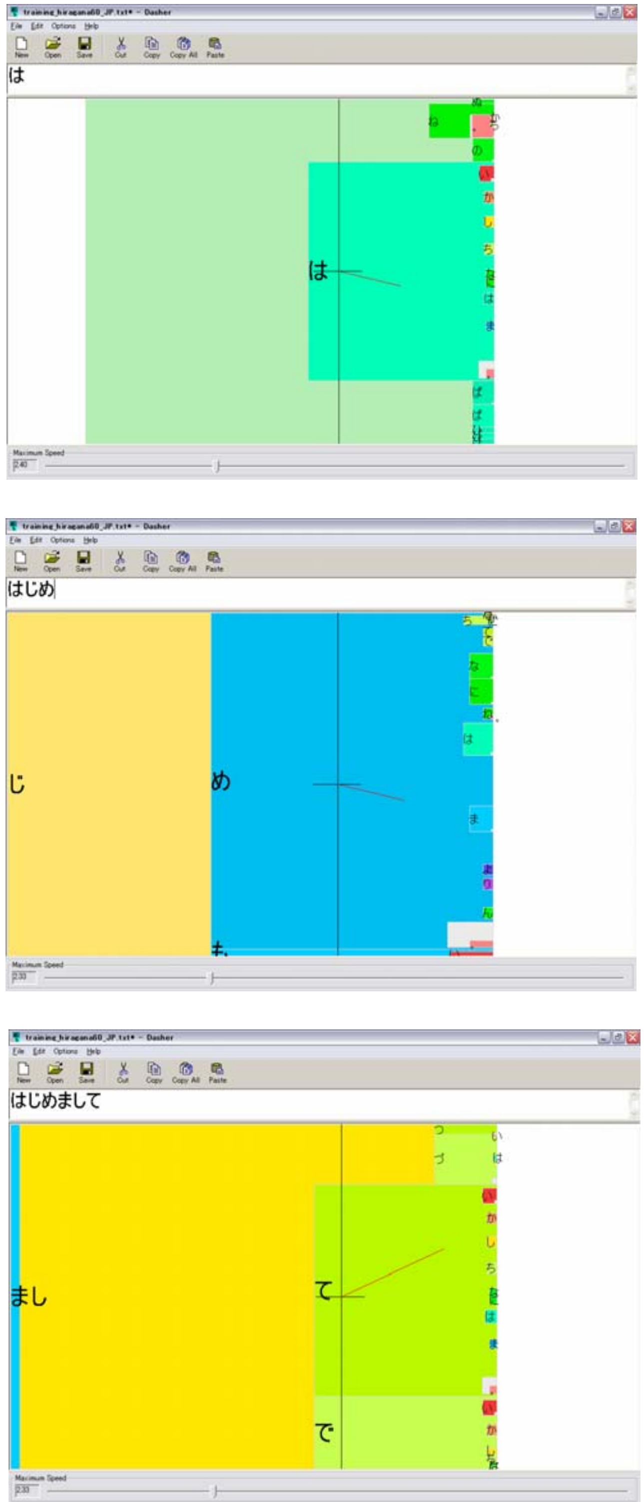
In the training corpus, each pause between clauses is indicated by ｜. In addition, each word that should be represented in Kanji/Katakana/alpha-numeric characters is indicated by ＞, with its Hiragana representation. Using these training texts, including words both before and after their conversion, Dasher calculates the probability of the corresponding texts.

Three challenges remain for the Kana-Kanji conversion in the Windows version of Dasher, although the Unix version has successfully overcome them:

1. *Appearance of converted words are highly dependent on the training corpus.* A correctly converted word can be shown in Dasher only when exactly the same word was included in the training corpus. For example, the correct conversion for the word むら, found in the above text, may or may not be 村 in a sentence a user types. The correct conversion could be any of the following: むら, ムラ, 屯, 邑, 邨, 群, 斑, 武良, 叢, etc. In general, any word represented in Hiragana characters can be converted into many different representations, depending mainly on the context of the specific sentence. And yet if the word that a user needs is not included in the training corpus, it is never shown in the text-input field in Dasher. This frequently results in misrepresentation of the typed words represented in Hiragana characters, and frustration for the user.

2. *Extensive probability calculations are necessary.* As mentioned above, any word written in Hiragana characters can have many possible representations, for which different Kanji characters would be needed. This means that the training corpus must include sufficient Kanji characters (at least 1,000 different characters) for Dasher to be able to provide appropriate conversion alternatives. This results in a need for a vast amount of probability calculations in contrast to the English version, and may noticeably retard Dasher's operating time.

3. *Redundant pre-conversion text is not removed from text field automatically.* The text field retains text from both before and after conversion. This means that a user must manually delete each vestigial pre-conversion sentence or word after exporting the text in other systems. This process can be quite time-consuming and is often annoying.



a

b

c

Figure 7. The Dasher interface.

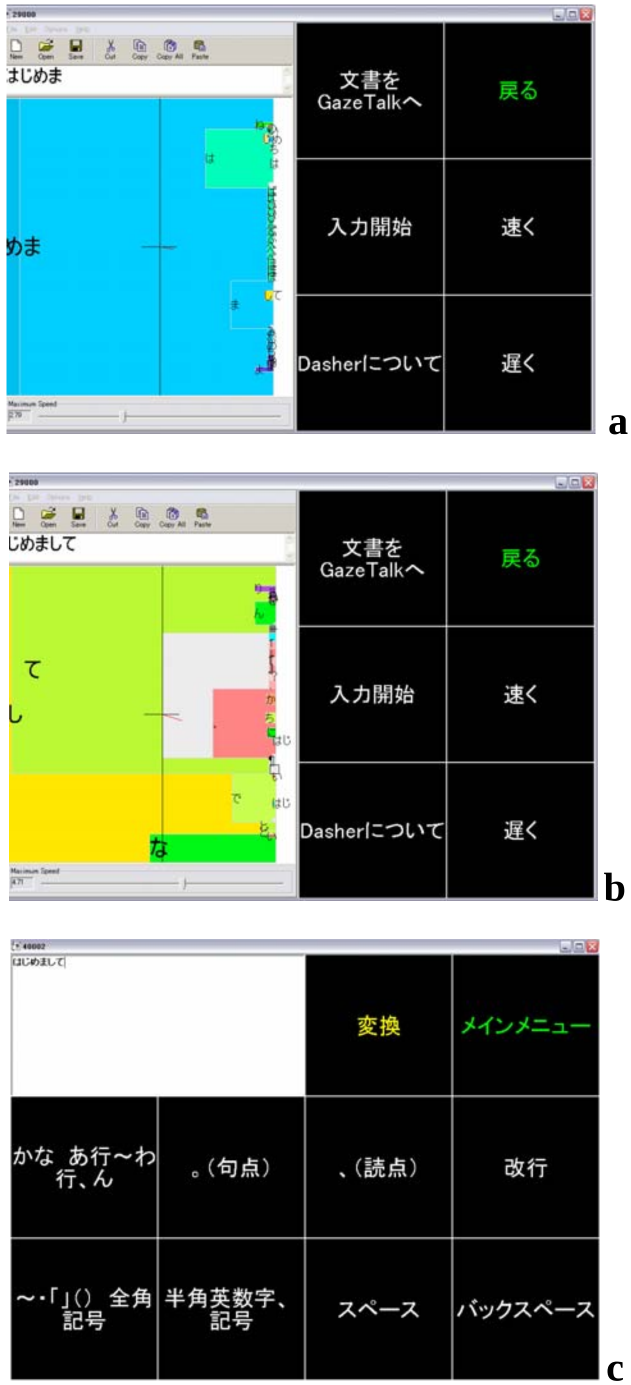


Figure 8. Dasher integrated with GazeTalk-J.

2 Dasher in Korean

2.1 The Korean language system

Written Korean (Hangul) is a beautifully logical phonemic language. Each hangul character represents a single syllable, and is made up of two or three letters called jamo. There are 24 jamo letters: 14 consonants, and 10 vowels. The first letter in a hangul character is one of the 14 consonants. The second letter is one of the vowels. If there is a third letter, it is one of the consonants. Some five of the consonants exist in ‘double’ form.

In Unicode, any of the Hangul characters can be created from 67 ‘combining’ jamo characters. Why 67 – why not 24? There are 5 double consonants, which get their own jamo character in unicode; and every consonant is available in an ‘initial’ form and a ‘final’ form. Plus there are about 9 extra final consonants that represent consonant-pairs like ‘lm’, ‘nj’, and ‘lh’. Plus instead of having just 10 vowels, there are 21 vowels, including various diphthongs. Figure 9 shows the 67 combining jamo characters as rendered by Dasher.

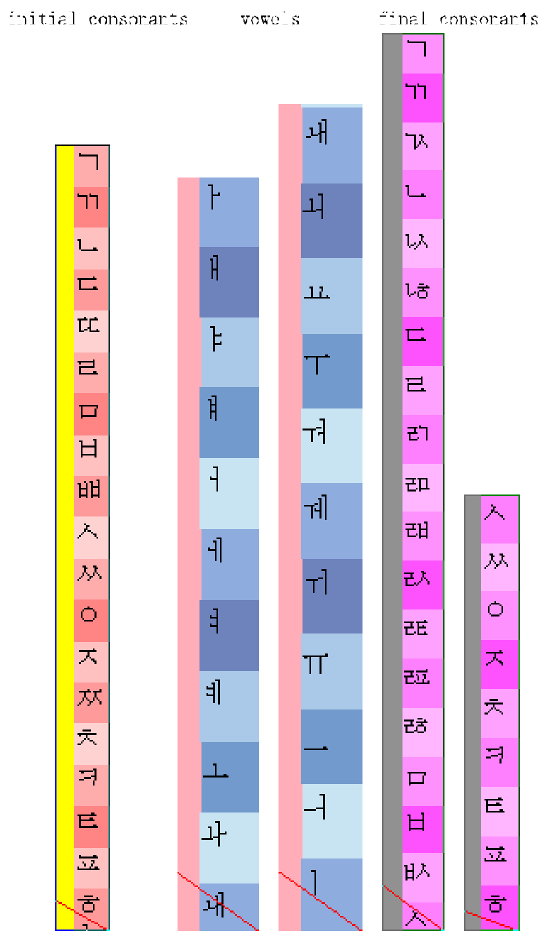


Figure 9. The 67 jamo characters in Unicode.

2.2 Text-entry procedure in Dasher

There are two ways to use Dasher in Korean. The recommended way is to write using these 67 ‘combining’ jamo characters. It is also possible to write directly in the Hangul alphabet, but this is not recommended as the alphabet consists of about 11,000 characters, and Dasher’s performance with such a large alphabet is likely to be poor.

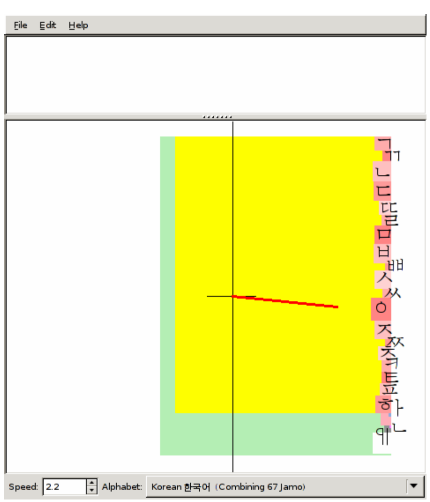
In Dasher’s default ‘European/Asian’ colour scheme, Korean writing proceeds as follows:

1. Every new hangul character is initiated by entering a yellow box; this box contains all the initial consonants (*choseong*).
2. The initial consonants are coloured sand/pink. After the initial consonant, you must enter a vowel (*jungseong*); the vowels are in blue boxes.
3. After the vowel, you may either start a new hangul syllable by entering the yellow box, or add a final consonant (*jongseong*) found in the grey box; final consonants are contained in magenta-coloured boxes.
4. Another option after the vowel is entered, or after a final consonant is entered, is to enter a non-jamo character such as a space character (shown by the box in a white square).

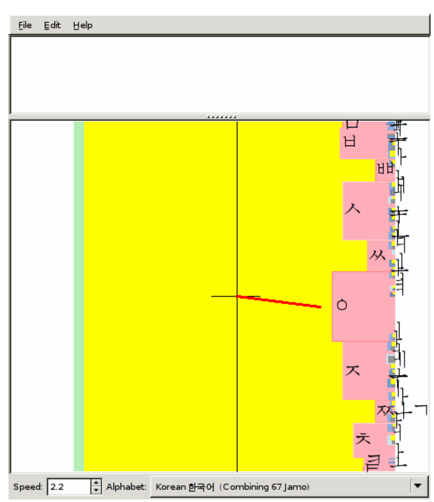
It is recommended that the Dasher option ‘display box outlines’ is switched off when using Dasher in Korean (because the group boundaries between initial consonants and vowels might otherwise be found distracting). You may also find that the user experience is improved by turning the Language→Smoothing parameter down from its default value of 5.0.

2.3 Screenshots

Korean works beautifully on Linux systems, thanks to the excellent Pango rendering engine; the following screenshots (Figure 10 A-N) are made on an Ubuntu Linux machine. As of 2005, Microsoft Windows did not provide such good font support for Korean, so Phil Cowans had to add a special workaround to get Korean to look good on Windows. Here are screenshots showing writing ‘hello’ in Korean.

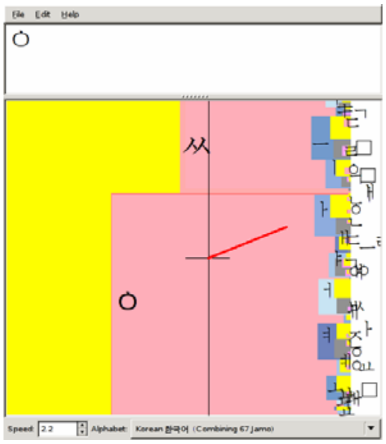


A. Initial Dasher view.

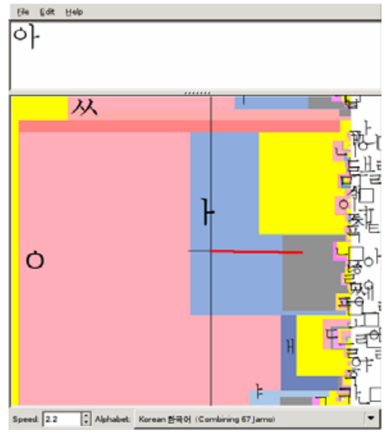


B. Entering the first consonant.

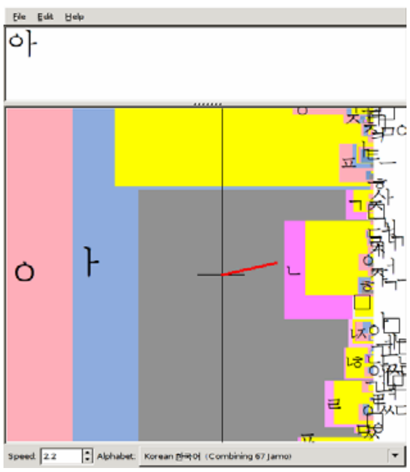
Figure 10. Screenshots of Korean version of Dasher (A-B).



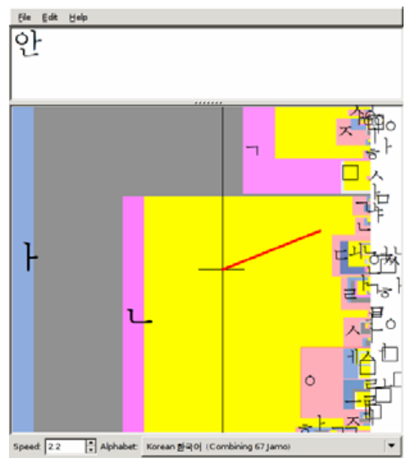
C. Choosing the vowel.



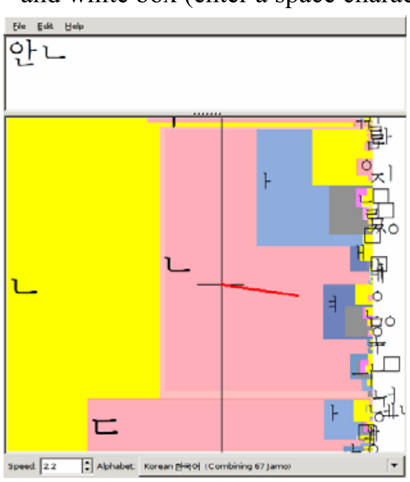
D. As we enter the vowel, we see two main options: yellow box – start a new syllable; grey box – add a final consonant to this syllable. Here, we choose the grey box.



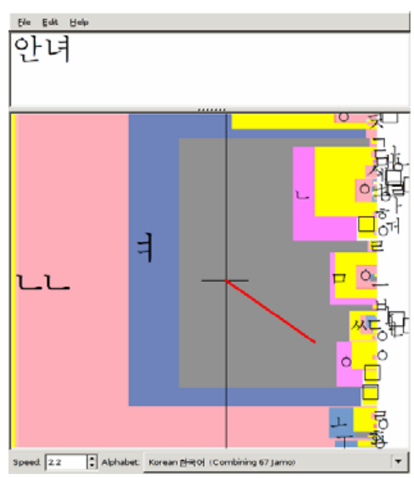
E. When we enter the final consonant 'n', the two probable options inside its magenta box are: yellow box (start a new syllable) and white box (enter a space character).



F. We start a new syllable with another 'n'.

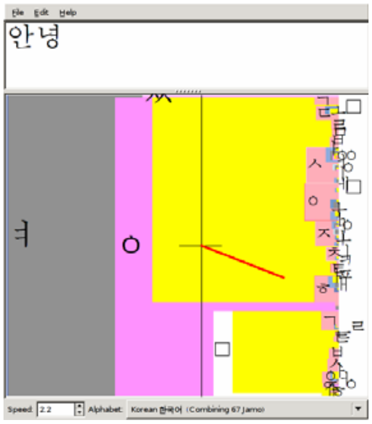


G. Choose the next vowel

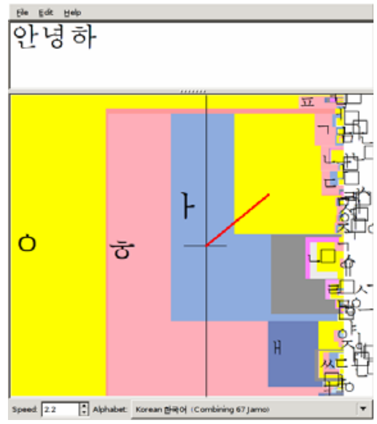


H. Choose the final consonant

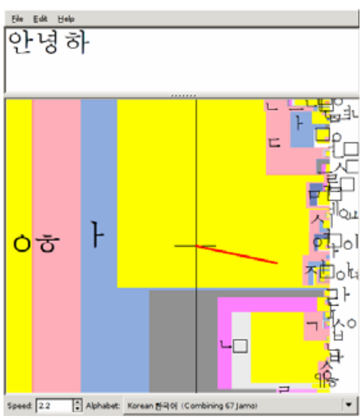
Figure 10. Screenshots of Korean version of Dasher (C-H).



I. Enter the yellow box to start the next syllable

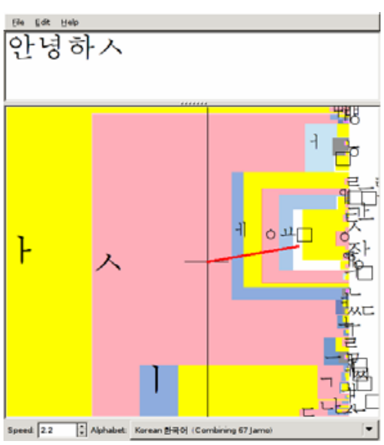


J. Grab the consonant and vowel; again we can either start the next syllable by entering the yellow box or add a final consonant by going in the grey box. This time we go in the yellow box, as the syllable is complete.

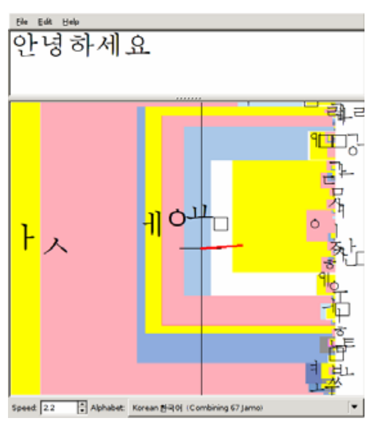


K. As we select the leading consonant ‘s’…

L. … and the next vowel…



M. the language model does a good job of predicting the entire next syllable.



N. As this syllable (which ends the word we are writing) is completed, the white space characters (space and newline) are both very probable.

Figure 10. Screenshots of Korean version of Dasher (I-N).

2.4 Training texts

We gratefully acknowledge the help of KAIST BOLA who provided a 258-kbyte corpus (Choi, 2001) which was used to make the screen-shots. This corpus is available for research purposes but may not be freely distributed. At present we have very little public-domain training text. We would very much like to receive corpuses of freely-distributable Korean text. We are aware that there are several Korean writing styles, and ideally we would like to have a separate training text for each style. Please send corpora to David MacKay (david.mackay@cogain.org). UTF8 is the preferred plain text format.

3 Experimental examination of GazeTalk-J: Evaluation of users' performance

3.1 Objective

The objective of this section is to evaluate GazeTalk-J's usability—and especially learnability—by studying users' scan-paths (Aoki et al. 2006). A scan-path is the connected consecutive fixations in the user's eye-movement sequence. In other words, the scan-path directly represents the user's process of selecting and activating a desired key. For our analysis, we adopted the metric "Cumulative deviation from the most efficient scan-path" to represent the degree that gaze behaviour differs from the calculated most efficient way. We evaluated GazeTalk-J's learnability by tracing each user's scan-path efficiency estimated by the metric. We discuss our results in terms of the learnability of the gaze interface, and the characteristics of users' gaze behaviour in the early stages of learning.

3.2 Scan-path metric

Figure 11 describes the procedure for calculating the scan-path metric, including the three scan-path options for an example GazeTalk-J scenario. The "initial position" is the user's fixation position immediately following the completion of activation of the previous key. The new "target key" is the most appropriate key for the next key activation.

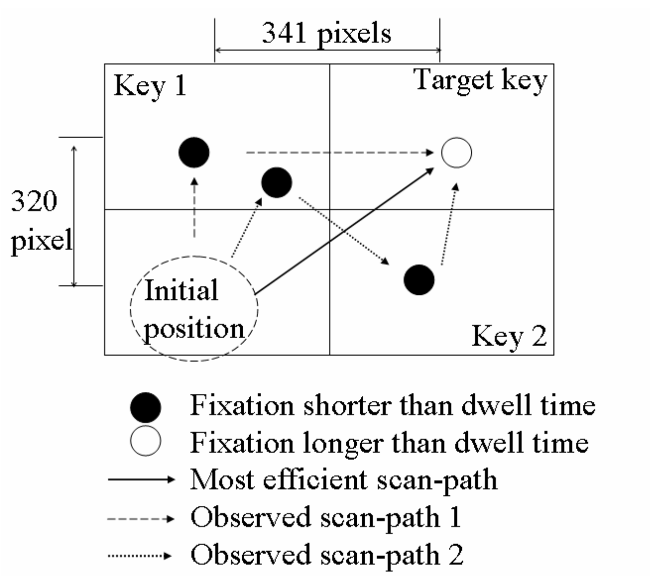


Figure 11. Example calculations of scan-path metric.

The solid-line scan-path in Figure 9 represents the most efficient, ideal path to the target key—whereby movement from the initial to the target position is direct. The dashed-line scan-path is also a relatively smooth eye movement towards the target key. In contrast, the dotted-line scan-path represents inefficient eye movement. This user's gaze is directed at unnecessary keys first, before finally settling on the target key.

Our scan-path metric, "Cumulative deviation from the most efficient scan-path" (Cd) represents the difference between a user's scan-path and the most efficient, ideal one. The Cd is calculated by summing the distances between each fixation and the current target key. The distances are measured in pixels, with the central pixels of the scan-path keys serving as measuring posts. The Cd for the two less-efficient scan-paths in Figure 9 were thus 341 and 661 pixels (= 341 + 320), respectively.

The rationale for choosing keys' centre pixels to describe these measurements is twofold. First, nearly all users gaze naturally at the centre of a key because that is where the characters themselves are always displayed. Secondly, the size of the character begins to shrink as the user continues to gaze at it—a feedback service, letting the user know activation is in progress—and this shrinking encourages the user's gaze to fix more and more precisely on the exact centre of the key. This makes a key's central point a logical post on which to base our distance measurements.

Dwell time and overall typing time are not taken into account in this metric. In this study, any fixation longer than 100 milliseconds (msec) is counted as a fixation.

3.3 Experiment

Subjects. Eight Japanese students participated in our seven-day experiment. They were selected from among university students who had never used any type of gaze interface before. Ranging in age from 18 to 22 years, three were female, and five male. All of them had normal or corrected-to-normal vision. For their participation, each was paid 5,000 Japanese yen (approx. 42 U.S. dollars). As an added incentive, in the interests of our longer-term experiment, we offered an additional 5,000 yen to the top three performers in terms of typing speed and accuracy.

Task. The task was to type a Japanese text as quickly and accurately as possible using GazeTalk-J. Subjects first heard a target sentence spoken aloud by an experimenter, which they then were to type. As they typed, the same sentence was shown on their screen near the GazeTalk-J window. The subjects were asked to compose the sentence using the appropriate combination of character systems, and using the conversion function. In addition, they were asked to correct every typing error they happened to notice. The sentences used in the experiment were selected from the Japanese translation of H.C. Andersen's fairy tales (children's edition) so that they could be easily understood.

Apparatus. The gaze-typing system was run on a personal computer (933-MHz CPU) operating with Windows 2000, which included the IME Kana-Kanji conversion program and a 17-inch colour monitor (1024 × 768 pixels). The viewing distance between the subject and the screen was approximately 60 cm. The typing system was equipped with a logging facility that automatically recorded keys typed, fixation durations, and more.

The dwell time for key activation was set at 500 msec. A QuickGlance system (EyeTech Digital Systems) was combined with the typing system as an eye-tracking device with a tuning of 15 for the update-rate and 7 for the smoothing factor.

Procedure. Each subject was given 22 experimental blocks of typing in total. The experiment ran over seven days, but we tried to keep each subject's work days as close together as possible, preferably over consecutive days, with a four-day pause the maximum allowed. Each block consisted of five sentences, for a total of approximately 90 characters. We designed each experimental "day" to involve no more than an hour and a half of work per subject, as we expected the time needed to complete a block to decrease as the subjects' learning increased. The number of blocks given for Days 1, 2, and 3 were thereby one, two and three, respectively. For each of the following days, the subject performed four blocks of typing. Subjects were allowed to take short breaks of approximately five minutes between successive blocks.

On Day 1 subjects were instructed on how to perform the typing tasks. The typing system itself was also explained, as were the experimental procedure and how to calibrate of the eye-tracking device. Subjects were then given a five-minute gaze-typing demonstration performed by an experimenter. This was followed by a training session during which the subject was given a few sentences to type with a mouse—not the gaze interface—using the same menu system as for the experimental typing blocks. In this way, the subjects obtained the prerequisite understanding of the hierarchical menu structure, but no training in actual gaze interaction.

Calibration of the eye-tracking system was done at the beginning of each new block. After typing the five Japanese sentences in the block, the subject was asked to rate, on a five-point scale, the usability attributes of the typing system as well as the subject's perception of errors in the sentences.

3.4 Results

As is well known among eye-tracking researchers, the transcription of eye-tracking data is greatly time-consuming. To calculate the scan-path metric, every "target key" for any given moment must be identified, which can change both dynamically as well as frequently depending on typing processes. As a result, we have not yet completed data analysis for all of our subjects. We do present preliminary results obtained from three subjects (S1–S3) out of the eight.

Transitions for the scan-path metrics (Cds) for each subject's sentences are shown in Figure 12, and the mean Cds are presented in Table 1. Cds were calculated on a per-sentence basis. As for S3, the Cds for two sentences in experimental block 13, as well as all S3's sentences from experimental blocks 14–22, are not shown because they have not been obtained yet. The graphs in Figure 12 also show the plots for characters typed per minute (CPM), which were calculated on a per-block basis.

As shown in Figure 12, all three subjects experienced a rapid decrease in Cd during the first three to four experimental blocks; this decrease was found to continue until approximately the 10th block. This seems to indicate that novice users can achieve relatively efficient and stable eye movement for the purpose of gaze-typing after approximately 50 sentences of practice, or 10 experimental blocks.

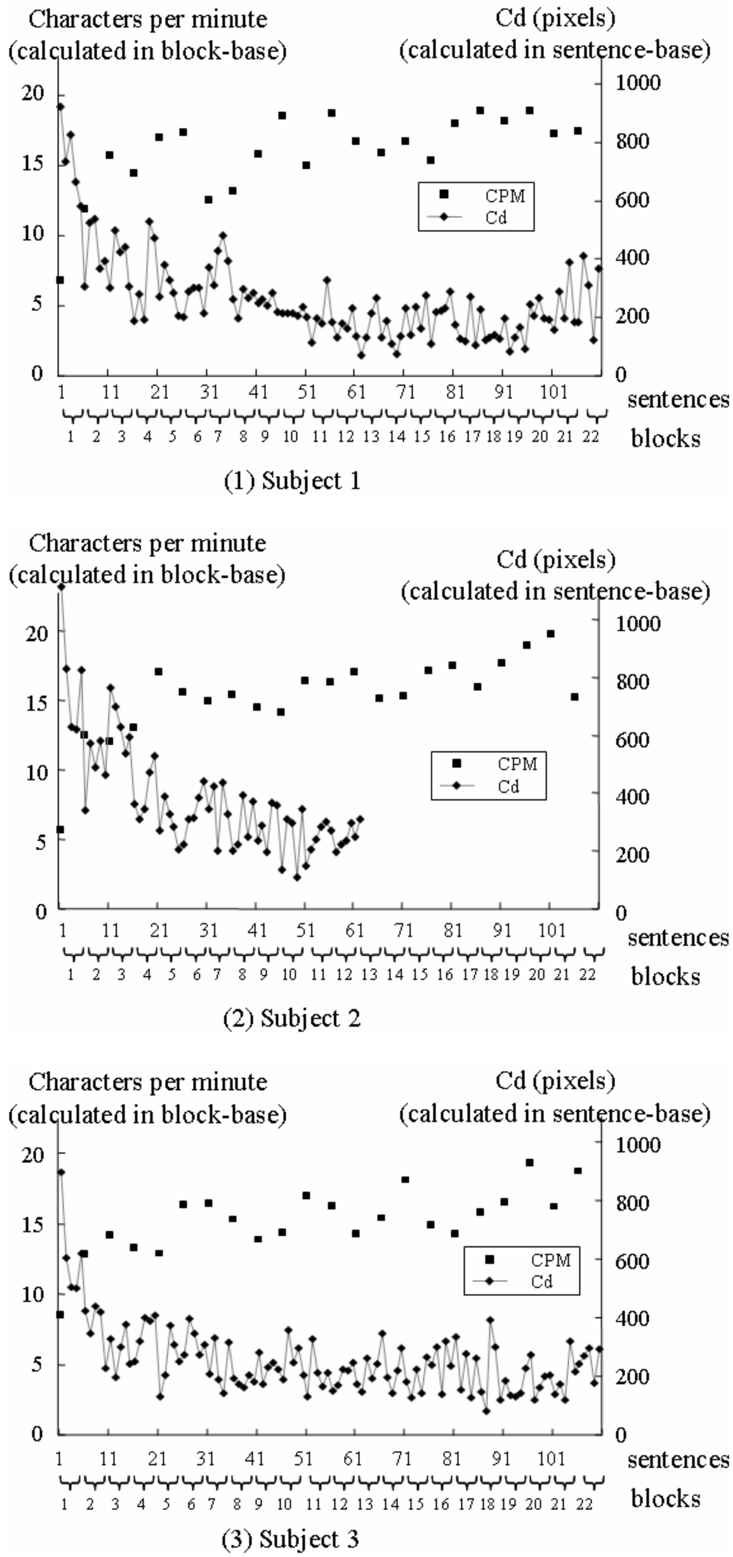


Figure 12. Metric transition for the three subjects.

Table 1. Mean Cds for the three subjects.

Subject	Mean blocks 1-12	Mean blocks 13-22	Grand mean
S1	293.95	212.06	256.73
S2	384.10	-	380.75
S3	328.91	189.70	265.63

To compare learnability in terms of both Cd and typing speed (1/CPM), we fitted each subject's transitions to learning curves, for both units. For the learning curve, we adopted the "power law of practice" model, represented by $t_n = t_1 n - \alpha$, where n is the number of sentences (Cd) or blocks (1/CPM) practiced, t_n = Cd or 1/CPM at n, and α is the learning coefficient. Table 2 lists the parameters estimated by regression analysis, in which the least-squares error method was applied. As can be seen in this table, the learning models for each subject are highly significant.

Comparing the learning coefficients in Cd and 1/CPM, we can see the same tendency: the learning coefficients in Cd are higher than those in 1/CPM for all subjects. This tendency indicates that all the subjects were able to adapt their eye movement to gaze typing at a relatively early stage. Though typing speed is highly dependent on efficient eye movement, it seems other factors may also have had an influence on typing speed, probably design factors such as key allocation and dwell time.

Table 2. Results of regression analysis in terms of Cd and 1/CPM.

Sub.	*Regression analysis on Cd* Learning coefficient	t_1	R^2	F_0	*Regression analysis on 1/CPM* Learning coefficient	R^2	$F_0(1,20)$
S1	0.38	966.04	0.49	105.23**‡	0.22	0.61	31.88**
S2	0.40	1201.63	0.60	88.65**†	0.25	0.68	41.91**
S3	0.28	672.55	0.41	74.88**‡	0.17	0.64	35.24**

†$F_0(1,60)$, ‡$F_0(1,108)$, ** $p<0.01$

3.5 Discussion and summary of the first experiment

We analysed the learnability of GazeTalk-J by applying a new metric, Cd. From the eye-tracking data obtained from our seven-day learning experiment, we observed that the three subjects, who had never used any kind of gaze interface before, initially showed inefficient eye movement for typing, but they improved very quickly within 3–4 experimental blocks, or 15–20 sentences, of gaze-typing. They also showed continuous improvement in eye-movement efficiency until approximately the 10th experimental block (50 sentences, or six hours of typing). Combined with the results obtained by Aoki et al. (2005), our result seems to be in support of Bates (2002): people need some practice to learn to use gaze interaction, but eventually they will master it. By comparing the improvement in eye-movement (Cd) with improvement in typing speed (1/CPM), we observed that the learning coefficients estimated from Cd are higher compared with 1/CPM. From this we inferred that, after a brief learning period in which skill in efficient gaze typing was acquired, other factors (e.g., design factors such as key allocation, dwell time, and so forth) had a great influence on the continued improvement in users' overall performance.

4 Comparative evaluation of GazeTalk-J and Dasher

4.1 Objective

In this section (based on manuscript for Itoh et al. 2006) we evaluate typing performance with GazeTalk-J by comparing it with Dasher (Ward, 2001; Ward and MacKay, 2002). We also compare GazeTalk-J with another interface that is a derivative of GazeTalk-J. The derivative version of GazeTalk-J has exactly the same functionality as the original GazeTalk-J, the only difference being that the text field is located in the centre of the display, as shown in Figure 13.

We conducted an experiment in which subjects performed seven 300-character typing blocks with each system. Several performance indices, including typing speed and error-related frequencies such as over-production rate, were calculated from the experimental data. Of the usability attributes, learnability, efficiency, and error were derived from these performance indices for each interface. Subjective satisfaction was also analysed using questionnaire responses collected during the experiment. With the comparative usability results gained from this experiment, we discuss the effects some design components have had on them.

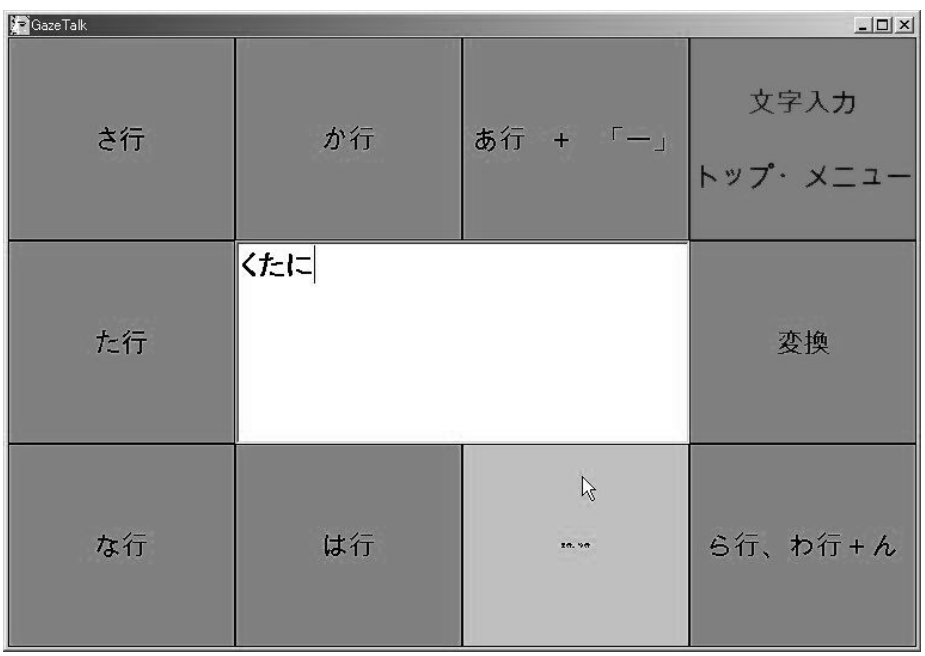


Figure 13. The derivative version of GazeTalk-J.

4.2 Experiment

Subjects. Fifteen Japanese students participated in the experiment. Their mean age was 21 years (ranged 19–25 years). Subjects were divided into three groups, each of which performed experimental trials with one of the typing systems mentioned above: GazeTalk-J, the derivative version of GazeTalk-J, and Dasher. To obtain a homogeneous allocation of the subjects to the three typing conditions, we conducted a preliminary test with another Japanese gaze-typing system, Hearty Ladder, which was combined with a conventional on-screen keyboard of Hiragana characters; we were thereby able to estimate subjects' initial gaze-typing skills. Each subject was paid 750 Japanese yen (~6 U.S. dollars) per hour for participation in the experiment. To maintain a high motivation for participation in the four-day typing trials, we informed subjects before starting the experiment that the top two typists for each typing-system group (measured in terms of both typing speed and accuracy) would receive an extra prize: 5,000 yen (~$40) to the best typist, and 3,000 yen (~$25) to the second-best.

Task. The experimental task was to gaze-type Japanese phrases and short sentences of everyday conversation using Hiragana characters only, simulating ALS patients' use of the system. The task design was based on the fact that everyday communication of this kind need not use Kanji characters. Some example sentences might be "Give me some water", "Turn on the TV set", and "What time is it?" Each sentence was made up of 8–23 characters (15 on average), including Hiragana characters and numbers (in numeric form). The task was performed one block at a time. Each block comprised 20 sentences, each of which contained approximately 300 characters. Subjects were instructed to type these sentences as quickly and accurately as possible. For the first ten sentences of each block, the eye tracker was calibrated using the standard calibration procedure to obtain good tracking accuracy, and the remaining ten sentences were typed with a deliberate miscalibration to simulate a low-resolution eye tracker.

The test supervisor read each sentence aloud for the participants. Long sentences were divided into shorter parts. Subjects were instructed to request repetitions as necessary. We also instructed them not to correct typing errors discovered later (at the time of text verification).

Apparatus. The gaze-typing system was run on a personal computer (933-MHz CPU) operating with Windows 2000, which included the IME Kana-Kanji conversion programme and a 17-inch colour monitor (1024 × 768 pixels). The viewing distance between the subject and the screen was 70 cm. The dwell time for key activation in both systems initially was set at 500 msec, and the subject could change it according to his or her preference in any experimental block. We set an initial speed parameter for Dasher of 1.5 (maximum speed), and subjects could also adjust that setting as they wished. For this study we chose the "Hiragana 60" version of Dasher (version 3.2.11)—which features a set of 60 Hiragana characters, plus numerals and special Japanese symbols—and used it with a 380,000-Hiragana-character training corpus of everyday phrases. We did not apply the self-correction option that Dasher features for inaccurate input in this experiment. A QuickGlance system (EyeTech Digital System, version 3.1) was used as an eye-tracking device, set to 15 for the update rate and to 7 for the smoothing factor.

Procedure. We focused specifically on the following three experimental factors: (1) the typing systems (inter-subject), (2) accuracy of the eye-tracking systems (intra-subject), and (3) learnability (intra-subject). For the first factor, we used the three gaze-typing systems GazeTalk-J, the derivative version of GazeTalk-J, and Dasher. The second factor was controlled by special calibration of the eye tracker: high accuracy was produced by calibrating the tracker using the standard QuickGlance procedure, whereas low accuracy was achieved deliberately by applying a miscalibration procedure in which each fixation point to be calibrated was distorted slightly from the real target—specifically, with a 2-degree error in one of four randomly selected directions, up, down, left, or right. Table 3 lists the actual calibration errors for each of the 16 QuickGlance calibration points in terms of mean, standard deviation, minimum, and maximum values, as calculated from the data of the 15 subjects for seven experimental blocks. This intentional miscalibration produced mean errors of 1.07–1.50 degrees (approximately 0.5–0.6 degrees σ).

Table 3. Actual eye-tracking errors (in degrees) with deliberate miscalibration.

μ=1.25 σ=0.51 min=0.30 max=2.82	1.07 0.45 0.15 2.64	1.07 0.49 0.19 2.45	1.10 0.54 0.21 2.58
1.31 0.57 0.00 2.81	1.45 0.54 0.16 2.78	1.39 0.54 0.37 2.68	1.31 0.54 0.27 2.65
1.12 0.61 0.23 2.67	1.38 0.60 0.00 2.69	1.49 0.60 0.00 2.60	1.35 0.49 0.29 2.75
1.35 0.50 0.22 2.85	1.50 0.52 0.29 2.75	1.34 0.51 0.00 2.66	1.32 0.52 0.27 2.70

Note: For aggregated values of 16 calibration points: μ = 1.30, σ = 0.55, max = 2.85, min = 0.00. Each cell corresponds to a calibration point on the display.

Learnability was examined in terms of transitions during all the seven experimental blocks, and in terms of the differences noted between several earlier blocks, for the performance indices described below. The experiment involved each subject for four days, requiring at most one and a half hours of the subject's time per day. Approximately one week prior to the experimental session, the preliminary test was conducted for the purpose of allocating subjects to the three typing systems.

On Day 1, before the experimental session, a subject performed a 10-minute training session with the typing system that he or she would be using in the experiment. Afterwards the subject performed one experimental block, which included gaze-typing ten sentences with high eye-tracking accuracy followed by ten sentences with low accuracy, as described above. A short break (~5 min.) separated the high- and low-accuracy sessions. At the end of Day 1, subjects filled in a questionnaire that asked for their opinions on the system they had used in the experiment.

On Days 2–4, subjects first received a five-minute warm-up, after which they performed two blocks of the typing task. At the end of Day 4, they were asked to fill in the same questionnaire as on Day 1.

Analysed measures. This section focuses on four of the usability attributes suggested by Nielsen (1994): learnability, efficiency, error, and satisfaction. (Memorability was not included in the scope of this study.) We examined these four attributes and their inter-associations for each gaze-typing interface in terms of usability-related indices derived from data concerning typing speed, errors, and subjectively stated attitudes.

As mentioned previously, typing speed is typically measured in terms of words or characters per minute. Text composed in Japanese is best measured in CPM, whereas WPM is applied to European languages. It is difficult to set an exact conversion factor between WPM and CPM, but from our previous experiments with identical text in the English and Japanese versions, a factor of 2 seems feasible.

Interface efficiency can be examined in either CPM or WPM as carried out by expert or skilled users. Efficiency may also be estimated in terms of optimal speed in a perfect-user simulation or using a learning model. In this study, we evaluated the efficiency of the gaze-typing interfaces in terms of CPM after sufficient trials estimated by learning models.

The learnability of an interface can be evaluated in terms of either a learning factor, identified by applying the power law of practice model to experimental data, or the differential rate of typing speed measured at two different times in an earlier stage of practice. In addition to the learning factor and the differential rate, in this study, mean CPM in several earlier experimental blocks was compared among the gaze-typing systems to evaluate their learnability.

As error-related measures, several indices—such as over-production rate, rate of backspacing, and minimum string distance (MSD) (Soukoreff and MacKenzie, 2003)—have been suggested in addition to the rate or frequency of errors per character or unit time. The MSD is a sentence-based error measure based on how many key-manipulation steps one needs to take to arrive at the target sentence from the sentence that was actually typed (including any errors). In the performance data collected from our experiment, almost all the sentences were error-free (with or without correction) during gaze-typing, and therefore it would be impractical to apply MSD to our data. The over-production rate is referred to as the rate of the actual number of (gaze) selections over the optimal (least) number of selections needed for constructing a given sentence. This index is particularly useful when examining the frequency of mistyping when using a hierarchical-menu system like GazeTalk-J. It is not possible to measure the number of single selections in Dasher, since Dasher is operated by continuous navigation and not by single selections; however, the rate of backspacing can be calculated by dividing the total number of backspace-key activations (or the total number of characters erased prior to the cursor position) by the total number of typed characters. We used the over-production rate primarily, along with the backspacing rate, to evaluate systems' learnability.

From subjective opinions of interface design, we evaluated not only subjects' satisfaction, but also task-performance aspects of speed and error rate that were perceived by users. Question items related to these aspects were described with a five-point semantic differential scale—a pair of terms with opposite meanings, such as "very fast" and "very slow". The following subjectively rated items were included in the questionnaire: perceived typing speed, perceived likelihood of error, interface preference, satisfaction with the system, perceived fatigue, and uncomfortable feeling of motion sickness.

4.3 Results

Typing speed. Experimental data on typing speed (CPM) were analysed with a three-way ANOVA test, with the typing systems (GazeTalk-J, the derivative version of GazeTalk-J, and Dasher), eye-tracking accuracy (low and high), and learnability (blocks 1–7) as the independent variables. Subjects were treated as repetitions. The results of the ANOVA are given in Table 5. There was a significant difference in CPM between the three typing systems. The standard version of GazeTalk-J exhibited significant and slightly better performance in typing speed compared with the other two systems. There were no significant differences between the derivative version of GazeTalk-J and Dasher. The grand mean typing speed was 23.3 CPM ($\sigma = 4.0$). Mean CPM for GazeTalk-J was 24.2 ($\sigma = 3.9$), 22.7 CPM for the derivative version of GazeTalk-J ($\sigma = 2.9$), and 23.0 CPM for Dasher ($\sigma = 5.1$).

Table 5. Result of ANOVA on CPM.

Factor	s.s.	d.f.	*V*	F_0
System (A)	84.8	2	42.4	3.49*
Accuracy (B)	8.5	1	8.5	0.70
Block (C)	1000.2	6	166.7	13.73**
A × B	3.7	2	1.9	0.15
A × C	62.0	12	5.2	0.43
B × C	146.9	6	24.5	2.02
A × B × C	212.9	12	18.3	1.51
Error	2039.6	168	12.1	
Total	3565.6	209		

* $p < 0.05$.
** $p < 0.01$.

As subjects' typing speed increased with each block for each system, we note a significant learning effect for the seven blocks. The learnability for each typing system is depicted in Figure 14. Quantitative estimation of the learning effect on typing speed was made by applying the power law of practice learning model to data from individual subjects. Table 6 gives the results of parameter estimation of the learning model, based on the typing systems, for all subjects. We observed that the power law of practice seemed well-fitted to all subjects using GazeTalk-J and the derivative version of GazeTalk-J, whereas only two out of the five Dasher users exhibited learning effects that could be explained well by this model. The learning factors—which themselves may represent a measure of learnability—are alike and reasonably high (about 1.6 on average) for the two versions of GazeTalk-J. On the other hand, the learning effect for those Dasher users for whom the power law of practice was well-fitted to their CPM data was very high, even as compared with the best-learned users of GazeTalk-J.

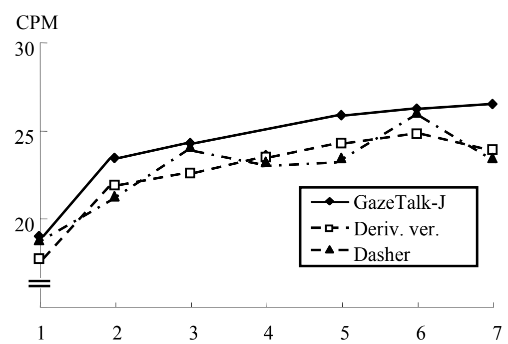


Figure 14. CPM transitions per experimental block for each typing system.

Table 6. Parameter estimation of the power law of practice on CPM for each typing system.

System	Sub.	$a^{\dagger}$	$b^{\dagger\dagger}$	R^2	F_0
GazeTalk-J	S1	0.047	-0.159	0.761	15.89*
	S2	0.044	-0.122	0.985	332.86**
	S3	0.056	-0.121	0.598	7.43*
	S4	0.047	-0.195	0.892	41.53**
	S5	0.063	-0.202	0.931	67.97**
Mean learning factor			-0.160		
Deriv.ver. of GazeTalk-J	S6	0.052	-0.119	0.720	12.88*
	S7	0.051	-0.178	0.891	40.97**
	S8	0.057	-0.149	0.668	10.04*
	S9	0.052	-0.119	0.644	9.06*
	S10	0.058	-0.229	0.754	15.35*
Mean learning factor			-0.159		
Dasher	S11	0.079	-0.332	0.884	38.01
	S12	0.041	-0.066	0.241	1.59
	S13	0.064	-0.289	0.697	11.48*
	S14	0.040	-0.049	0.231	1.50
	S15	0.059	-0.045	0.257	2.08
Mean learning factor			-0.156 -0.310‡		

†, †† Parameters of the power law of practice: $y = ax^b$, where x = blocks practiced (ca. 300 characters/block), and y = typing time per character = 1/CPM (min./character).

‡ Mean obtained only from subjects having a significant learning effect.

* $p < 0.05$.

** $p < 0.001$.

Subjects' typing speed, which is related to efficiency, can be estimated after a certain number of typing blocks have been completed using the power law of practice models. Table 7 lists the estimated CPM of the subject with the greatest learning factor, showing the estimation from each typing-system during the 7th, 10th, 20th, and 50th blocks.

The best typing speed for Dasher was estimated to catch up with that of GazeTalk-J at approximately 34 CPM after 20 blocks of typing, which corresponds to gaze-typing about 6,000 characters, or nearly four hours of gaze-typing. In subsequent trials, the best typing speed for Dasher was then expected to exceed that for GazeTalk-J. The learning model estimated an increased typing speed with Dasher at 46 CPM, which might be equivalent to approximately 23 WPM for European languages. (As mentioned previously, we used a conventional factor of exchange between Japanese CPM and English WPM of roughly 2). The learning model also estimated an increased typing speed with the derivative version of GazeTalk-J at 42 CPM after 50 blocks of typing (or 15,000 characters).

Table 7. Estimated CPM of the best-learned user for each typing system at various points in time.

	Experiment		Estimated by models			
	***n*=1**	***n*=7**	***n*=7**	***n*=10**	***n*=20**	***n*=50**
GazeTalk-J	15.6	23.6	23.5	25.3	29.1	35.0
Deriv.ver.	15.4	24.9	26.7	29.0	34.0	41.9
Dasher	12.8	21.0	24.0	27.1	34.1	46.2

Note: n = number of blocks typed (each block consists of ~300 characters), e.g., n = 50 indicates CPM after 15,000 characters typed.

Typing errors: Over-production rate. The result of a 3-way ANOVA for the over-production rate is shown in Table 8. This index can not be applied to Dasher, and therefore only two versions of GazeTalk-J were analysed. The only significant difference was observed between the blocks, and there were no significant effects for any other factors. As depicted in Figure 15, the over-production rate of each version of the GazeTalk-J gradually decreased with each block. In particular, the learning effect on this index is seen until Block 5, and subsequently the rate seems to become constant.

Typing errors: Backspacing rate. The ANOVA result for the rate of backspacing with the same three factors is shown in Table 9. For this error-related index, a significant difference was observed between the three typing systems. In particular, as can be seen in Figure 16, the backspacing rate for Dasher was far higher than for GazeTalk-J. The grand mean was 0.037 (per typed character; $\sigma = 0.032$). The mean for GazeTalk-J was 0.029 ($\sigma = 0.021$), 0.028 for the derivative version of GazeTalk-J ($\sigma = 0.021$), and 0.053 for Dasher ($\sigma = 0.041$). There was no significant difference between the two versions of the latter typing system.

Table 8. ANOVA results on the over-production rate.

Factor	**s.s.**	**d.f.**	***V***	**F_0**
System (A)	0.014	1	0.014	0.61
Accuracy (B)	0.013	1	0.013	0.54
Block (C)	0.514	6	0.86	3.69*
A × B	0.027	1	0.027	1.17
A × C	0.177	6	0.030	1.27
B × C	0.215	6	0.036	1.54
A × B × C	0.233	6	0.039	1.67
Error	2.603	112	0.022	
Total	3.800	139		

* $p < 0.05$.

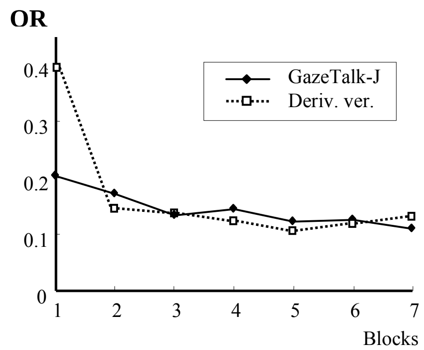


Figure 15. Transitions of the over-production rate with experimental blocks for the two versions of GazeTalk-J.

Table 9. ANOVA result for the frequency of using backspace.

Factor	s.s.	d.f.	*V*	F_0
System (A)	0.0276	2	0.01380	15.24*
Accuracy (B)	0.0034	1	0.00340	3.76
Block (C)	0.0045	6	0.00075	0.82
A × B	0.0026	2	0.00132	1.46
A × C	0.0112	12	0.00093	1.03
B × C	0.0049	6	0.00082	0.91
A × B × C	0.0057	12	0.00048	0.53
Error	0.1521	168	0.00091	
Total	0.2120	209		

* $p < 0.01$.

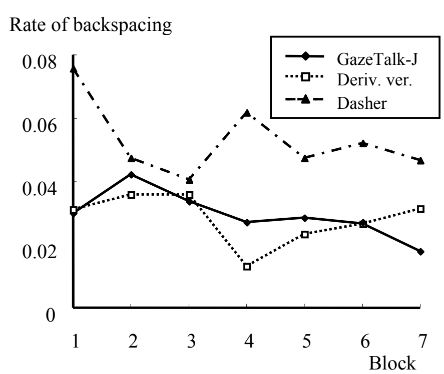


Figure 16. Transitions of the rate of backspacing with experimental blocks for the three typing systems.

Subjective ratings. The results of subjective ratings on two different occasions during system usage were described in this subsection, one at an early stage (after the first experimental block), and the other at the conclusion of the experiment (after seven blocks). As an overall trend, no remarkable difference between the typing systems was identified from the subjective responses for most items. Differences between the typing systems for between the time instances were 20% for most items, which could be made by only a single subject's response, as only five subjects were included in each system group. In addition, there might have been individual differences in the subjective criteria for rating each item. Therefore, we cannot derive a sound conclusion about subjective satisfaction with each typing interface from these results.

4.4 Discussion and summary

In this study, we conducted usability evaluations for two gaze-typing systems that operate in Japanese, representing different approaches to character selection, either by dwell-clicking (GazeTalk-J) or continuous navigation (Dasher). We made two versions of GazeTalk-J, one with the text field in the upper-left corner of the menu (GazeTalk-J) and one with the text field in the centre (derivative version of GazeTalk-J). GazeTalk-J's learning process was slightly more efficient than the other systems. The clear separation of the text field from the key area in the display and being able to see all keys via parafoveal vision may explain its superiority over the derivative version. The use of a well-established button metaphor and a well-known hierarchical structure may also explain its initial superiority over Dasher.

Typing speed seems to be sensitive to the small variations in design between GazeTalk-J and the derivative version, even though the differences could not be quantified via subjective rating. Some designers on our team expected the derivative version to be more efficient than GazeTalk-J, as the user would be able to verify typed characters in the text field using shorter saccades from the key area—or possibly by parafoveal vision. However, our results suggest that ease of verification is of less importance than is maintaining an overview of an ever-changing character layout. In future experiments, we would like to investigate the relative importance of saccade lengths and parafoveal vision, for instance by comparing typing speed and error measures using different sizes of the same keyboard.

A higher learning factor was obtained for some Dasher users than for users of GazeTalk-J and the derivative version of GazeTalk-J when estimated by the power law of practice. Dasher was expected to become more efficient for these users than for any users of the two versions of GazeTalk-J, after some amount of practice. Consequently, we would like to encourage GazeTalk-J users to try Dasher by providing direct access to Dasher from within future versions of GazeTalk-J.

Contrary to our expectations, there was no significant effect on eye-tracking accuracy for any of the performance measures examined in this study. This is a promising result for future development of low-resolution eye trackers. The method of incorporating deliberate miscalibration could be applied to every typing system to identify the threshold of tracking errors that cause poor performance. For Dasher and GazeTalk-J, the thresholds are more than 1 degree. Traditional *QWERTY* on-screen keyboards with a large number of small keys will presumably be more sensitive to inaccuracies in tracking.

In conclusion, the present study confirms that gaze-typing can be learned within a few hours practice, and can become almost error-free even when novel typing systems like Dasher are applied. We consider a productivity rate of more than 20 CPM to be acceptable; in fact, it is slightly higher than the typing speeds that have been reported previously for gaze-typing in English (Majaranta and Räihä, 2002).

5 References

Aoki, H., Itoh, K., and Hansen, J.P. (2005) Learning to type Japanese text by gaze interaction in six hours. *Proc. of the 11th International Conference on Human-Computer Interaction*, Las Vegas, NV, USA. July 2005 (CD-ROM).

Aoki, H., Itoh, K. and Hansen, J. P. (2006) Usability evaluation of gaze interface based on scan-path analysis. *Proc. of the 16th World Congress on Ergonomics*, Maastricht, Netherland, July 2006 (CD-ROM).

Bates R. (2002) Have patience with your eye mouse! Eye-gaze interaction with computers can work. *Proc. of the 1st Cambridge Workshop on Universal Access and Assistive Technology*. Available at http://www.cse.dmu.ac.uk/~rbates/research/cambseyemouse.htm

Choi, K-S. (2001) Kaist language resources. Ministry of Science & Technology Software Project results, 1995- 2000 (kibs.kaist.ac.kr).

Hansen, J.P., Hansen D.W., and Johansen, A.S. (2001) Bringing Gaze-based Interaction Back to Basics. *Proceedings of Universal Access in Human-Computer Interaction (UAHCI 2001)*, New Orleans, Louisiana, 325–328.

Itoh, K., Aoki, H., and Hansen, J. P. (2006) A comparative usability study of two Japanese gaze typing systems. In *Proc. of the Eye Tracking Research & Applications Symposium (ETRA 2006),* San Diego, CA, USA, ACM Press, New York, NY, 59–66.

Majaranta, P., and Räihä, K.-J. (2002) Twenty years of eye typing: Systems and design issues. *Proc. of the Eye Tracking Research & Applications Symposium (ETRA 2002)*, New Orleans, LA, USA, 15–22.

Nielsen, J. (1994) *Usability Engineering*. Morgan Kaufmann: San Francisco, CA, USA. (p. 362)

Soukoreff, R.W. and MacKenzie, I.S. (2003) Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric. *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, Fort Lauderdale, FL, USA, 113–120.

Ward, D.J. (2001) *Adaptive Computer Interfaces*. Ph.D. dissertation, University of Cambridge, U.K. Available online at http://www.inference.phy.cam.ac.uk/djw30/papers/thesis.pdf

Ward, D.J. and MacKay, D.J.C. (2002) Fast hands-free writing by gaze direction. *Nature* 418 (August 22): 838.