

Neurosci. 15, 463–485 (1997).
 5. He, D. Z. Z., Zheng, J. & Dallos, P. *Hearing Res.* **162**, 113–125 (2001).
 6. Zine, A. & Romand, R. *Brain Res.* **721**, 49–58 (1996).
 7. Kaltenbach, J. A., Falzarano, P. R. & Simpson, T. R. *J. Comp. Neurol.* **350**, 187–198 (1994).

8. Nguyen-Legros, J. & Hicks, D. *Int. Rev. Cytol.* **196**, 245–313 (2000).
 9. Miller, J. D., Watson, C. S. & Covell, W. P. *Acta Otolaryngol.* **176** (suppl.), 1–89 (1963).
 10. Dumont, R. A. *et al. J. Neurosci.* **21**, 5066–5078 (2001).
Competing financial interests: declared none.

Artificial intelligence

Fast hands-free writing by gaze direction

Here we describe a method for text entry based on inverse arithmetic coding that relies on gaze direction alone and which is faster and more accurate than using an on-screen keyboard. These benefits are derived from two innovations: the writing task is matched to the capabilities of the eye, and a language model is used to make predictable words and phrases easier to write.

For people who cannot use a standard keyboard or mouse, the direction of gaze is one of the few means by which they can convey information to a computer. Many systems for gaze-controlled text entry provide an on-screen keyboard with buttons that can be ‘pressed’ by staring at them. But eyes did not evolve to push buttons, and this method of writing is exhausting.

Moreover, on-screen keyboards are inefficient because typical text has considerable redundancy¹. Although a partial solution to this defect is to include word-completion buttons as alternative buttons alongside

the keyboard, a language model’s predictions can be better integrated into the writing process. By inverting an efficient method for text compression — arithmetic coding² — we have created an efficient method for text entry, which is also well matched to the eye’s natural talent for search and navigation.

One way to write a piece of text is to delve into a theoretical ‘library’ that contains all possible books, and find the book that contains exactly the desired piece of text³; writing thus becomes a navigational task. In our idealized library, the ‘books’ are arranged alphabetically on one enormous shelf. As soon as the user looks at a part of the shelf, the view zooms in continuously on the point of gaze. So, to write a message that begins “hello”, the user first steers towards the section of the shelf marked ‘h’, where all the books beginning with ‘h’ are found. Within this section are different sections for books beginning ‘ha’, ‘hb’, ‘hc’ and so on; the user enters the ‘he’ section, then the ‘hel’ section within it, and so forth.

To make the writing process efficient, we use a language model, which predicts the probability of each letter’s occurrence in a given context, to allocate the shelf space

for each letter of the alphabet (Fig. 1a). When the language model’s predictions are accurate, many successive characters can be selected by a single gesture.

We previously evaluated this system, which we call ‘Dasher’, with a mouse as the steering device⁴. Novices rapidly learned to write and an expert could write at 34 words per minute; all users made fewer errors than when they were using a standard ‘QWERTY’ keyboard.

Figure 1b shows an evaluation of Dasher driven by an eye-tracker, compared with an on-screen keyboard. After an hour of practice, Dasher users could write at up to 25 words per minute, whereas on-screen keyboard users could manage only 15 words per minute. Moreover, the error rate with the on-screen keyboard was about five times that obtained with Dasher.

Users of both systems reported that the on-screen keyboard was more stressful to use than Dasher for two reasons. First, they often felt uncertain whether an error had been made in the current word (the word-completion feature works only if no error has been made); an error can be spotted only by looking away from the keyboard. Second, a decision has to be made after ‘pressing’ each character on whether to use word completion or to continue typing — looking to the word-completion area is a gamble as it is not guaranteed that the required word will be there, and finding the correct completion requires a switch to a new mental activity. By contrast, Dasher users can see simultaneously the last few characters they have written and the most probable options for the next few. Furthermore, Dasher makes no distinction between word completion and ordinary writing.

Dasher works in most languages — the language model can be trained on sample documents and adapts to the user’s language as he or she writes. It can also be operated with other pointing devices, such as a touch screen or rollerball. Dasher is potentially an efficient, accurate and fun writing system not only for disabled computer users but also for users of mobile computers.

David J. Ward, David J. C. MacKay
 Cavendish Laboratory, Cambridge CB3 0HE, UK
 e-mail: mackay@mrao.cam.ac.uk

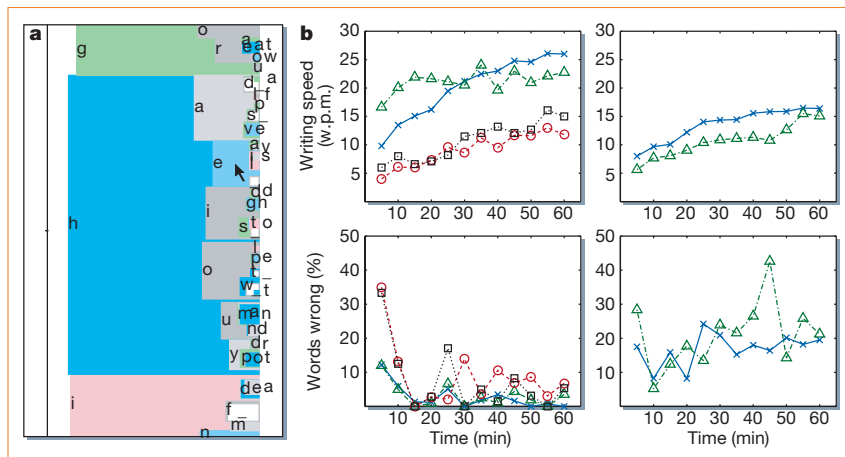


Figure 1 Hands-free text entry. **a**, Screenshot of ‘Dasher’⁵ when the user begins writing “hello”. The shelf of the alphabetical ‘library’ is displayed vertically. The space character (represented as an underscore) is included in the alphabet after ‘z’. In this example, the user has zoomed in on the portion of the shelf containing messages beginning with ‘g’, ‘h’ and ‘i’. Following the letter ‘h’, the language model makes the vowels and ‘y’ easier to write by giving them more space. Common words such as ‘had’ and ‘have’ are visible. The arrow indicates the gaze of the user; its vertical coordinate controls the zooming-in point and its horizontal coordinate controls the rate of zooming; looking to the left makes the view zoom out, allowing recent errors to be corrected. **b**, Comparison of writing speeds and error rates for two methods of gaze-driven text entry. Left, Dasher with eye-tracker, as recorded for two expert users of the system (crosses, triangles) and two novices (circles, squares); right, on-screen keyboard, used by two experts on the ‘QWERTY’ keyboard. The eye-tracking system was EyeTech’s Quick Glance eye-tracker. Each user took dictation from Jane Austen’s *Emma* in 5-min sessions. The language model PPM5 predicts the next character when given the previous five characters^{6,7}; it was trained on passages from *Emma* not included in the dictation. Right panels, the two experts took dictation using the same eye-tracker to control the WNVik on-screen keyboard (a standard ‘QWERTY’ keyboard) with the word-completion buttons enabled.

1. Shannon, C. E. *Bell Syst. Tech. J.* **27**, 379–423, 623–656 (1948).
 2. Witten, I. H., Neal, R. M. & Cleary, J. G. *Commun. Assoc. Comput. Machin.* **30**, 520–540 (1987).
 3. Borges, J. L. *The Library of Babel* (Godine, Boston, 1941).
 4. Ward, D. J., Blackwell, A. F. & MacKay, D. J. C. *Proc. 13th Annu. ACM Symp. User Interface Software Technol.* **2000** 129–137 (2000).
 5. Ward, D. J. Dasher, version 1.6.4 (2001); www.inference.phy.cam.ac.uk/dasher
 6. Cleary, J. G. & Witten, I. H. *IEEE Trans. Commun.* **32**, 396–402 (1984).
 7. Teahan, W. J. *Proc. NZ Comp. Sci. Res. Stud. Conf.* (1995); citesecr.nj.nec.com/teahan95probability.html
Competing financial interests: declared none.