Title
Writer recognition in cursive eye writing: A Bayesian model

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 36(36)

ISSN
1069-7977

Authors
Chanceaux, Myriam
Rynik, Vincent
Lorenceau, Jean
et al.

Publication Date
2014

Peer reviewed

Writer recognition in cursive eye writing: A Bayesian model

Myriam Chanceaux (myriam.chanceaux@upmf-grenoble.fr)
Univ. Grenoble Alpes, LPNC, F-38000, Grenoble, France
CNRS, LPNC, F-38000 Grenoble, France

Vincent Rynik (vincent.rynik@gmail.com)
Univ. Grenoble Alpes, LPNC, F-38000, Grenoble, France
CNRS, LPNC, F-38000 Grenoble, France

Jean Lorenceau (jean.lorenceau@upmc.fr)
Laboratoire des Systèmes Perceptifs, CNRS UMR 8248,
Département d'études cognitives, Ecole normale supérieure
29, rue d'Ulm, 75005, Paris, France

Julien Diard (julien.diard@upmf-grenoble.fr)
Univ. Grenoble Alpes, LPNC, F-38000, Grenoble, France
CNRS, LPNC, F-38000 Grenoble, France

Abstract

Using a novel apparatus coupling a visual illusion with an eye tracker device, trained participants are able to generate smooth pursuit eye movements, even without a target to follow. This allows them to perform arbitrary continuous shapes, and, for instance, write letters with their eyes. In a previous study, based on data from a single writer (author JL), we developed and tested a Bayesian computational model – the BAP-EOL model – able to simulate character recognition. In the present study, data from different writers provide the opportunity to study the signal characteristics of eye-written letters. More precisely, we extend the model to perform writer recognition. Experimental results, and high performance we obtained, show that eye writing is as writer specific as handwriting is, and that motor idiosyncrasies are present in eye-written letters.

Keywords: Bayesian modeling; writer recognition; eye writing

Introduction

A novel apparatus was recently designed that, in essence, allows users to write with their eyes (Lorenceau, 2012). It is commonly admitted that a target is needed to generate smooth eye movements, even if previous exceptions have been reported (Madelain & Krauzlis, 2003). Thanks to the use of a perceptual illusion, smooth pursuit control of the eyes even in the absence of a visual target to follow is possible. An eye tracking device records the user's eye movements, which can then be visualized on a screen. Using this system requires some training, as the visual illusion takes some time to be getting used to. Initially, users usually only perceive the illusion and generate smooth pursuit movements for short durations, so that the resulting trajectories are heavily contaminated by blinks and spurious saccades. With training however, some users become able to generate long, smooth trajectories in any desired shape.

An obvious application, and our long term objective, is to provide this system to motor impaired patients, for instance patients with amyotrophic lateral sclerosis (ALS, also known as Lou Gehrig's disease). Even if eye writing, in this manner, turns out to have a low communication throughput compared with virtual keyboard-based systems, the production of arbitrary trajectories would potentially help patients conserve artistic and self-expression capabilities longer. Eye writing, and more precisely the recording of eye produced trajectories, also provides an opportunistic window into the state and evolution of motor capabilities in patients and, by proxy, the state and evolution of their disease.

However, since eye writing is such a novel object of study, not much is known about the motor processes involved in preparing and performing letter traces with the eyes, and the signal characteristics of resulting trajectories. Although handwriting is widely studied (see Plamondon and Srihari (2000) for a review), it is not known, for instance, whether letters written with the eyes have stable shapes across repetitions, whether they have shapes similar to letters traced with other effectors (*i.e.*, whether motor equivalence carries over to the eyes as a writing effector), or whether they have shapes that allow writer recognition (*i.e.*, whether eye writing contains recognizable user idiosyncrasies, as handwriting does). This last issue is the main topic of the study we present here.

To answer this question, we developed and simulated a Bayesian model of writer recognition. It is an extension of the BAP-EOL model (for Bayesian Action-Perception for Eye On-Line), a Bayesian model that we used previously for character recognition in the context of eye writing (Diard, Rynik, & Lorenceau, 2013). The BAP-EOL model was itself an adaptation of the BAP model (Bayesian Action-Perception) of reading and writing handwritten letters (Gilet, Diard, & Bessière, 2011). Thanks to the flexibility of Bayesian inference, character recognition and writer recognition turn out to be similar tasks. Then, writer recognition was extended to take as input the letters of a complete word, instead of isolated letters, using a sensor fusion approach.

In the rest of this paper, we first recall the structure and main features of the BAP-EOL model, and introduce its extensions for writer recognition, based on single letters first, and on sequences of letters second. We then present an experiment with three different writers, its results and their analysis, in terms of performance and information accumulation.

BAP-EOL Model

Trajectory analysis

A preliminary step consists in extracting, from raw data, different variables to summarize the signal.

For each input trace, *i.e.*, each written character, recorded data is a series of x, y coordinates and, from these, velocities $\dot{x}, \dot{y}$ are computed using a finite difference approximation. Input traces are noisy, of course, and a smoothing filter (binomial filter of order 20) is applied to position and velocity dimensions. At the beginning of most traces, saccades are observed before the beginning of the letter proper, that is to say, before smooth pursuit eye movement begins. To remove these saccades, another filter is applied, based on an acceleration threshold, at the beginning and end of each trace: for the first 30 points (resp. last 10 points), if the observed acceleration exceeds a certain threshold, empirically fixed at 0.5 unit$/s^2$, all points before this peak (resp. after) are deleted. This efficiently removes most intrusive saccades in the signal.

We summarize the filtered traces by a sequence of via-points (Gilet et al., 2011). We choose, as via-points, the points of the trace where either x-velocity or y-velocity is zeroed (or both); the first and last point of the trace are also defined as via-points. For each via-point we memorize the displacements $\Delta x, \Delta y$ (relative to the preceding point) and velocities $\dot{x}, \dot{y}$: the k-th point ($k > 1$) is associated with positions $C^k_{\Delta x}$, $C^k_{\Delta y}$ and velocities $C^k_{\dot{x}}$ and $C^k_{\dot{y}}$ (the first via-point always has position $(0,0)$, and via-points from the last to the 25th have special values indicating trace termination). Relative positions were used, instead of absolute positions, so that characters could be written at any location on the display and via-point information recorded as the letter was being traced (absolute positions would require some size normalization process, which is only possible after the trace is completed). The system treats at most 25 via-points in a trajectory, which is more than enough for the current application (min 3, max 13, mean 5.9 in the learning database). Figure 1 shows an example of a trajectory before and after filtering, and the corresponding via-points.

After a trace is completed, other variables are added to the trajectory summary: the letter width S_x, its height S_y, and a variable that characterizes the density of the signal A, *i.e.*, the proportion of high-frequency components in eye movements. This last variable is based on the Fourier Transform of the signal in both x and y dimensions.

Probabilistic model of isolated letters

The BAP-EOL model is a probabilistic model of letters, where, in a nutshell, each letter l (in the set L of all considered letters, in our case, letters 'a' to 'z') is represented by a probability distribution over all dimensions introduced previously,

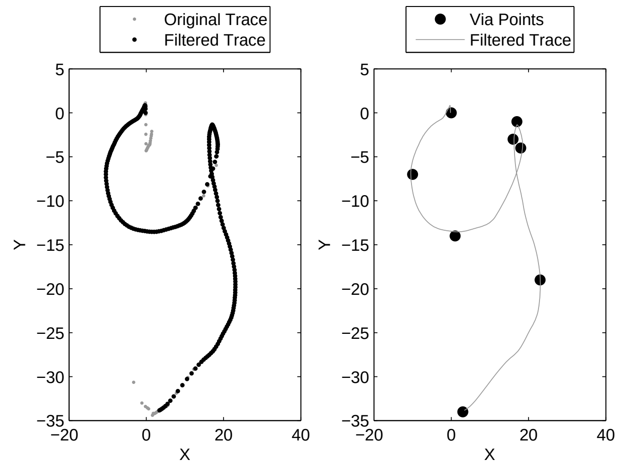

Figure 1: **Left:** input trajectory with initial and end segments filtered out because of intrusive saccades (gray portions). **Right:** via-points positions extracted from the filtered trace (velocity information is not shown).

i.e., $P(C^{1:25}_{\Delta x}\ C^{1:25}_{\Delta y}\ C^{1:25}_{\dot{x}}\ C^{1:25}_{\dot{y}}\ S_x\ S_y\ A\ L)$. For lack of space, we only present here its main features; technical details are provided in a previous paper (Diard et al., 2013). The BAP-EOL model was extended to introduce writer identity W, in a manner similar to the original BAP model of handwritten letter production and perception. We define W as a set of values $\{w_1, w_2, \ldots\}$, one for each possible writer.

The model is formally defined by its joint probability distribution $P(C^{1:25}_{\Delta x}\ C^{1:25}_{\Delta y}\ C^{1:25}_{\dot{x}}\ C^{1:25}_{\dot{y}}\ S_x\ S_y\ A\ L\ W)$. In order to obtain a computationally tractable model, discrete domains of suitable precision are chosen for each variable in the trace description: via-point relative positions $C^k_{\Delta x}$, $C^k_{\Delta y}$ have 81 possible values in the range $[-40, 40]$, via-point velocities $C^k_{\dot{x}}$ and $C^k_{\dot{y}}$ have 21 values in the range $[-10, 10]$, letter width S_x and height S_y have 51 possible values in the range $[0, 50]$ (not to scale with $C^k_{\Delta x}$, $C^k_{\Delta y}$ units), and, finally, the proportion of high-frequency components in the eye movements A has 31 possible values in the range $[0, 30]$.

Furthermore, conditional independence hypotheses are chosen so as to break down the dimensionality of the joint probability distribution. We define:

$$
\begin{aligned}
& P(C^{1:25}_{\Delta x}\ C^{1:25}_{\Delta y}\ C^{1:25}_{\dot{x}}\ C^{1:25}_{\dot{y}}\ S_x\ S_y\ A\ L\ W) = \\
& \quad P(L)P(W) \\
& \quad P(C^{1:25}_{\Delta x} \mid L\ W)P(C^{1:25}_{\Delta y} \mid L\ W) \\
& \quad P(C^{1:25}_{\dot{x}} \mid L\ W)P(C^{1:25}_{\dot{y}} \mid L\ W) \\
& \quad P(S_x \mid L\ W)P(S_y \mid L\ W)P(A \mid L\ W)\ ,
\end{aligned}
\tag{1}
$$

with $C^{1:25}_{\Delta x}$ a shorthand for the sequence $C^1_{\Delta x}, C^2_{\Delta x}, \ldots, C^{25}_{\Delta x}$.

In this decomposition of the joint probability distribution, the terms $P(L)$ and $P(W)$ are prior probability distributions over letters and writers, and are associated with uniform distributions, to represent ignorance of the frequency of letters and no preference for any writer. The next four terms encode the geometrical form of the trajectory, using a further decomposition (shown below for Δx positions, but it is similar on other dimensions):

$$P(C_{\Delta x}^{1:25} \mid L\, W) = P(C_{\Delta x}^{1} \mid L\, W) \prod_{i=2}^{25} P(C_{\Delta x}^{i} \mid C_{\Delta x}^{i-1}\, L\, W) \ . \qquad (2)$$

Each of the terms is associated with a conditional probability table, whose parameters are identified from a learning database.

Finally, the parameters of the last terms $P(S_x \mid L\ W)$, $P(S_y \mid L\, W)$ and $P(A \mid L\, W)$ are also learned from data, but these terms are associated to Gaussian probability distributions (with proper care taken to approximate these distributions over discrete, finite domains). Concerning $P(S_x \mid L\, W)$ and $P(S_y \mid L\, W)$ they are considered, for simplicity, independent of $P(C_{\Delta x}^{1:25} \mid L\, W)$ and $P(C_{\Delta y}^{1:25} \mid L\, W)$ conditionally to the learned data.

For simplicity, in the remainder of the paper, we note T the conjunction of all probabilistic variables involved in the description of a trace, *i.e.*, $C_{\Delta x}^{1:25}, C_{\Delta y}^{1:25}, \ldots, A$. With this notation, the structure of the probabilistic model simply becomes:

$$P(T\ L\, W) = P(L)P(W)P(T \mid L\, W) \ .$$

In other words, our model describes the most likely shapes and sizes of traces, for each letter and each writer, in a probabilistic manner.

Writer recognition from isolated letters

Once the parameters of all terms in the joint probability distribution definition are set, Bayesian inference is used to solve the task at hand. We are here interested in writer recognition, that is to say, given an input trace, identify the writer that produced it (but the letter is unknown). In probabilistic terms, this is solved by computing:

$$P(W \mid T) \propto \sum_{L} P(T \mid L\, W) \ . \qquad (3)$$

Probabilistic model of sequences of letters

We now extend the previous model so as to take into account sequences of written traces $T^{1:k} = T^1, T^2, \ldots, T^k$, as would be obtained from a written word. We directly consider a sequence of isolated letters and do not consider the segmentation problem, since, to this day, very few "eye writers" are expert enough to produce complete words in a single trace, without blinking.

The probabilistic model is extended and becomes a naive Bayesian fusion model where letters are assumed to be independent given the writer W, and the writer is assumed to be the same for all letters:

$$P(T^{1:k}\ L^{1:k}\ W) = P(W) \prod_{i=1}^{k} P(L^i) P(T^i \mid L^i\ W) \ .$$

$P(W)$ and all $P(L^i)$ are assumed to be uniform probability distributions, as previously. Each term $P(T^i \mid L^i\ W)$ is also structured and defined as for isolated letters.

Writer recognition from sequences of letters

In this task, input is a sequence $T^1, T^2, \ldots, T^k$ of written traces, and we compute the probability distribution over writers:

$$P(W \mid T^{1:k}) \propto \prod_{i=1}^{k} \left(\sum_{L^i} P(T^i \mid L^i\ W) \right) \ . \qquad (4)$$

Combining Eqs. (3) and (4), we obtain:

$$P(W \mid T^{1:k}) \propto \prod_{i=1}^{k} P(W \mid T^i) \ . \qquad (5)$$

In other words, recognizing the writer given a sequence of letters amounts to a sensor fusion of writer identification tasks for isolated letters, *i.e.*, probability distributions about writer identity given each letter are simply multiplied together.

Method

Participants

Three participants (one woman) produced a set of traces. One is author JL, and the two others were also involved in the project during data collection (JM and MV). All participants were French native speakers and reported having normal or corrected to normal vision. After a training phase consisting of practicing how to move their eyes using smooth pursuit with the illusion, they were able to produce data.

Procedure and Apparatus

In order to obtain voluntary smooth pursuit eye movements, the classic perceptual illusion of "reverse-phi motion" is presented to the participant (Anstis, 1970). The visual stimulus consists of a set of pairs of visual patterns presented in strobe and staggered in space with their polarity contrast reversed simultaneously. The perceived motion is the inverse of the shift of direction. During the illusion the whole display seems to move in the same direction as the eye. This illusion allows the oculomotor system to generate smooth pursuit eye movements without visual target (see Lorenceau (2012) for more details on this principle).

After a calibration phase, the illusion was displayed. During the presentation of the illusion, participants were moving their eyes, writing letters. After each letter, participants had to blink to indicate segmentation and start writing another letter. There was no feedback during the record.

A head-mounted camera EyeTechSensor equipped with CCD for ocular tracking (Pertech company) was used to record eye positions. The eye tracker had a sampling rate of 75 Hz. The illusion was presented on a monitor with a screen size of 1024*768 pixels. Stimulus presentation was controlled with the homemade Jeda software. Eye movements were recorded from the left eye.

Results

Learned parameters of the probability distributions

The participants produced a database of 933 characters (245 for author JL, 328 for JM, 360 for MV). The characters are letters from ‘a’ to ‘z’, with an average for each letter of 9.8 samples for JL (min 6, max 10), 13.12 for JM (min 10, max 17) and 14.16 for MV (min 9, max 19). Data collection was a result of participants’ practice sessions, without an explicit instruction of systematic alphabetic production.

We computed the parameters of the probability distributions using a cross-validation method: for each writer a set of 26 letters (one complete alphabet) was randomly selected as the test database (on which the recognition task performance was assessed), the remaining letters were the learning database. Thus the test database was of size 3*26, and the learning database of size 933-3*26. This random procedure was repeated 100 times to ensure that each letter was both in the learned and the tested database at least once.

All results presented below are the average measures over these 100 repetitions.

For each of these measurement, the parameters of the probability distributions of Eqs. (1) and (2) were learned: for instance, the terms about via-point relative positions and velocities are Conditional Probability Tables, implemented using Laplace succession laws (Gilet et al., 2011). They are a variant of histograms that start from a uniform distribution and converge, when data accumulates, to a histogram. To palliate the lack of experimental data compared to the number of free parameters to identify, a Gaussian filter was applied in order to smooth the obtained distributions (in effect, simulating a larger database with additional traces similar in shapes to the ones available). The Gaussian filters parameters are of order 15 and variance 2 for relative positions and of order 7 and variance 1 for velocities.

With these parameters the BAP-EOL model becomes operational, and can be used to perform automatic writer recognition.

Writer recognition: experimental results

From isolated letters We performed writer recognition a hundred times for each letter and each writer. Averaging results over the 100 repetitions, we obtained three confusion matrices of size 26*3. They are shown Figure 2 (upper panel). On these matrices we observe that some letters and some writers are more easily recognized. For example, input from writer MV is efficiently recognized, whatever the written letter. On the other hand, writer JL is harder to recognize, with a low recognition rate for letters like ‘j’ or ‘t’. This could be due to the fact that the learning database is smaller for JL than for MV (245 vs. 360 samples). Another explanation relies on letter similarity and distinguishability: for instance, JL usually writes the letter ‘s’ in a very particular way, and they are easy to recognize, whereas JL’s ‘t’s or ‘j’s are not characteristic.

Further averaging over letters, we obtained a 3*3 confusion matrix (Figure 2, bottom), whose diagonal values represent correct recognition rates of writers, independently of the written letter: these global recognition rates are 70.00 % for writer JL, 73.82 % for writer JM and 90.54 % for writer MV.

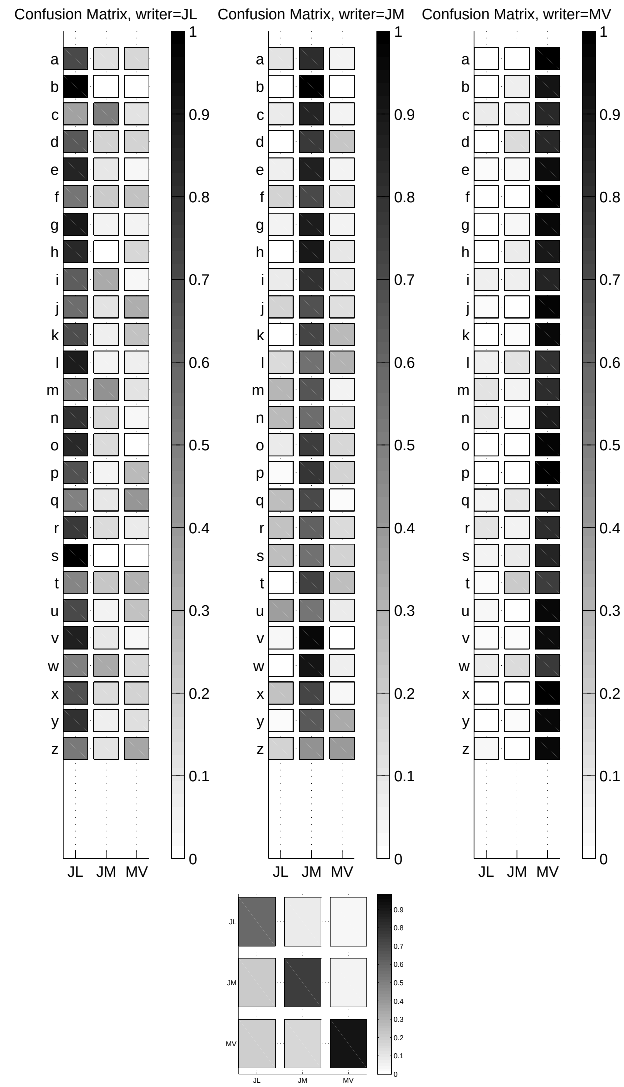

Figure 2: **Upper panel**:Confusion matrices for writer recognition, for each letter and each writer. Each row is the probability distribution over writers, computed from Eq. (3), averaged over 100 experimental repetitions. **Bottom panel**: Global confusion matrix.

From sequences of letters To test this model over sequences of letters, a dataset of words from a French corpus *Lexique* (New, Pallier, Brysbaert, & Ferrand, 2004) was used. 2126 words were selected, all were singular nouns (length 3 to 12 letters), with a mean printed frequency greater than 15 occurrences per million.

We created input trajectories for words by randomly extracting from the database, for a given writer, trajectories for the letters of that word, and repeating that procedure 100 times. We obtained a test database of 2126*100*3 words. As above, for each letter of each word, a probability distribution over writers is computed, and these probability distributions are then gradually multiplied to obtain writer recognition from the first two letters, from the first three, etc, until

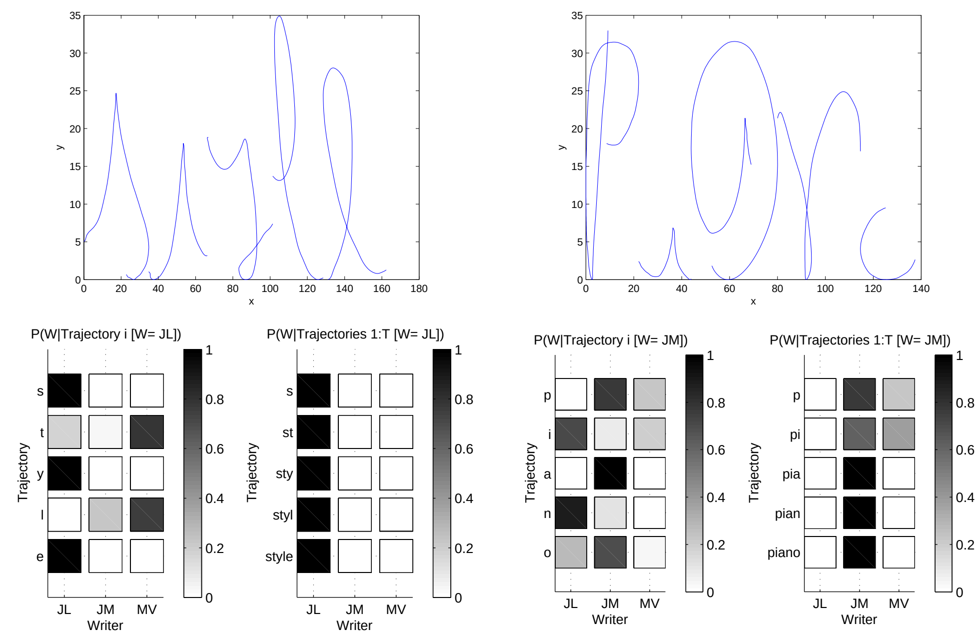

Figure 3: Examples of writer recognition based on words: input word "*style*" by writer JL on the left, input word "*piano*" by writer JM on the right. **Upper panels**: input trajectory for the words, obtained by sequencing samples of letters that constitute the word (letters not scaled; notice that participants did not dot the 'i' nor cross the 't'). **Bottom panels**: confusion matrices obtained by writer recognition, from each isolated letter (left sub-panels), and aggregated for growing prefixes, as letters are fed to writer recognition and Eq. (5) is applied (right sub-panels).

writer recognition from the complete word is obtained. Two examples are displayed Figure 3, for the input word "*style*" by writer JL, and the word "*piano*" by writer JM.

On these examples, one can see that even if individual letters do not always yield correct writer recognition, the fusion model provides a more reliable estimate of writer identity. Of course this is not always the case, with counterexample words sometimes incorrectly recognized. This mostly happens for words that contain several difficult letters. For instance, "*accent*" by JL is sometimes recognized as being written by JM, but still 82.09 % of the times correctly recognized as JL's.

Overall results are satisfactory. Averaged on all words and all repetitions, correct recognition rates after the last letter are 95.43 % for JL, 98.29 % for JM and 99.92 % for MV. These results are in line with writer recognition rate in handwritten documents (Bensefia, Paquet, & Heutte, 2005). We notice that the accuracy is better in term of recognition rate with entire words than with only one letter.

Therefore, we analyzed the evolution of writer recognition as letters are fed, one by one, to the system. We computed the average correct recognition rates, as a function of the number of letters. This is shown Figure 4 for 7-letter words. It of course starts at chance level (33 %) before the first letter is seen, as the prior probability distribution over writers, $P(W)$, is uniform. For all writers, correct recognition rate then quickly increases, with 2-3 letters being sufficient to obtain near final performance.

We also computed the average entropy of probability distribution over writers, as a function of the number of letters. This is shown Figure 5, also for 7-letter words. Recall that entropy of a discrete probability distribution $P(x)$ is defined as $-\sum_x P(x) \ln P(x)$. In our case, entropy initially starts from $\ln 3 \approx 1.09$ nats, as $P(W)$ is uniform, and decreases sharply as information is gathered and the probability distribution over writers concentrates. Slight differences can be observed between writers but, overall, 2-3 letters also are sufficient to be near final entropy, showing a fast convergence speed of the model.

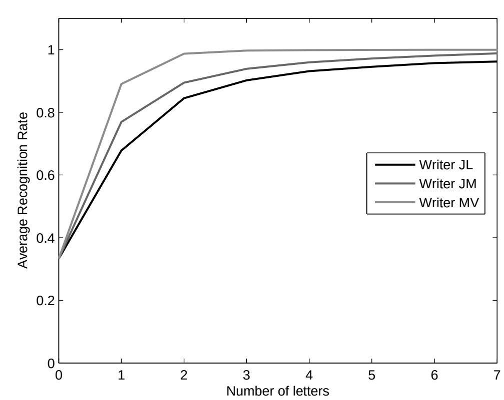

Figure 4: Evolution of the correct recognition rate of writers, as a function of the number of input letters, averaged for all 7-letter words of the test database.

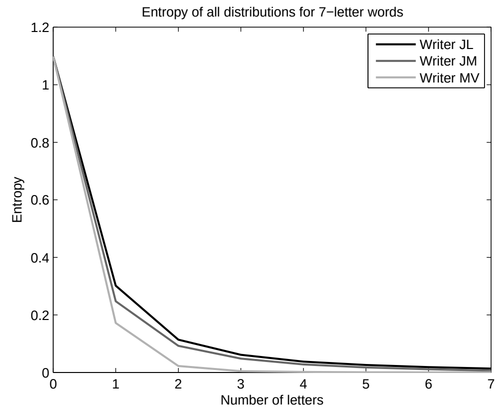

Figure 5: Evolution of the entropy of the probability distribution over writers, as a function of the number of input letters, averaged for all 7-letter words of the test database.

Discussion

In this paper, we refined the BAP-EOL model to simulate letter perception and writer recognition, in the context of writing with the eyes. We showed how Bayesian modeling and inference could be used to solve the writer recognition task, both based on single isolated letters, and on sequences of letters. We tested our model on a database containing 3 different writers, and a corpus of several thousands of words. Experimental results show that the model is quite efficient in the task, especially using the fusion approach with entire words.

In our experiments, we noted that some letters made writer identification more easy, whereas some made it more difficult, because of similarity in writing styles of some letters between writers, as in handwriting. The Bayesian model could be extended, however, to include this information. Instead of a uniform probability distribution over letters, which, in effect, gives the same weight to all letters, a prior distribution reflecting the distinguishability of letters could be used. This could be done by adapting a Bayesian meta-model of the distinguishability of models, from previous research in another domain (Diard, 2009). In a nutshell, given a database, the model would test itself and give more weight to letters that yield good writer distinguishability, and less weight to letters that are similar across writers.

We have presented, in the context of writer recognition, a first model using sequences of letters. However, in a classical naive Bayesian fusion approach, we have assumed letters to be independent, given writer identity W. This could of course be refined, by introducing knowledge about letter frequency in a given language, bigram frequencies, or even word frequency, or other high-level orthographic, lexical and semantic information. In a Bayesian framework, introducing such knowledge takes the form of top-down prior probability distributions, which can be hierarchically combined. The domain of probabilistic visual word recognition is indeed currently growing along these lines (Norris, 2006, 2013).

Finally, we have shown that the BAP-EOL model is able to recognize writers. Recall that our long term objective is disability assessment. Imagine a user that begins to show signs of motor deterioration, like micrography for instance. In this example, this would affect the letter sizes, which would become uncharacteristically small. Our system would then not recognize the user as the writer anymore, which could raise an alarm to the patient's caregivers. Of course, carefully calibrating our model so that it is robust to usual variations in eye writing but able to detect such motor deterioration, even if there are more writers in the database, would require much more data about the use of our system by disabled patients, than is currently available. However, we believe the writer recognition mechanism we have described here is a promising first step in this direction.

Acknowledgments

This work was supported by the Eye-On-Line ANR Grant (ANR-12-TECS-0009-04). We thank MV and JM, who produced and collected trajectories used in this research.

References

Anstis, S. M. (1970). Phi movement as a subtraction process. *Vision research*, *10*(12), 1411–30.

Bensefia, A., Paquet, T., & Heutte, L. (2005). Handwritten document analysis for automatic writer recognition. *Electronic Letters on Computer Vision and Image Analysis*, *5*(2), 72–86.

Diard, J. (2009). Bayesian model comparison and distinguishability. In *Proceedings of the international conference on cognitive modeling (ICCM 09)* (pp. 204–209).

Diard, J., Rynik, V., & Lorenceau, J. (2013). A Bayesian computational model for online character recognition and disability assessment during cursive eye writing. *Frontiers in psychology*, *4*(843).

Gilet, E., Diard, J., & Bessière, P. (2011). Bayesian action-perception computational model: interaction of production and recognition of cursive letters. *PLoS ONE*, *6*(6), e20387.

Lorenceau, J. (2012). Cursive Writing with Smooth Pursuit Eye Movements. *Current Biology*, *22*(16), 1506–1509.

Madelain, L., & Krauzlis, R. (2003). Effects of learning on smooth pursuit during transient disappearance of a visual target. *Journal of Neurophysiology*, *90*, 972–982.

New, B., Pallier, C., Brysbaert, M., & Ferrand, L. (2004). Lexique 2: a new French lexical database. *Behavior research methods, instruments, & computers: a journal of the Psychonomic Society, Inc*, *36*(3), 516–24.

Norris, D. (2006). The Bayesian Reader: Explaining word recognition as an optimal Bayesian decision process. *Psychological Review*, *113*(2), 327–357.

Norris, D. (2013). Models of visual word recognition. *Trends in Cognitive Sciences*, *17*(10), 517–524.

Plamondon, R., & Srihari, S. N. (2000). On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(1), 63–84.