Writer identification by means of explainable features: shapes of loop and lead-in strokes

Vivian Blankers Ralph Niels Louis Vuurpijl

NICI, Radboud University Nijmegen

Abstract

Writer identification is an important issue in forensic investigations of handwritten documents. A particularly well-established method employed by forensic experts is to (visually) explore distinguishing features of handwritten characters for comparing pieces of handwriting. Our research within the NWO Trigraph project aims at automizing this laborious process. In this paper, we propose a novel method for identifying a writer by means of features of loops and lead-in strokes of handwritten characters. Using a k-nearest-neighbor classifier, we were able to yield a correct identification performance of 98% on a database of 41 writers. These results are promising and have great potential for use in the forensic practice, where the outcomes of handwritten document comparisons have to be justified via explainable features like the ones explored in this paper.

1 Introduction

Handwriting is one of the traces by which an individual can be identified. This is important in forensic investigations of so-called questioned handwriting, because identifying the writer could assist in solving a crime [8]. For this task, human forensic document examiners compare questioned handwriting to a collection of reference documents, by exploring distinguishing characteristic features present in the handwriting. In cases where large databases need to be examined, however, human comparison becomes practically impossible and is prone to subjective decisions [1]. In the past years, various attempts have been made to automize the process of forensic document examination [1, 2, 3, 7, 8], with the goal to develop the required knowledge and technology to process large amounts of documents in an objective manner, while achieving high writer identification rates. Large studies have quantitatively shown that handwriting is individual and that a combination of such distinguishing features can indeed be used to identify the writer [8, 10]. To this end, global features like slant, distances between lines or words, or statistical information on ink distributions and local features on the allographic (character shape) level (like allograph dimensions, shape, or the presence of certain loops or trajectory crossings) can be used [8].

This paper focuses on the question to what extent certain local, sub-allographic features can be used for writer identification. Sub-allographic features are distinguishing characteristics computed from parts of a letter. Such features are particularly important for forensic writer identification, since human experts use distinctive parts of characters as significant clues for providing evidence: Based on such features, an explainable proof can be given on writer resemblances [7].

Two groups of structural features will be explored below: loops and lead-in strokes. Loop features are found in the loops of *ascenders*, as they appear in the letters ‘l’, ‘k’ and ‘b’ and *descenders*, as they appear in the letters ‘g’ and ‘j’ (see Figure 1). There has been some research on loop features for writer identification [4], but that primarily focused on the different loop sizes for one particular writer, or was eventually classified as irrelevant [7]. However, human forensic experts [5] and the results presented in this paper, show that loop features can be very distinctive for writer identification. Lead-in features represent the first part of a character and are found in almost all letters of the alphabet, especially in cursive handwriting [5]. To the best of our knowledge, there has not been any similar kind of research on lead-in strokes before.

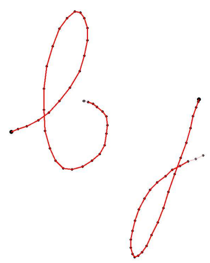

Figure 1: Ascender ('b') and descender ('j') loops.

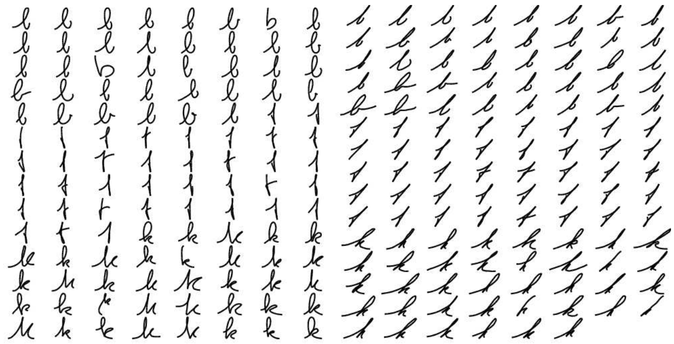

Figure 2: Loops from the same writer are distinctive from other writers and may be similar for different letters written by the same writer.

Our studies show that there is a similarity in the different loops and lead-in strokes produced by a single writer. This means that the loops occurring in the 'l's writers write, will be quite the same every time they write this letter 'l', and may even bear resemblance to the loops they produce in other letters containing ascenders, like the 'k', 'b' (see Figure 2) or even letters containing descenders, e.g. 'j' or 'g'. Similarly, the lead-in stroke of an 'a' could correspond with the lead-in strokes of a 'c' or a 'd', for example. Intrigued by these observations, we have explored loop and lead-in features with the following two main questions in mind:

1. Can a writer be identified by comparing one of the letters he or she wrote to a database of *equal* letters of which the identity of the writer is known, using only features of the loops and lead-in strokes? For example: If a 'b' of an unknown writer is available, can the writer be found by comparing that 'b' to the 'b's in a labeled database?

2. Can a writer be identified by comparing one of the letters he or she wrote to a database of *similar* letters of which the identity of the writer is known, using only features of the loops and lead-in strokes? For example: If a 'b' of an unknown writer is available, can the writer be found by comparing that 'b' to the 'h's, or the 'k's in a labeled database?

This research is part of the NWO ToKeN project *Trigraph*, which pursues the development of novel writer identification techniques. Within Trigraph, very good results on using global directional features derived from scanned handwritten documents have been achieved [2]. Our investigations on a local scale are performed in close collaboration with experts from the Dutch Forensic Institute and are targeted at structural allographic and sub-allographic features that are common in forensic practice. The explorations of loops and lead-in strokes treated in this paper, are believed to provide a promising step towards explainable features that eventually can be used in court as forensic evidence. Below, we will first introduce our method in Section 2, containing a description of the datasets and derived structural features. Results are presented in Section 3 and this paper concludes with a discussion and pointers to future research in Section 4.

2 Method

2.1 Data collection

A selection of 41 writers (volunteers and paid volunteers) from the *Plucoll* database [9] of handwritten letters was used to develop and test our technique. For this study, a database containing pre-segmented characters were used. The data in the *Plucoll* set was recorded using a WACOM PL100-V tablet, with an electromagnetic wireless pen, a sample rate of 100 Hz, a resolution of 0.02 mm/unit, and an accuracy of 0.1 mm. This data is online data, which means that time and pressure data are available. Data is recorded not only when the pen is on, but also when it hovers (0-10mm) above the tablet. If the pen is on the tablet, the produced trajectory is called *pendown*, while data recorded with the pen above the tablet is called *penup*. Note that in scanned images of ink, also called *offline* data, *penup* trajectories cannot be distinguished. We are aware of the fact that for many forensic investigations, online data will not be available. However, for assessing our research questions, it is valid to use dynamic trajectories. Furthermore, we have recently shown that in many cases dynamic trajectories can be restored from static images [6].

2.2 Loop features

A loop is defined as a closed curve, where the ends cross each other at some intersection point (see Figure 3). Typically, loops of ascenders have an counter-clockwise orientation and loops of descenders follow a clockwise direction [5]. To enable within-group comparison of these distinctive groups, characters were distinguished in a group containing ascending loops ('b', 'd', 'f', 'h', 'k', and 'l') and a group containing descending loops ('d', 'f', 'g', 'j', 'p', 'q' and 'y').

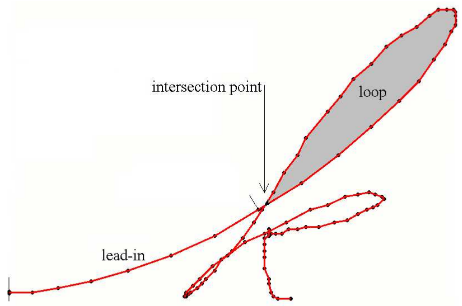

Figure 3: The ascending loop and the lead-in stroke of a 'k'.

To compare writers based on the loops and lead-in strokes they produce, a program which calculates various features was developed. The most important features are described below. Only the letters that contain lead-in strokes and/or loops were considered:

1. **Length** The length of the total trajectory of the letter is computed as the sum of Euclidian distances between each pair of succeeding coordinates. The relative length of the loop, with respect to the length of the total letter, is also calculated.

2. **Area** A loop is a polygon with size:

$$Area = \frac{1}{2} * (x1y2 - x2y1 + ... + xny1 - x1yn) \tag{1}$$

3. **Width/height ratio** The width of the loop is calculated by finding the difference between the minimum and the maximum x-value of a loop, and the height is calculated by finding the difference between the maximum and the minimum of the y-values of the loop (see Figure 4).

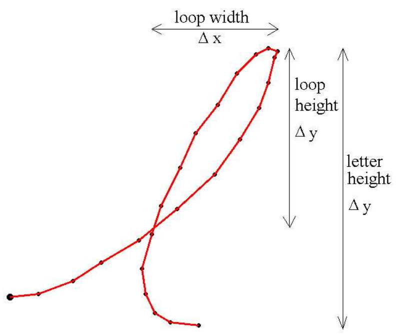

Figure 4: Width and height of a loop and letter.

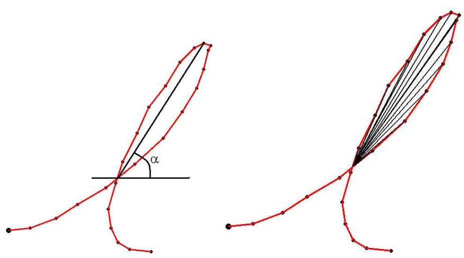

Figure 5: Direction and average direction of a loop.

4. **Relative height** The relative height is the ratio between the height of the loop and the total height of the letter (see Figure 4).

5. **Direction** The direction of a loop is the angle between the x-axis and the vector between the intersection point and the highest point of the loop (in case of an ascender) or the lowest point of the loop (in case of a descender). See Figure 5. The loop of an ascender has a direction between 0 and 180 degrees, while the loop of a descender usually has a direction between 180 and 360 degrees. The direction is calculated using Eq. 2.

$$Direction = \arctan \frac{\Delta\, y\ of\ the\ loop}{\Delta\, x\ of\ the\ loop} \tag{2}$$

6. **Average direction and standard deviation** The average direction of a loop is the average angle between the x-axis and each of the coordinates in the loop (see Figure 5) using Eq. 2. This is calculated by adding up those angles, and dividing the result by the number of angles. The standard deviation is computed to quantify how much the loop directions differ from the mean. With this information the broad and narrow loops can be distinguished (see Figure 6).

7. **Curvature** The curvature of a trajectory is defined by the average angle between each couple of succeeding vectors (see Figure 7). Given two vectors $\vec{a}$ and $\vec{b}$, the angle between them is calculated using Eq. 3.

$$Angle = \arccos \left(\frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}||\mathbf{b}|} \right) \tag{3}$$

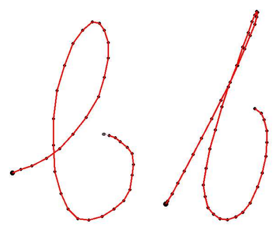

Figure 6: A broad and a narrow loop.

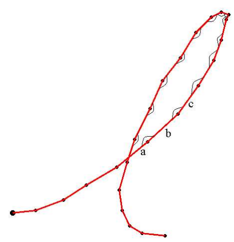

Figure 7: The curvature of a loop.

2.3 Lead-in features

Because the shape of lead-in strokes differs between letters [5], we have divided the lead-in letters into four groups that each contain letters with similar shaped lead-in strokes. In each group, the lead-ins start at the first coordinate of the sample. The difference is in the position where the lead-in strokes end.

For most ascenders, we define the lead-in stroke as the part of the letter before the loop (i.e., the end of the lead-in stroke is the coordinate where the loop begins). This group contains the letters ‘b’, ‘h’, ‘k’, ‘l’, ‘t’. Note that if the letters lack a loop, we defined the lead-in as the part of the letter before the maximum y-coordinate (i.e., the top of the letter). An example can be found in Figure 8.

The second group contains the letters ‘e’ and ‘f’. The lead-in stroke of these letters is defined as the part of the letter before the first intersection (see Section 2.2). If no intersection exists, we have not defined a lead-in stroke.

The third group consists of the letters ‘i’, ‘j’, ‘m’, ‘n’, ‘p’, ‘s’, ‘u’, ‘v’, ‘w’, ‘y’ and ‘z’. In this group, lead-in strokes end where the trajectory starts going down. This means that the lead-in strokes are all directed upwards.

The last group contains the letters ‘a’, ‘c’ ‘d’ and ‘q’. In this group, lead-in strokes usually are directed towards the right. They end at the position where a sharp turn left is made.

It should be noted that not all lead-ins can be captured by these definitions. Particular letter shapes, like the ‘a’ written in typewriter fashion (‘a’ instead of ‘*a*’) and a ‘c’ that resembles the letter ‘e’, are so different from most group members, that their lead-in strokes cannot be found given the rules stated above. These problem cases, were excluded from our database.

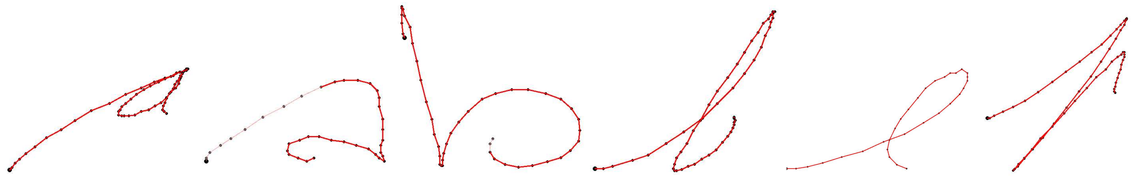

Figure 8: Different types of lead-ins.

1. **Length** After calculating the absolute length of the lead-in stroke, we calculate the relative length for the lead-in stroke with respect to the total letter.

2. **Direction** The direction of a lead-in stroke is the angle between the x-axis and the vector between the first and last coordinate of the stroke. Because most lead-in strokes are directed towards the upper right, the angle is usually between 0 and 90 degrees. The direction is calculated as in Eq. 2.

3. **Average direction and standard deviation** The average direction of a lead-in stroke is calculated by summing the angles between the succeeding vectors in the stroke, and dividing the result by the number of vectors (see Figure 9). The standard deviation indicates how much the directions differ from the mean.

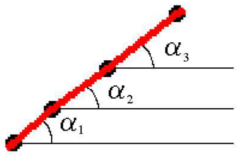

Figure 9: The average direction of a lead-in stroke.

4. **Curvature** The curvature is the average angle between two vectors of a lead-in stroke, calculated as in Eq. 3.

2.4 Analysis

The total dataset (see Section 2.1) was randomly divided over two subsets of equal size, a *trainset* and a *testset*. The *trainset* was used to optimize the parameters of the k-nearest-neighbor classifier that was used to perform writer identification tests on the *testset*.

2.4.1 Training

The k-nearest neighbor algorithm is a method for classifying objects based on closest training examples in the feature space. The training examples are mapped on a multidimensional feature space, by means of Euclidean distances. The training phase of the algorithm consists of storing the feature vectors and writers of the training samples. In the actual classification phase, the same features as before are computed for the test letter (whose writer is unknown). Distances from the test vector to all stored vectors are computed and the k closest samples are selected. The identified writer is predicted as the most numerous writer within these k known samples (unweighted majority voting). We also implemented a weighted majority voting algorithm, where the nearest vector gets a higher score than the second nearest, et cetera. To optimize the quality of the to-be-obtained results, we used the *trainset* to decide the best value of k (we tried $k = 0, 1, .., 20, 25, 30, 35, 40, 45, 50$), and the best decision algorithm (weighted or unweighted majority voting).

To this end, we tested all letters from the *trainset* in a leave-one-out manner, and counted how often the classifier returned the correct writer. Weighted majority voting turned out to generate the best results, and the optimal value of k was different for each letter.

2.4.2 Testing

For the tests, we created 9 different groups from of the *testset*, as depicted in Table 1.

group	features	letters
combination	loop and lead-in features	b, d, f, h, j, k, l, p, q, y
ascenders	loop features	b, d, f, h, k, l
descenders	loop features	g, j, p, q, y
loops	loop features	ascenders and descenders
a-leadins	lead-in features	a, c, d, q
b-leadins	lead-in features	b, h, k, l, t
e-leadins	lead-in features	e, f
i-leadins	lead-in features	i, j, m, n, p, s, u, v, w, y, z
all leadins	lead-in features	all letters, except g, o, r, x

Table 1: The different test groups.

Two different tests were carried out. The first explored how well a writer could be identified when comparing a letter only to *equal* letters (e.g., comparing questioned 'b's only to 'b's in the database). The second explored how well the system performed when comparing the letters to *similar* letters, or letters belonging to the same group (see Table 1).

Comparing to equal letters

For every letter, the feature-vector is compared to all feature-vectors of that kind, e.g. a 'b'-loop is compared to all other 'b'-loops, using the k-nearest-neighbor classifier and the optimal k found in the previous step. We performed this test in a leave-one-out manner for all letters. To test whether the identification performance would increase given more available data, we have performed tests on different amounts of available letters. Random selections of letters produced by the same writer were offered to the knn classifier, and for each letter, the k nearest samples were listed. The lists for each letter were combined, and the writer with the highest weight was returned by the system. Sets of $1, 2, .., 9, 10$ and $15, 20, 25, 30$ random letters were used.

Comparing to similar letters

To find out to which extend similar letters can be used as a query for writer identification, we performed another test. This test does not compare feature-vectors with feature-vectors of the same letter, but with feature-vectors of similar letters: letters from the same group (see Table 1). For example: the lead-in feature-vector of an 'a' is compared to the lead-in feature-vectors of all letters 'c' (both contained in the *a-leadins* group). Similarly, the loop feature-vector of a 'b' is compared to the loop feature-vectors of all 'h's (the *ascenders* group), and also to all loop feature-vectors of all the 'p's in that group (the *loops* group).

In this test, we used $k = 10$ and weighted majority voting for all combinations. This test was done for all groups except for the *all-leadins* group, since the feature values between different letters differed so much that no useful results could be expected.

3 Results

To test how well our system is able to find the correct writer given an unknown handwritten letter, we counted how often the classifier was able to find the correct writer.

3.1 Equal letters

With 30 available letters per writer, our program correctly identified 85.85% of the lead-in letters. Loop letters yield a higher performance: with 30 available ascender-loop letters, the program correctly classified 96.05% of the writers, whereas descender-loop letters gives a little less performance: 94.76%. With 30 available letters, mixed ascender and descender loop letters, the performance is 98.05%. Given 30 letters that contain both loops and lead-in strokes, a score of 95.05% is obtained. Note that chance level for 41 writers is 2.44%.

The results for different amounts of available letters are summarized in Figure 10.

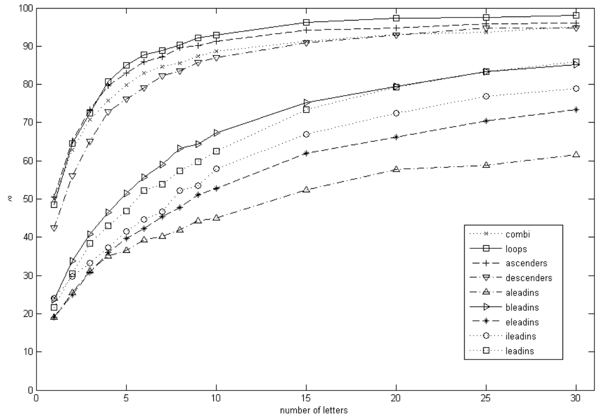

Figure 10: Results of the *equal letters* test.

3.2 Similar letters

In the second test, we evaluated how well our system is able to identify the writer of a letter using a database of only *similar* letters (i.e., letters that belong to the same group, as described in Table 1). We have performed evaluations for each combination of letters in all groups (i.e., we have used 'g's as query in a database of 'j's, a database of 'p's, a database of 'q's and a database of 'y's, et cetera). The average within-group performances can be found in Table 3.2 and Figure 11.

group	performance
combination	13.64
ascenders	31.82
descenders	18.04
loops	13.43
aleadins	10.60
bleadins	14.73
eleadins	6.50
ileadins	7.21

Table 2: Average within-group performance of the *similar letters* test.

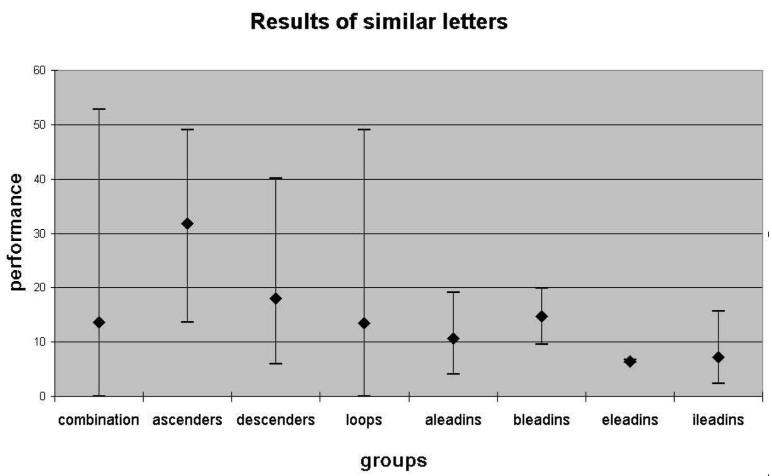

Figure 11: Summarized results of the *similar letters* test.

4 Discussion

In this paper, we have presented a novel method for writer identification based on sub-allographic structural properties of handwriting. We have tested our features in two different settings (*equal* letters and *similar* letters) using a knn classifier. The obtained results show that our system is very well able to identify a writer given a database of equal letters (e.g., a database of 'b's when the query is a 'b'). Correct identification percentages of over 95 percent are achieved if 30 letters of a person are available, but even with less data, very useful results are obtained (over 90% if less than 10 letters are available). A prominent conclusion that can be made is that — as opposed to earlier findings from the literature — the contribution of loop features is paramount for success. As can be observed in Fig. 10, loop features have a far better performance than lead-in features.

Preliminary results are shown for the second test, which explored whether queries to a database of similar, but not equal, letters (e.g., a database of 'h's when the query is a 'b') can be used for writer identification. Please note that for this test, only one character was used, which explains the low identification rates which were achieved. However, comparing these results to the first point depicted in Figure 10 (mimicking the situation that only one query character is available), provides hope for further research. In the current setting, identifying a writer boils down to the situation in which only one character is retrieved from a crime scene, which is used to match to a reference database of distinct characters. If we take this into account, the results come quite close to the other test, which explored *equal* characters. In collaboration with forensic experts, we are now investigating which combinations of query characters would result in improved writer identification results. Furthermore, more studies will be performed, also optimizing on different parameter settings for the knn classifier. The results of these explorations will definitely become available before the BNAIC conference.

References

[1] A. Bensefia, T. Paquet, and L. Heutte. A writer identification and verification system. *Pattern Recognition Letters*, 26(13):2080–2092, October 2005.

[2] M. Bulacu and L. Schomaker. Text-independent writer identification and verification using textural and allographic features. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(4):701–717, April 2007.

[3] K. Franke and M. Köppen. A computer-based system to support forensic studies on handwritten documents. *International Journal on Document Analysis and Recognition*, 3(4):218–231, 2001.

[4] R. Marquis, F. Taroni, S. Bozza, and M. Schmittbuhl. Size influence on shape of handwritten characters loops. *Forensic Science International*, 2007. In Press.

[5] R.N. Morris. *Forensic handwriting identification : fundamentals, concepts and principals*. Academic Press, San Diego, California, 2000.

[6] R. Niels, L. Vuurpijl, and L. Schomaker. Automatic allograph matching in forensic writer identification. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(1):61–81, February 2007.

[7] V. Pervouchine and G. Leedham. Extraction and analysis of forensic document examiner features used for writer identification. *Pattern Recognition*, 3:1004–1013, March 2007.

[8] Srihari, S.-H. S. N., Cha, H. Arora, and S. Lee. Individuality of handwriting. *Journal of Forensic Sciences*, 47(4):1–17, July 2002.

[9] L. Vuurpijl and L. Schomaker. Finding structure in diversity: a hierarchical clustering method for the categorization of allographs in handwriting. In *ICDAR '97: Proc. 4th International Conference on Document Analysis and Recognition*, pages 387–393, Washington, DC, USA, 1997. IEEE Computer Society.

[10] L. Vuurpijl, L. Schomaker, and M. van Erp. Architectures for detecting and solving conflicts: two-stage classification and support vector classifiers. *International Journal of Document Analysis and Recognition*, 5(4):213–223, July 2003.