Handwriter: An R Package For Statistical Writership Analysis

Presenters: Félix Báez-Santiago, Julia Lundstrum

Amy Crawford, Nick Berry, Ben Escobar, James Taylor, Stephanie Reinders, Danica Ommen, Alicia Carriquiry

Iowa State University

Overview

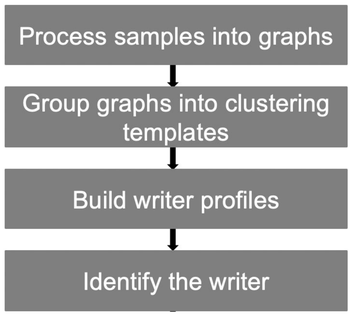
EVIDENCE: An investigator has two pieces of evidence:

1. A handwriting document from an unknown writer, called a *questioned document*.
2. Handwriting samples from a set of *potential writers* (suspects).

GOAL: Estimate the probability of each potential writer being the questioned document's writer.

METHOD [1, 2]:

1. *Pre-process* handwriting samples and decompose handwriting into disjoint graphs that roughly, but not exactly, represent letters.
2. Group similar graphs into clusters to form a *clustering template*.
3. Build a *writer profile* for each potential writer by estimating the rate at which they produce graphs that belong to each cluster in the clustering template.
4. *Identify the writer* by comparing the questioned document with each of the writer profiles to estimate the (posterior) probability of writership.



Handwriter 1.0 is an open-source R package developed to pre-process a handwriting sample and decompose it into graphs. The investigator scans the handwritten documents and saves them as PNG image files. The following sections explain each step in more detail.

Future Releases of handwriter will allow users to create clustering templates, build writer profiles and estimate the probability of writership of a questioned document.

Handwriter 1.0

Step 1: Computational Pre-Processing:

We will now explain how handwriter pre-processes a handwriting sample using the following handwritten word as an example.

1. Installation: The investigator installs the handwriter package into an open session of R. A step by step for installation can be found in the "Where to Find It" section.

2. Binarization: Handwriter does two things at once with the *readPNGBinary* function: it imports a PNG image and turns the image's pixels into either black or white. This process lets R know where to find handwriting in the image.

3. Skeletonization: This step is done to crop and thin down the handwriting. Cropping is done with the *crop* function, making the writing cleaner and easier to read. The thinning process is done by the *thinImage* function which reduces the writing to a 1-pixel wide skeleton. Structural components of the writing are easier to identify in the cropped and thinned version.

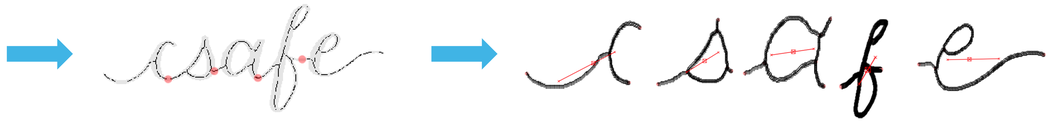
4. Decomposition into graphs: Handwriter breaks the skeletonized writing into connected pieces of ink. These pieces might be a cursive word or a single printed letter. Handwriter then identifies potential breakpoints in the connected pieces, where the breakpoints estimate breaks between individual letters. The *processHandwriting* function converts the writing segments in-between breakpoints into graphs. These graphs roughly correspond to Roman letters and form the basis for the statistical modelling part of the project.

1. Original image
Image before being scanned into handwriter

2. Binarization
Image gets scanned in and turned into black and white

3. Skeletonization
Image gets cropped and reduced to a 1-pixel wide skeleton.

4. Decomposition
Image gets broken into graphs that may correspond to Roman letters.



5. Breaks
After decomposing, handwriter knows where the word breaks into potential letters.

6. Graphs
Handwriter now has access to individual graphs on which it takes measurements.

Future Releases

Step 2: Create Clustering Template

Amy Crawford and Nick Berry developed R code to create clustering templates, build writer profiles, and identify the writer of a questioned document from a closed set of writers [1, 2]. We plan to incorporate this functionality into future releases of handwriter and briefly describe each of these steps here.

In the future, handwriter will be able to create a clustering template from a reference set of handwriting samples from a variety of writers. (These handwriting samples can come from publicly available handwriting databases such as [3] and do not need to include samples from the potential writers.) Handwriter will pre-processes these samples, and then apply a k-means type algorithm to group similar graphs into clusters. The produced clusters are called a clustering template and can be used to construct a writer profile for each potential writer. For more details, see [1].

Step 3: Build Writer Profiles

Each writer can be characterized by the rate in which graphs from their handwriting samples are assigned to various clusters in the clustering template. A future release of Handwriter will use a Bayesian hierarchical model and performs Markov Chain Monte Carlo (MCMC) estimates to estimate the true rates at which each writer produces graphs in each cluster. For more details, see [1].

Step 4: Writer Identification

In order to identify the writer of a questioned document from the set of potential writers, handwriter will pre-process the document (current release) and assign its graphs to clusters in the clustering template (future release). Then handwriter will use MCMC estimates from the model in step 3 to find $\vec{p} = (p_1, \ldots, p_n)$ the posterior probability of writership for each writer in the set of n potential writers. More specifically, p_i is the probability that the questioned document was written by writer i wrote the questioned document. For more details, see [1].

Where To Find It:

A guide on how to install and use the Handwriter package can be found at:
csafe-isu.github.io/handwriter

The package is open-source and the code can be viewed at:
github.com/CSAFE-ISU/handwriter

Scan the QR code to easily find the package.

References

[1] Crawford, Amy M., Nicholas S. Berry, and Alicia L. Carriquiry. "A clustering method for graphical handwriting components and statistical writership analysis." *Statistical Analysis and Data Mining: The ASA Data Science Journal* 14, no. 1 (2021): 41-60.

[2] Crawford, Amy M. "Bayesian hierarchical modeling for the forensic evaluation of handwritten documents." PhD diss., Iowa State University, 2020.

[3] "The CSAFE Handwriting Database." https://data.csafe.iastate.edu/HandwritingDatabase/

This work was funded (or partially funded) by the Center for Statistics and Applications in Forensic Evidence (CSAFE) through Cooperative Agreements 70NANB15H176 and 70NANB20H019 between NIST and Iowa State University, which includes activities carried out at Carnegie Mellon University, Duke University, University of California Irvine, University of Virginia, West Virginia University, University of Pennsylvania, Swarthmore College and University of Nebraska, Lincoln.