HAnnoI: A Handwriting Annotation Interface to Extract Data for Linguistic Analyses of Graphetic Detail

Joshua Wieler[1,3], Simon Petitjean[1], Kristian Berg[1], Henriette Huber[2], Stefan Hartmann[2]

[1]Carl von Ossietzky University of Oldenburg, Institute for German Studies
simon.petitjean, kristian.berg@uni-oldenburg.de

[2]Heinrich Heine University Düsseldorf, Department of German Linguistics
henriette.huber, hartmast@hhu.de

[3]Ruhr University Bochum, Linguistics Department
joshua.wieler@ruhr-uni-bochum.de

Abstract

In this paper, we present *HAnnoI* – short for *Handwriting Annotation Interface* –, an open-source GUI application developed in Python that allows its users to identify and annotate so-called Regions of Interest (ROIs) within digital images. Several meta data such as their coordinates are retained for each ROI and they can be annotated on user-defined annotation layers. HAnnoI comes with a function to export all annotations to a CSV file, enabling further processing as well as quantitative analyses. HAnnoI also has a function to extract single PNG image files of all ROIs. It was developed to mark and annotate single letters in scans of handwritten (alphabetic) texts for linguistic analyses, yet it is not limited to this particular use case. In this paper, we first provide information on HAnnoI's conception and technical details as well as an overview of alternative applications. We then showcase HAnnoI's capabilities in a letter annotation task, where five annotators marked instances of lower case <s> in handwritten texts. Finally, we report on an exploratory analysis of this data, showing what kinds of investigations are enabled by using HAnnoI. The tool is available for free use at https://github.com/pywielR/HAnnoI.

Keywords: Tools, Systems, Applications, Handwritten, Typewritten Document Recognition, Corpus (Creation, Annotation, etc.)

1. Introduction

This paper presents *HAnnoI*, the *Handwriting Annotation Interface*, a lightweight, platform-independent, easy-to-use GUI application developed for identifying and annotating single letters in handwritten data. Originally designed for working with scans of texts written by hand in an alphabetic writing system, the tool can be used to process just about any kind of visually available, digitized (language) data. If one thinks of an image file as a map, HAnnoI provides a toolkit that allows users to chart its geography by adding rectangular frames modifiable in size, color and position. Each frame marks a so-called Region of Interest (ROI), whose measures and coordinates are being tracked by the program, and that can be enriched with a variable amount of user-defined annotations. HAnnoI also allows for exporting the ROIs as individual PNG image files and the annotations as a CSV spreadsheet, where each row represents one ROI with all of its metadata and "geographical" information as well as its custom annotations retained in individual columns. This data may then be used for further processing and quantitative analyses. Furthermore, it can be imported back into the tool to continue working on a document.

HAnnoI's built-in function to save individual PNG images of all ROIs furthermore enables investigations of the graphetic part of language production, i.e., the concrete shape that linguistic units take in handwriting. Because each extracted image is linked to its annotations in the CSV spreadsheet via an index, and said CSV may be imported back into the program, HAnnoI makes it very easy to trace back, inspect and revise previous work, making it a powerful resource for data documentation as well.

HAnnoI was developed to handle specific problems that we encountered during the annotation of handwritten letters in an alphabetic, left-to-right directed script. This is of course reflected in the features currently implemented in the tool, which may also limit its applicability to other scripts, especially those that do not adhere to a horizontal writing direction. We still suggest that it can be useful even beyond its original purpose of annotating single letters. For example, users might choose to annotate whole words or even sentences, handwritten or in print. Any field researching such data may be supplemented using HAnnoI, for example, studies on

Proceedings of the Third Workshop on Computation and Written Language (CAWL 2026) @LREC 2026, pages 33–44
12 May, 2026. ©ELRA Language Resources Association (ELRA), 2026

graphetic variation (e.g., Reinken, 2023), but also Optical Character Recognition (OCR) technology (see Memon et al., 2020 for a review on handwriting specifically).

Because the tool tracks geographical information, it allows for analyses of spatial relations between annotated units. In research on handwriting, this could involve investigations of the influence that adjacent letters have on each other. Kandel and Perret (2014), for example, have observed stroke length differences in cursive realizations of the letter <l> when it was connected to a following <l> as compared to when it was connected to a following <e> or <n>, indicating anticipatory effects in handwriting, a notion they refer to as *motor anticipation*. While this particular study had children produce isolated bigrams using a digital pen and writing tablet, with HAnnoI, this could be extended to naturalistic data like scans of handwritten texts, as can be found in the GraphVar Corpus (Berg et al., 2021) that will be introduced in more detail in Section 3.

Also, suppose a researcher has collected scans of written documents and is interested in a specific linguistic phenomenon, yet the data has not been transcribed and annotated yet. In this case HAnnoI can be used to identify linguistic units of interest and annotate them on a custom set of layers, while simultaneously tracking their position. Both the units of interest as well as the annotations can be extended at a later stage of the project if necessary. All too often, researchers still use pen and paper tally sheets in these cases, which is not only cumbersome but also more error-prone – HAnnoI offers a significant improvement while at the same time ensuring reproducibility.

The remainder of this paper is structured as follows: First, we compare HAnnoI to similar applications that are available for free use. After that, in Section 2, we provide a more detailed description of the tool's functionalities. This is followed by a concrete application example in which five annotators were instructed to annotate instances of lower case <s> in handwritten documents (Section 3). These annotations have then been used in an analysis of <s> variation in handwriting, specifically, whether there are variables that can predict the usage of either cursive or block letter <s> (see Figure 1 for an example of this variation). Further details on the analysis will be provided in Section 4. The paper closes discussing some of HAnnoI's current limitations, as well as providing an outlook on further developments.

1.1. Comparable tools

Most tools specifically designed to annotate linguistic data focus on already digitized texts (e.g., INCEpTION, Klie et al., 2018; doccano, Nakayama et al., 2018) or video/audio recordings (ELAN, Sloetjes and Wittenburg, 2008). When it comes to undigitized texts, eScriptorium[1] can be used for transcription of handwriting. It also allows users to mark certain regions in an image, though this feature is not suited for multi-layer annotations as it only allows for one label per marking.

This limitation is also prevalent in other tools we tested, such as the web-based application makesense.ai by Skalski (2019) and PixLab Annotate[2]. These tools are more oriented towards object detection and labeling (generating training data for machine learning), not towards extensive linguistic annotation, which explains their limitations with regard to the accumulation of multivariate data.

The applications that are closest to HAnnoI in functionality are the VGG Image Annotator (VIA) by Dutta and Zisserman (2019) and IMMARKUS by De Weerdt et al. (2024). Both tools are web-based and allow their users to (1) mark ROIs of different shapes within an image (e.g., individual letters/words in scans of handwritten text), and (2) add annotations on multiple layers to each ROI. However, for VIA, we found that the marking process is less precise than in HAnnoI and it also does not allow for the extraction of the marked areas as PNG or similar image files. As for IMMARKUS, it is a powerful tool that brings many of HAnnoI's features and offers additional functionality such as smart scissors and auto transcription, making it a good choice in a wide range of situation. We encourage anyone who is interested in conducting research similar to what is described throughout this paper, to also test IMMARKUS for its applicability, as depending on what their objective is, it might suit their specific needs better.

Beyond VIA and IMMARKUS, we did not find many other tools that can be used in a similar manner. Mick O'Donnell's UAM ImageTool[3] provides some of the same functionalities, though it lacked precision when drawing ROIs and seems to be no longer maintained, with its latest version having been released in 2011. Note that in our comparison we did not consider applications that require a paid subscription such as Transkribus[4].

2. Technical details

The code for HAnnoI is written in Python, using the PyQt framework (version 6) for GUI development. The tool runs from a single Python script that loads all of its dependencies upon launch (they have to be installed once beforehand). The source code and a quick installation guide can be found here: https://github.com/pywielR/HAnnoI.

[1] https://ocr-bw.bib.uni-mannheim.de/escriptorium/
[2] https://annotate.pixlab.io/
[3] http://www.wagsoft.com/ImageTool/index.html
[4] https://www.transkribus.org/de

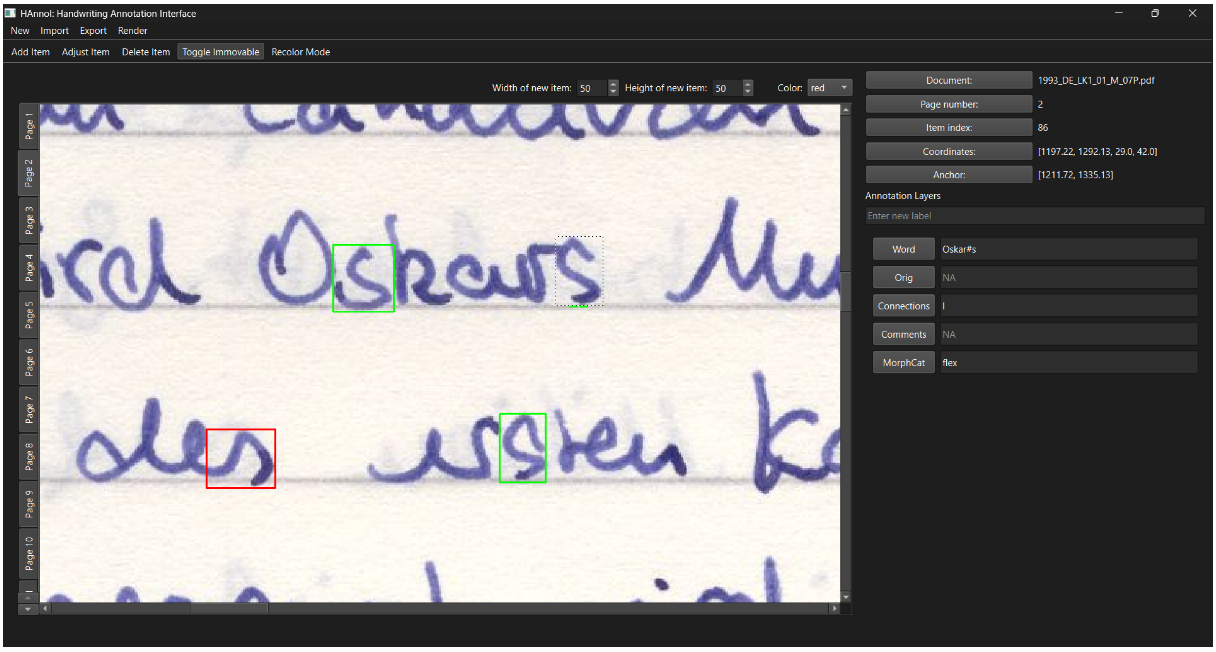


Figure 1: HAnnoI's interface displaying part of one page of a document containing handwritten text. In this zoomed-in view, one of the four visible ROIs is currently selected (indicated by the dashed lines); its corresponding annotations are shown in the panel on the right. Different colors have been used to mark either cursive instances of <s> (red) or block letters (green).

Besides the menu bar at the top, which provides options to start, save and continue with a project, HAnnoI's user interface consists of two main areas (see Figure 1): (1) the graphical view on the left, in which the user may add ROIs to imported images; and (2) the annotation panel on the right.

2.1. Graphical view and meta data

In HAnnoI, it is possible to either load in single image files or multiple images extracted from a PDF document. The user may zoom in and out of the graphical view as well as drag it around with the mouse. If an imported PDF document consists of multiple pages, each page gets an individual tab on the left-hand side of the graphical view. The header above offers buttons for adding, modifying and deleting ROIs, though some of these actions are also covered by keyboard or mouse inputs. For example, ROIs can be added by right-clicking into the displayed image and selecting *Add Item* from a context menu. An ROI can be selected and dragged around using the mouse. One can also use arrow keys to adjust its position. By holding Ctrl or Shift, arrow keys change a ROI's size in 1- or 5-pixel-increments, respectively. For a full list of HAnnoI's shortcuts, we refer to the manual in the GitHub repository.

By default, the annotation panel on the right displays the meta information for any selected ROI, such as the source document, the page number, the index as well as its coordinates and size[5]; furthermore, there is the option to set a so-called anchor for each ROI (the green dashed line just below the selected ROI in Figure 1), which in the example here was utilized to track the position of the horizontal line on the writing paper which the ROI-corresponding word was written on. Anchors are always set along the bottom edge of a selected ROI by pressing Space. A set anchor retains its position even when moving the ROI it belongs to, but can be overridden by pressing space again.

2.2. Custom annotations

Custom annotation layers can be added by entering a label in the designated field and pressing Return. Five of these layers have been added to the example shown in Figure 1 (*Word*, *Orig*, *Connections*, *Comments* and *MorphCat*). Currently, all annotation layers are text input fields. After selecting a ROI, users may type in their annotations. Additionally, it is possible to implement a categorical variable after which ROIs can be annotated with a certain feature by simply selcting it with the mouse. The user manual in the GitHub repository contains more detailed instructions on how to use this as

[5]The coordinates and size of a selected ROI are displayed together as a list in the *Coordinates* annotation layer in this order: x-coordinate, y-coordinate, width, height. The x- and y-coordinates always refer to the upper-left corner point of a rectangle.

well as other features implemented to streamline the annotation process.

2.3. Exporting data

While working with HAnnoI, meta data as well as annotations are stored in standard Python dictionaries, which are compiled into CSV format upon export. For an example of the output, see Appendix A1. Exporting annotations is also the way to save one's work, since the resulting CSV can be imported again in order to continue with the file. The corresponding single image or PDF document has to be retained by the user as a separate file, however, because the source of the image(s) itself is only included as the file's name in the CSV (in the Source column). For easy import, it is recommended to just keep the CSV and source file in the same folder, in which case HAnnoI automatically loads in the corresponding images. If the source file is not in the same folder, the user is asked to select it manually.

CSV files can be edited outside of HAnnoI and still be imported. For example, one could manually change the entries in the color column in order to change the visual appearance of ROIs inside the program. One should be cautious, however, as some changes, in particular those to the meta information or the column structure, may cause HAnnoI to fail upon trying to import a CSV file. As for the custom annotation layers, it is possible to add whole new columns at the right end of the spreadsheet outside of HAnnoI, which would then be recognized as being one of the custom annotation layers inside the tool. This allows for working on a file in and outside of the tool, offering flexibility to the annotator's work.

Finally, the function to render single PNG image files for each ROI placed in the document can be found at the right end of the upper menu bar. The file name of these extracted images contains a reference to the source as well as an index that links each image to its entry in the corresponding spreadsheet. As the CSV export function and the rendering function are independent from each other, one should keep in mind that changes to the indices in the CSV file do not carry over to the PNG files until one uses the rendering function once again. Extracted images display exactly the material that was framed, so a 29x42 pixel ROI would be saved as a 29x42 pixel PNG file. The colored frames of ROIs are hidden in the extracted images. Appendix A2 gives an example of a folder containing rendered images extracted from a multi-page document that has been annotated with HAnnoI.

The next two sections demonstrate in two complementary ways the functionalities of HAnnoI in practice. Section 3 presents a concrete annotation task in which multiple annotators gathered handwritten data using the tool, allowing us to assess its usability and the quality of the collected data. Section 4 then shows how the annotated data can be immediately used for quantitative linguistic analysis. Together, these two examples show a typical research workflow that HAnnoI is meant to support, making data collection faster, easier to reproduce, and convenient to maintain and extend over time.

3. Annotation example and assessment

HAnnoI has been tested on PCs running on Windows as well as macOS, in an annotation task involving five student assistants. After ascertaining that the system's core features are properly functioning, we allocated a set of German school-exit exams taken from the GraphVar Corpus (Berg et al., 2021) to the annotators. Their task involved framing all instances of lowercase <s> and <e> as ROIs. Each of these letters was then annotated with the (orthographically corrected[6]) word form it was a part of. Morpheme boundaries within that word annotation were marked using the special sign #. HAnnoI's Annotation Mode was used to furthermore annotate morpheme categories, for example, whether a letter was part of the word's stem or of an inflectional suffix. Figure 1 as well as Appendix A provide a snippet taken from the results of this annotation task.

Before HAnnoI was introduced to the student assistants, they had already collected data in a similar annotation task, though instead of marking and annotating ROIs using a dedicated tool, they were instructed to take individual screenshots of either <s> or <e> and catalog these screenshots in a spreadsheet, adding annotations along the way. To document a letter's position in an exam, the coders manually annotated the page and line a letter was written on. In HAnnoI, positional information is tracked automatically and with higher precision as it retains the x- and y-coordinate of any ROI. Additionally, indexing is automatized in HAnnoI as well. Before, the coders had to annotate indices themselves. While it is hard to quantify whether the implementation of HAnnoI brought an increase in efficiency, for this application example, we argue that, when compared to before, the strongest asset of the tool is the traceability it provides for the data that is produced with it, and that this necessarily culminates in data that is of higher quality. To justify this claim, we will elaborate on two examples that were taken from the outlined annotation task.

As of the time of writing, only using HAnnoI and not regarding the data that had been collected be-

[6] For spelling errors, we created a separate layer labeled *Orig*.

fore, the coders have gathered annotations for 7056 instances of <s> across twelve exams as well as for 2084 instances of <e> in a single exam. While the annotators mostly worked on different exams, we sometimes assigned the same exam to multiple coders in order to assess agreement across annotators as well as the functionality of HAnnoI.

Example I: Working on the same file

In case of the annotation of <e>, we had a first annotator frame all instances of this letter as ROIs alongside annotating the corresponding word. After that, the second annotator was provided with the resulting CSV to load into the program, effectively reproducing all of the first annotator's work in an instant. The second annotator then added information on morpheme category as well as noting whether a framed letter had a visible connection to adjacent material (such as is the case for the selected <s> in Figure 1). In HAnnoI, revisions like this are rather uncomplicated, as it brings together every annotation and its visual reference in a single interface. Before introducing the tool, this would have required the second annotator juggling the screenshots, the spreadsheet and the source file, to manually search the latter for every instance of <e> and then update the annotations, rendering this approach very slow and prone to errors, if it was at all feasible. Besides updating entire files, with HAnnoI one can also easily trace back and edit the annotations of single observations.

Example II: Comparing annotations

As for the <s> annotations, in one case we asked two student assistants to annotate two pages of the same exam. This was done so as to be able to directly compare the work of different annotators on identical data. With this, we calculated inter-annotator agreement mainly as an (early) evaluation of our annotation guidelines and method, though aspects of this can be regarded as indication of HAnnoI's viability.

First, we compared the anchors set by the annotators for each ROI. To reiterate, the instruction was to place these anchors on top of the horizontal baseline an annotated letter was written on. Over the two pages, 94 pairs of <s> annotations were collected by the student assistants. On average, the distance between the anchors was about 0.55 pixels on the y-axis (which is the axis that matters here). Given that the lowest increment to which adjustments to the anchors can be done (via keyboard inputs) is one pixel, we concluded that the annotators aligned their anchors quite nicely, demonstrating that HAnnoI allows for gauging such precise information.

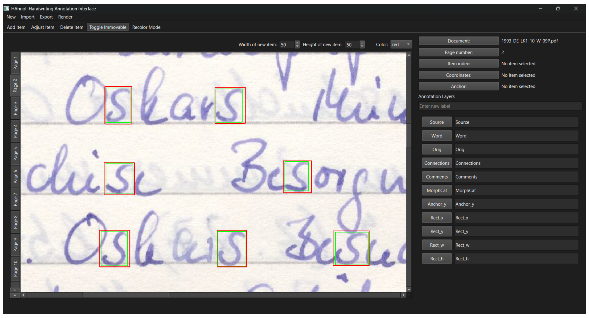


Figure 2: ROIs added by separate annotators during an early stage of the annotation process; red frames were added by the first annotator and green frames by the second annotator.

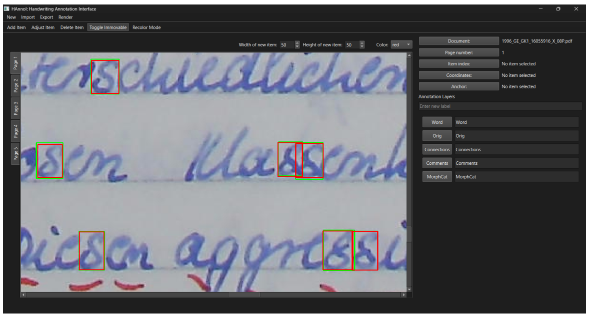


Figure 3: ROIs added by the same annotators from Figure 2, this time during a later stage of the annotation process and after receiving feedback on their earlier framing of ROIs; red frames were again added by the first annotator and green frames by the second annotator.

Furthermore, we used the <s> annotations provided by both annotators to assess how closely they drew ROI frames around identical letters. To achieve this, we copied and pasted the annotations from the second annotator into the CSV file of the first annotator, which, after loading the combined data into HAnnoI, places the frames from both annotators on top of each other. In the spreadsheet, we manually changed the color of the ROIs of the second annotator from red to green, so that after importing, the contribution of both annotators would be distinguishable. With this, we were able to immediately get an impression of the differences between the two annotators.

As can be seen in Figure 2, the green frames were drawn more closely around the letters. On average, the 94 ROI frames placed by the first annotator were about 3.46 pixels wider and 3.22 pixels higher than those placed by the second annotator[7]. As tighter frames give less room for intrusions from adjacent letters, in a discussion of these findings,

[7] To calculate this difference, just as the difference between the anchor placement, we imported the combined data into R.

we concluded that going forward we wanted the frames to fit the letters as close as possible. These and other points discussed were documented for the student helpers to refer to in the continuation of the annotation task. Of course, the annotators could also revisit the annotations they had already collected and adjust their ROIs' frames directly in the tool. Without HAnnoI, such an undertaking would require the students to search the documents for instances of <s> or <e> again (as there was no visual marker in the document itself before) in order to take new screenshots, effectively meaning they would have to repeat a task that had already been done before.

At a later stage, we once again provided the two annotators from above with identical data to annotate in HAnnoI. We repeated the comparison of ROI sizes, though this time on a database comprising 438 pairs of <s>. We found that the difference between both annotators had come down by about 50%. In fact, this time the ROI frames placed by the second annotator were about 1.64 pixels wider and 1.57 pixels higher than those of the first annotator. This suggests that after getting feedback on their annotations, the first annotator paid special attention to drawing the frames as close as possible. This is supported by Figure 3, which exemplifies how the frames of both annotators are drawn rather closely around the letters. Furthermore, we take this as indication that, on the technical side, HAnnoI provides an adequate tool set to realize fine adjustments such as those of the first annotator.

Finally, we would like to point out that this assessment in itself shows that the data that HAnnoI provides can be used in qualitative as well as in quantitative research. We now turn to another example, in which we used part of the data generated during the annotation process to investigate <s> variation in handwriting.

4. Application example: exploring <s> variation in handwriting

The annotation work described in Section 3 was initiated as part of a research project investigating morphologically induced variation in handwriting. In one branch of this project, Petitjean et al. (n.d.) extracted the pen trajectories from individual realizations of <s>, for which single screenshots of these were collected. The extracted trajectories then served as the dependent variable of statistical models including morpheme category, morpheme position, syllable position as well as letter position as predictors.

One of the central findings of this study was that variation in writing trajectories can indeed be partially explained by morphological affiliation. For example, <s> realizations found in inflectional suffixes tend to be somewhat reduced in their shape (i.e., the upper part of <s> does not travel as wide of an arc) when compared to those found in stems. When <s> is part of a derivational morpheme, however, the difference to <s> in stems becomes smaller. Petitjean et al. (n.d.) argue that this supports the notion that the morpheme is a significant processing unit in (written) language production.

During the annotation work, we also noticed that in some of the exams, when writing <s>, the students switched between two types of script, namely cursive and block letters (see Figure 1). Following Meletis (2020), this could be regarded as alternation on the level of *basic shapes*, i.e., abstract forms of <s> that are visually distinct in the sense that their components such as individual strokes use different forms (e.g., straight lines vs. arcs) or connect to each other in different angles. Meletis also remarks that this kind of switching within the same text is rather unusual (p. 114), rendering the observation of this phenomenon in our data an interesting case to provide another demonstration of what kinds of analyses are enabled by using HAnnoI as an annotation tool.

4.1. Database of different <s> types

For our analysis, we only considered exams in which we found constant variation between the different types of <s>. This was the case for five exams out of the twelve mentioned in Section 3. As Table 1 shows, in these exams, we found a total of 3383 instances of <s>. The distribution of <s> types in our data was rather balanced with 1876 written in cursive script (55.5%) and 1507 written as block letters (44.5%).

We instructed the annotators to use red frames to mark what we defined as cursive <s> and green frames for block letters. Note that this binary classification is based mainly on the starting point and first stroke of <s>, where cursive <s> is defined by starting on the left, from where its first stroke travels, in some angle to the baseline, in an upwards-right direction (for comparison, see the one <s> marked in red in Figure 1). In contrast, block letters are defined by the presence (or at least suggestion) of an upper arc that starts more to the right, from where it travels to the left. In this regard, we apply a more coarse classification of basic shapes for <s>, of which Reinken (2023) has defined four in total (in his sample of the GraphVar Corpus). For example, the upper arc of the first <s> in the word *Oskars* in Figure 1 is not fully realized and the whole letter could therefore be regarded as being an instance of a third basic shape. However, as Appendix A2 indicates, even within writers there is a high amount of variation in the actual physical manifestation of a letter, so for this analysis we attributed more fine-grained differences such as the exact shape of the upper arc of <s> to variance within basic shapes[8] rather than between them.

Our distinction between cursive and block letter <s> becomes meaningful when one considers the writing direction of German, which goes from left to right. What we defined as cursive <s> can therefore easily connect with previous letters, whereas block letters start on a movement that opposes the natural writing direction of German, likely causing a lift of the pen. This notion is supported by our annotations: Using the *Connection* annotation layer, we checked how often cursive and block letter <s> connects with adjacent material. In 1270 (67.7%) out of 1876 cases, cursive <s> connects with a previous letter. For block letter <s>, there is a connection in 312 (20.7%) out of 1507 cases. For connections to following letters, this difference is much smaller, with 318 (17%) connections to the right involving cursive <s> and 402 (26.7%) involving block letter <s>. Note that in the *Connection* annotation layer, every time <s> so much as touches an adjacent letter, this is counted as a connection, meaning they are not necessarily indicative of a fluent writing motion where one letter goes into the next without lifting the pen[9]. However, the stark contrast between cursive and block letter <s> with regard to previous letters strongly suggests that this has to do with the fact that cursive <s> can more easily connect to previous letters.

Exam	**Instances of <s> written in ...**		
No.	*cursive script*	*block letters*	*total*
1	452 (47.7%)	495 (52.3%)	947
2	156 (25.3%)	461 (74.7%)	617
3	446 (69.3%)	198 (30.7%)	644
4	388 (69.5%)	170 (30.5%)	558
5	434 (70.3%)	183 (29.7%)	617
	1876 (55.5%)	1507 (44.5%)	3383

Table 1: Instances of either cursive or block letter <s> and their distribution across five handwritten exams.

4.2. Data processing and modeling

The annotated data underwent further processing in a separate Python script where we extracted information such as the position of each <s> in the word as well as in the morpheme it belonged to. We also extracted the immediately preceding and following letter of each observation. This data was then imported into R for the statistical analysis.

From the color coding applied in HAnnoI, we implemented a binary variable, specifically, whether a certain <s> was written in cursive script (TRUE) or as a block letter (FALSE). This served as the dependent variable of a mixed effects logistic regression model (using the `glmer` function in R).

Considering that cursive <s> connects to previous material more often, we wanted to explore whether connectivity can be attributed to certain letters, for which we included random intercepts for the preceding, but also the following letter. Though excluded from our sample, there are exams in which there is no variation between <s> types at all, which is why we assumed that this phenomenon is highly idiosyncratic. We therefore included random intercepts for the ID of each writer as well as for each target word.

With our fixed effects, we explored the possibility of (1) positional, (2) morphological and (3) geographical influences on the choice of <s>. Specifically, we included the position that <s> takes within a word as well as within a morpheme. Both were coded categorically, where <s> could occupy *initial*, *medial* or *final* position. For morpheme position, we implemented a fourth level labeled *single*, which refers to instances of <s> that make up the whole morpheme, such as is the case in the inflectional suffix marking of the genitive in the word *Lesers* ('reader-GEN'). Prior research suggests that the morpheme might be a relevant processing unit during written language production (e.g, Kandel et al., 2012; Berg et al., 2023). With our operationalization, a morphological effect could show in that the model returns a lower probability for cursive <s> to be realized in morpheme *initial* and *single* position, where a connection to a previous letter would be impeded by a morpheme boundary. Reinken (2023) reports a similar effect for syllable boundaries, where identical bigrams are less likely to connect as compared to when they are within the same syllable.

Furthermore, we added a factor signaling whether the previous <s> in the text was cursive or not. With this factor, we wanted to approximate priming effects, i.e., whether writers tend to repeat the same type of <s>. However, this variable on its own is blind to the distance between subsequent instances of <s>, meaning that a preceding <s> could be located on the same line, likely strengthening its priming effect, just as well as on a previous page. Using the geographical annotations in our data, we therefore implemented a continuous variable that measures the vertical distance (in pixels) between the anchor points of subsequent instances of <s>. We standardized the pixel data to deal with convergence issues during model calculation and then included it in the model as an interaction with previous <s> type. Because this measure cannot

[8] According to Meletis (2020) as well as Reinken (2023) this type of variation can be explained by factors such as the writing situation (here: exam), the writer's fatigue, the pen they use and more.

[9] The main idea behind this layer was to be able to identify instances of <s> without intrusive material.

be calculated for the first <s> on each page, we excluded these from the analysis. Our model was therefore fit on a database comprising 3300 observations.

Finally, we also included a measure of how far each <s> deviates from its baseline, calculated as the (standardized) vertical distance between the bottom edge of a rectangle and its corresponding anchor. As the starting point of block letter <s> is likely to be farther away from the baseline, maybe this results in a different spatial alignment relative to the baseline.

We did not include the height of rectangles in the model, as from a conceptual stand point, we thought that the measurements of any particular <s> would be a consequence of what type had been realized rather than it being the cause. We did check the relation between height and <s> type separately, however, and found that instances of block letter <s> are on average about 1 to 11 pixels higher than cursive <s>, depending on the writer. As for width, instances of block letter <s> are typically more narrow (about 6 to 12 pixels), though the left and right boundary of letters often cannot be as clearly defined as their height, especially when they connect to other letters.

4.3. Results

Our model yields a marginal R^2 of 0.03 and a conditional R^2 of 0.77 (using the theoretical method, see Nakagawa et al., 2017), so while our fixed effects only account for very little variance, our random effects exhibit a much larger impact.

We assessed the significance of our fixed effects using likelihood ratio-tests with single term deletions. For the position that <s> takes inside a word, we found only a marginally significant difference between our model and a model without this predictor ($\chi^2(2) = 5.317$, $p = 0.070$); the model's coefficients show a significant contrast between <s> in word medial and final position only, in that medial positions are more likely to be occupied by block letters ($\beta = -1.09$, SE = 0.39, z = -2.81, $p = 0.005$).

For the position that <s> takes within morphemes, we did find a significant effect ($\chi^2(3) = 12.883$, $p = 0.005$), with one highly significant contrast between morpheme initial and medial position, where block letter <s> is more likely to occur ($\beta = -0.98$, SE = 0.28, z = -3.46, $p < 0.001$). These positional results go against our expectation that we would find more realizations of cursive <s> in medial positions as connections to adjacent letters would not be impeded by morpheme boundaries. One fact that our model does not account for is that instances of <s> in word initial position always coincide with morpheme initial position (yet not vice versa), which possibly confounds this finding. However, looking into our data, we find that there are

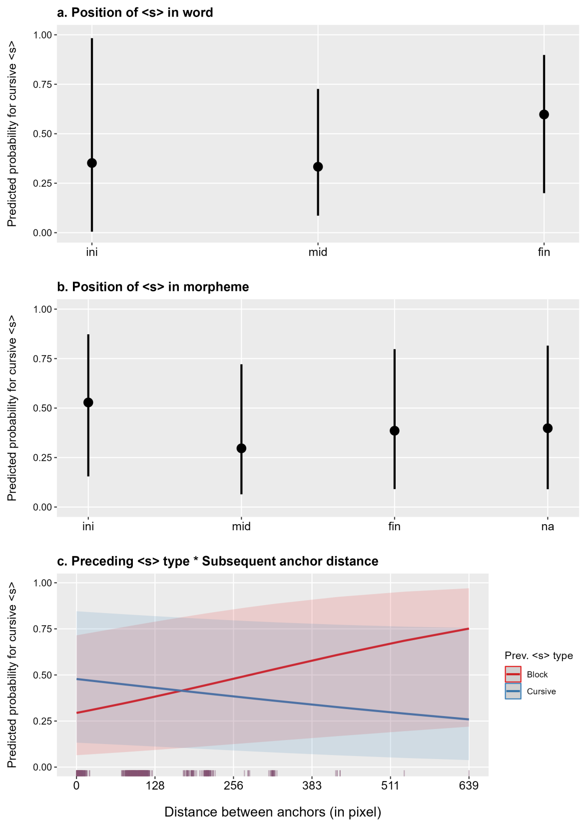


Figure 4: Visualization of the significant fixed effects in the model. For panel c. the scaled anchor distances used during model calculation were transformed back to their original scale. The rugs at the bottom of panel c. display the distribution of anchor distances in the data.

actually more cases of block letter <s> in word initial position (349) than there are of cursive <s> (299). Taking these out of the equation would therefore rather strengthen the effect of morpheme position that we found in our model.

As for the effects involving geographical information, we did find that the interaction involving previous <s> type and the vertical distance between subsequent anchors made a significant difference ($\chi^2(1) = 7.226$, $p = 0.007$); the vertical deviation from the baseline yielded no significant effect ($\chi^2(1) = 0.224$, $p = 0.640$). Figure 4 visualizes the significant fixed effects in our model, exhibiting highly overlapping confidence intervals in each panel, indicating that even significant effects in our model are rather small. Looking at panel c. specifically, which depicts the interaction in our model, we can see that this effect manifests as expected: at the closest vertical proximity, i.e., when it was written on the same baseline, a cursive <s> is more likely to follow another cursive <s>, yet this effect diminishes the farther apart these instances of <s> are from each other. Note that most subsequent anchors have distances (in pixels) that are either

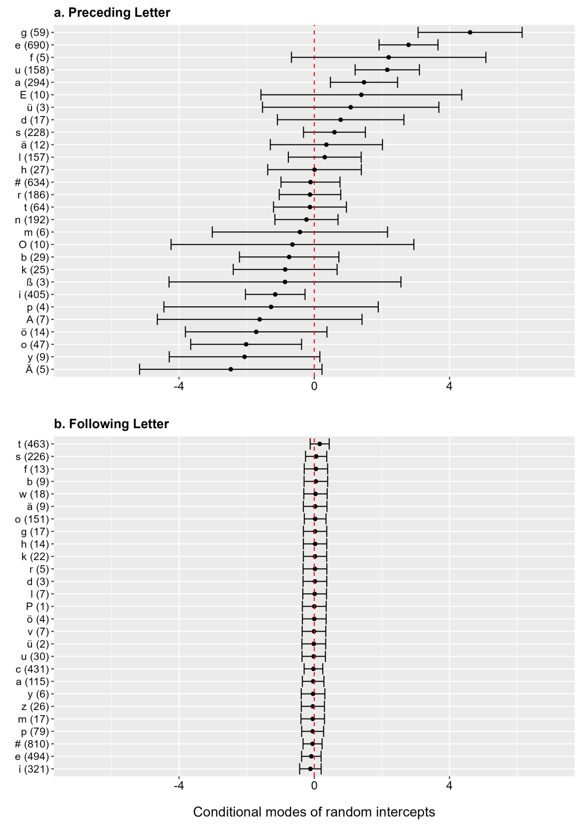


Figure 5: Conditional modes for letters that (a.) precede or (b.) follow ⟨s⟩. Values that deviate to the right (i.e., positive) side of 0 indicate a higher probability for ⟨s⟩ to be realized in cursive script; values that deviate to the left indicate a higher probability for ⟨s⟩ to be realized in block letters. The number in parentheses after each letter informs of its frequency of occurrence in the data; # signifies that ⟨s⟩ sits at a word boundary.

close to 0 (corresponding to the same baseline) or 100 (one line apart).

For the random effects, we report the variance calculated for each random intercept: 3.058 for individual words; 2.333 for individual students; 5.045 for preceding letters; and 0.032 for following letters. Evidently, preceding letters are associated with the largest variance in the outcome, i.e., whether ⟨s⟩ is realized in cursive or block letter script. In contrast, following letters cause little to no difference in the outcome. Figure 5 illustrates this contrast very clearly, where some preceding letters deviate strongly from 0, whereas following letters stay close to the center of the plot, with overlapping error bars.

Panel a. of Figure 5 suggests that the influence of preceding letters may be more pronounced for instances of cursive ⟨s⟩ than for block letter ⟨s⟩, as ⟨g⟩, ⟨e⟩, ⟨u⟩ and ⟨a⟩ especially deviate relatively far from 0 while also having rather narrow error bars. What is interesting about these four letters in particular is that in handwriting, they often end on an upward-right stroke which could connect to cursive ⟨s⟩ in one fluent motion, as its starting stroke goes in the same direction. Kandel and Spinelli (2010) have shown that complex graphemes (i.e., those consisting of multiple letters) modulate motor processes during handwritten language production, so maybe what we observe here is that some individuals anticipate the potential to seamlessly connect ⟨g⟩, ⟨e⟩, ⟨u⟩ or ⟨a⟩ with cursive ⟨s⟩, therefore processing them as chunks much akin to complex graphemes. Note, however, that our model currently does not include information on whether there actually is a connection between a preceding letter and ⟨s⟩. So while a preceding ⟨g⟩, for example, is generally associated with cursive ⟨s⟩, this does not mean that there necessarily is a connection between these two letters. Our reading of this effect should therefore be regarded as more of a starting point to further inquiry. Nevertheless, what we found is consistent with the idea of *motor anticipation* (Kandel and Perret, 2014) and other accounts which state that context has a role to play during motor production of handwriting (e.g., Meletis, 2020; Reinken, 2023). While in here, we investigated this by means of random intercepts, in an updated model, certain letters could be grouped based on specific features of their form, for example, whether they end on a stroke that could connect to a following letter. This could then enter the model as a fixed instead of a random effect. To retrieve (and annotate) such detailed descriptions, we could of course make use of the functionality that HAnnoI offers, extending our data to our needs.

5. Conclusion and outlook

In this paper, we presented a method to gather and annotate visual (linguistic) data using HAnnoI. We introduced in detail the tool's functionality as well as illustrated possible (and concrete) use cases. As HAnnoI's source code is publicly available, users may decide to extend its functionality to their own desire. From the perspective of the authors of this paper, HAnnoI's development is also still ongoing, with plans to address some of its current limitations (which we will outline below). Nevertheless, we already regard HAnnoI as a fully operational annotation and, more broadly, data collection tool, and we suggest its usage to scientists coming from many areas, including but not limited to research of handwriting.

Limitations

One current limitation of HAnnoI is that it only allows for placing rectangular ROI frames within an image. Other shapes are not possible at the moment. Of course, in HAnnoI, the main purpose of an

ROI is to connect an annotation with its visual reference. In contrast to some of the tools discussed in Section 1.1, HAnnoI was not designed to generate training data for machine learning tasks such as object detection, where polyline shapes would be far more beneficial. Still, even in linguistic research, being limited to rectangles might pose a serious hindrance. This would be the case, for example, when the script under investigations exhibits units that cannot be appropriately framed using rectangles.

Also, while not necessarily a limitation imposed by the rectangular shapes, the fact that the bottom line of an ROI is used to set an anchor, and said anchor is displayed by a horizontal line, utilization of this specific feature does favor situations in which vertical relations between the ROI and its anchor are of interest (such as in our case). This is not to say that anchors can only be used in this way, its positioning is completely up to the user, but it is certainly more convenient to use it in a similar manner as was shown in this paper.

HAnnoI is also limited in that the custom annotation layers are restricted to text input fields. While this is overcome somewhat by the Annotation Mode (and color coding) for categorical variables, additional options like drop down menus or check boxes are currently lacking.

Acknowledgments

We thank Lara-Celine Heikens, Melissa Jahn, Marco Sund, Jiabin Yao and CB for testing the program in the annotation task reported in Section 3, providing valuable data and feedback along the way. We are also grateful to two anonymous reviewers for helpful comments. The research reported here was funded by the German Research Foundation (DFG), project number 533131836.

Bibliographical References

Kristian Berg, Stefan Hartmann, and Daniel Claeser. 2023. Are some morphological units more prone to spelling variation than others? a case study using spontaneous handwritten data. *Morphology*, 34.

Kristian Berg, Jonas Romstadt, and Cedrek Neitzert. 2021. Graphvar–korpusaufbau und annotation version 1.0. *Online-Dokumentation: https://graphvar. uni-bonn. de/dokumentation*.

Hilde De Weerdt, Rainer Simon, Dawn Zhuang, Sunkyu Lee, Iva Stojević, Meret Meister, and Xi Wangzhi. 2024. IMMARKUS: Image annotation in X-MARKUS. Web application.

Abhishek Dutta and Andrew Zisserman. 2019. The VIA Annotation Software for Images, Audio and Video. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2276–2279, Nice France. ACM.

Sonia Kandel and Cyril Perret. 2014. How do movements to produce letters become automatic during writing acquisition ? investigating the development of motor anticipation. *International Journal of Behavioral Development*, 39:1–8.

Sonia Kandel and Elsa Spinelli. 2010. Processing complex graphemes in handwriting production. *Memory & Cognition*, 38(6):762–770.

Sonia Kandel, Elsa Spinelli, Annie Tremblay, Helena Guerassimovitch, and Carlos J. Álvarez. 2012. Processing prefixes and suffixes in handwriting production. *Acta Psychologica*, 140(3):187–195.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9. Association for Computational Linguistics. Event Title: The 27th International Conference on Computational Linguistics (COLING 2018).

Dimitrios Meletis. 2020. *The Nature of Writing. A Theory of Grapholinguistics*, volume 3.

Jamshed Memon, Maira Sami, and Rizwan Ahmed Khan. 2020. Handwritten optical character recognition (OCR): A comprehensive systematic literature review (SLR). *CoRR*, abs/2001.00139.

Shinichi Nakagawa, Paul C. D. Johnson, and Holger Schielzeth. 2017. The coefficient of determination r2 and intra-class correlation coefficient from generalized linear mixed-effects models revisited and expanded. *Journal of The Royal Society Interface*, 14(134):20170213.

Hiroki Nakayama, Takahiro Kubo, Junya Kamura, Yasufumi Taniguchi, and Xu Liang. 2018. doccano: Text annotation tool for human. Software available from https://github.com/doccano/doccano.

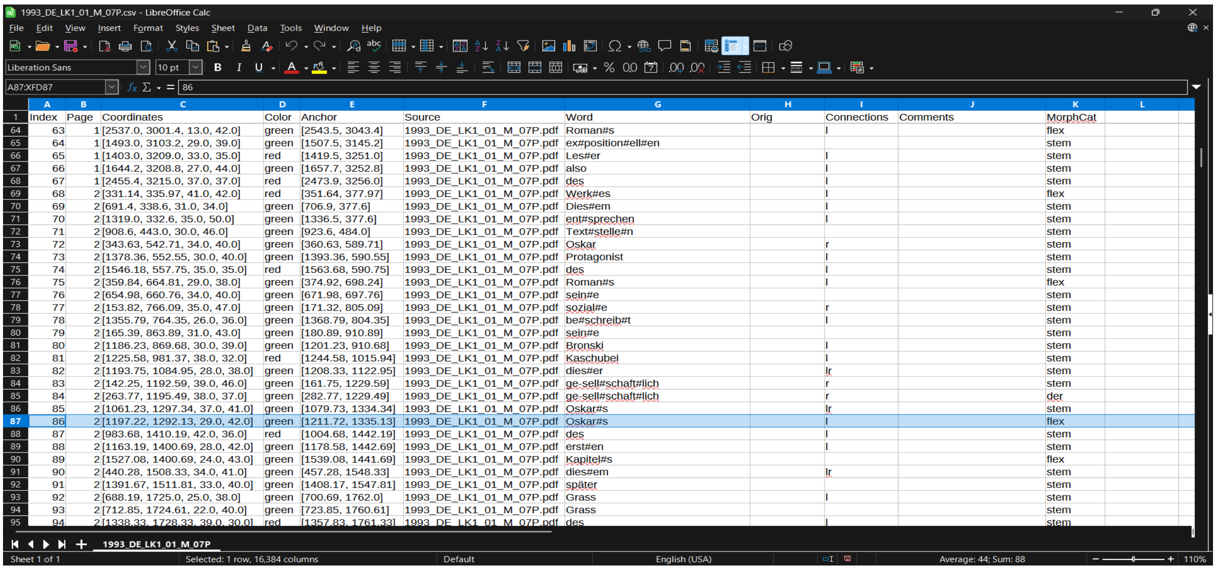
Simon Petitjean, Kristian Berg, Joshua Wieler, Henriette Huber, and Stefan Hartmann. n.d. Are handwritten letter forms shaped by morphology? Manuscript in preparation.

Niklas Reinken. 2023. *Die Grammatik der Handschriften*.

Piotr Skalski. 2019. Make Sense. https://github.com/SkalskiP/make-sense/.

Han Sloetjes and Peter Wittenburg. 2008. Annotation by category – elan and iso dcr. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Nijmegen. Cited via ELAN software page.

A. Exemplary HAnnol export files



	A	B	C	D	E	F	G	H	I	J	K	L
1	Index	Page	Coordinates	Color	Anchor	Source	Word	Orig	Connections	Comments	MorphCat	
64	63	1	[2537.0, 3001.4, 13.0, 42.0]	green	[2543.5, 3043.4]	1993_DE_LK1_01_M_07P.pdf	Roman#s		l		flex	
65	64	1	[1493.0, 3103.2, 29.0, 39.0]	green	[1507.5, 3145.2]	1993_DE_LK1_01_M_07P.pdf	ex#position#ell#en				stem	
66	65	1	[1403.0, 3209.0, 33.0, 35.0]	red	[1419.5, 3251.0]	1993_DE_LK1_01_M_07P.pdf	Les#er		l		stem	
67	66	1	[1644.2, 3208.8, 27.0, 44.0]	green	[1657.7, 3252.8]	1993_DE_LK1_01_M_07P.pdf	also		l		stem	
68	67	1	[2455.4, 3215.0, 37.0, 37.0]	red	[2473.9, 3256.0]	1993_DE_LK1_01_M_07P.pdf	des		l		stem	
69	68	2	[331.14, 335.97, 41.0, 42.0]	red	[351.64, 377.97]	1993_DE_LK1_01_M_07P.pdf	Werk#es		l		flex	
70	69	2	[691.4, 338.6, 31.0, 34.0]	green	[706.9, 377.6]	1993_DE_LK1_01_M_07P.pdf	Dies#em		l		stem	
71	70	2	[1319.0, 332.6, 35.0, 50.0]	green	[1336.5, 377.6]	1993_DE_LK1_01_M_07P.pdf	ent#sprechen		l		stem	
72	71	2	[908.6, 443.0, 30.0, 46.0]	green	[923.6, 484.0]	1993_DE_LK1_01_M_07P.pdf	Text#stelle#n				stem	
73	72	2	[343.63, 542.71, 34.0, 40.0]	green	[360.63, 589.71]	1993_DE_LK1_01_M_07P.pdf	Oskar		r		stem	
74	73	2	[1378.36, 552.55, 30.0, 40.0]	green	[1393.36, 590.55]	1993_DE_LK1_01_M_07P.pdf	Protagonist		l		stem	
75	74	2	[1546.18, 557.75, 35.0, 35.0]	red	[1563.68, 590.75]	1993_DE_LK1_01_M_07P.pdf	des		l		stem	
76	75	2	[359.84, 664.81, 29.0, 38.0]	green	[374.92, 698.24]	1993_DE_LK1_01_M_07P.pdf	Roman#s		l		flex	
77	76	2	[654.98, 660.76, 34.0, 40.0]	green	[671.98, 697.76]	1993_DE_LK1_01_M_07P.pdf	sein#e				stem	
78	77	2	[153.82, 766.09, 35.0, 47.0]	green	[171.32, 805.09]	1993_DE_LK1_01_M_07P.pdf	sozial#e		r		stem	
79	78	2	[1355.79, 764.35, 26.0, 36.0]	green	[1368.79, 804.35]	1993_DE_LK1_01_M_07P.pdf	be#schreib#t		l		stem	
80	79	2	[165.39, 863.89, 31.0, 43.0]	green	[180.89, 910.89]	1993_DE_LK1_01_M_07P.pdf	sein#e				stem	
81	80	2	[1186.23, 869.68, 30.0, 39.0]	green	[1201.23, 910.68]	1993_DE_LK1_01_M_07P.pdf	Bronski		l		stem	
82	81	2	[1225.58, 981.37, 38.0, 32.0]	red	[1244.58, 1015.94]	1993_DE_LK1_01_M_07P.pdf	Kaschubei		l		stem	
83	82	2	[1193.75, 1084.95, 28.0, 38.0]	green	[1208.33, 1122.95]	1993_DE_LK1_01_M_07P.pdf	dies#er		lr		stem	
84	83	2	[142.25, 1192.59, 39.0, 46.0]	green	[161.75, 1229.59]	1993_DE_LK1_01_M_07P.pdf	ge-sell#schaft#lich		r		stem	
85	84	2	[263.77, 1195.49, 38.0, 37.0]	green	[282.77, 1229.49]	1993_DE_LK1_01_M_07P.pdf	ge-sell#schaft#lich		r		der	
86	85	2	[1061.23, 1297.34, 37.0, 41.0]	green	[1079.73, 1334.34]	1993_DE_LK1_01_M_07P.pdf	Oskar#s		lr		stem	
87	86	2	[1197.22, 1292.13, 29.0, 42.0]	green	[1211.72, 1335.13]	1993_DE_LK1_01_M_07P.pdf	Oskar#s		l		flex	
88	87	2	[983.68, 1410.19, 42.0, 36.0]	red	[1004.68, 1442.19]	1993_DE_LK1_01_M_07P.pdf	des		l		stem	
89	88	2	[1163.19, 1400.69, 28.0, 42.0]	green	[1178.58, 1442.69]	1993_DE_LK1_01_M_07P.pdf	erst#en		l		stem	
90	89	2	[1527.08, 1400.69, 24.0, 43.0]	green	[1539.08, 1441.69]	1993_DE_LK1_01_M_07P.pdf	Kapitel#s				flex	
91	90	2	[440.28, 1508.33, 34.0, 41.0]	green	[457.28, 1548.33]	1993_DE_LK1_01_M_07P.pdf	dies#em		lr		stem	
92	91	2	[1391.67, 1511.81, 33.0, 40.0]	green	[1408.17, 1547.81]	1993_DE_LK1_01_M_07P.pdf	später				stem	
93	92	2	[688.19, 1725.0, 25.0, 38.0]	green	[700.69, 1762.0]	1993_DE_LK1_01_M_07P.pdf	Grass		l		stem	
94	93	2	[712.85, 1724.61, 22.0, 40.0]	green	[723.85, 1760.61]	1993_DE_LK1_01_M_07P.pdf	Grass				stem	
95	94	2	[1338.33, 1728.33, 39.0, 30.0]	red	[1357.83, 1761.33]	1993_DE_LK1_01_M_07P.pdf	des		l		stem	

Figure A1: Example of the CSV export that HAnnol produces. The file shown here is the same that is loaded into the tool in Figure 1, highlighting the ROI selected in that figure.

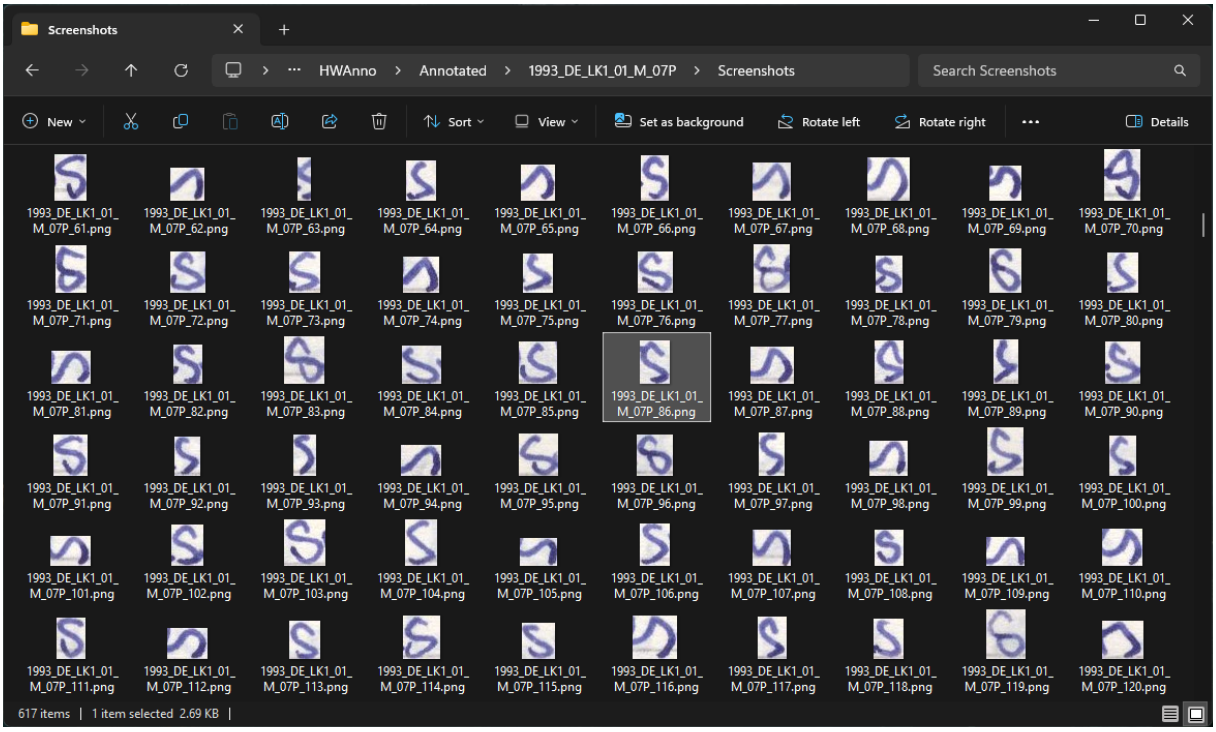


Figure A2: Folder containing PNG image files for single ROIs that have been annotated and rendered using HAnnol. The image in selection corresponds to the ROI/row selected in Figures 1 and A1, easily identifiable via the shared index (86).