Studia Neophilologica

Publication details, including instructions for authors and subscription information:

<http://www.tandfonline.com/loi/snec20>

Spotting Words in Medieval Manuscripts

Fredrik Wahlberg^a, Mats Dahllöf^a, Lasse Mårtensson^b & Anders Brun^a

^a Uppsala University, Sweden

^b University of Gävle, Sweden

Published online: 20 Jan 2014.

To cite this article: Fredrik Wahlberg, Mats Dahllöf, Lasse Mårtensson & Anders Brun (2014) Spotting Words in Medieval Manuscripts, *Studia Neophilologica*, 86:sup1, 171-186, DOI: [10.1080/00393274.2013.871975](https://doi.org/10.1080/00393274.2013.871975)

To link to this article: <http://dx.doi.org/10.1080/00393274.2013.871975>

PLEASE SCROLL DOWN FOR ARTICLE

Taylor & Francis makes every effort to ensure the accuracy of all the information (the "Content") contained in the publications on our platform. Taylor & Francis, our agents, and our licensors make no representations or warranties whatsoever as to the accuracy, completeness, or suitability for any purpose of the Content. Versions of published Taylor & Francis and Routledge Open articles and Taylor & Francis and Routledge Open Select articles posted to institutional or subject repositories or any other third-party website are without warranty from Taylor & Francis of any kind, either expressed or implied, including, but not limited to, warranties of merchantability, fitness for a particular purpose, or non-infringement. Any opinions and views expressed in this article are the opinions and views of the authors, and are not the views of or endorsed by Taylor & Francis. The accuracy of the Content should not be relied upon and should be independently verified with primary sources of information. Taylor & Francis shall not be liable for any losses, actions, claims, proceedings, demands, costs, expenses, damages, and other liabilities whatsoever or howsoever caused arising directly or indirectly in connection with, in relation to or arising out of the use of the Content.

This article may be used for research, teaching, and private study purposes. Terms & Conditions of access and use can be found at <http://www.tandfonline.com/page/terms-and-conditions>

It is essential that you check the license status of any given Open and Open Select article to confirm conditions of access and use.

Spotting Words in Medieval Manuscripts

FREDRIK WAHLBERG, MATS DAHLLÖF, LASSE MÅRTENSSON
AND ANDERS BRUN*

This article discusses the technology of handwritten text recognition (HTR) as a tool for the analysis of historical handwritten documents. We give a broad overview of this field of research, but the focus is on the use of a method called ‘word spotting’ for finding words directly and automatically in scanned images of manuscript pages. We illustrate and evaluate this method by applying it to a medieval manuscript. Word spotting uses digital image analysis to represent stretches of writing as sequences of numerical features. These are intended to capture the linguistically significant aspects of the visual shape of the writing. Two potential words can then be compared mathematically and their degree of similarity assigned a value. Our version of this method gives a false positive rate of about 30%, when the true positive rate is close to 100%, for an application where we search for very frequent short words in a 16th-Century Old Swedish cursive recentior manuscript. Word spotting would be of use e.g. to researchers who want to explore the content of manuscripts when editions or other transcriptions are unavailable.

Purpose and Introduction

The purpose of this article is to discuss the technology of handwritten text recognition (HTR) as a tool for the analysis of historical handwritten documents. We will start by giving a broad overview of this field of research, but the focus will be on the use of a method called ‘word spotting’ for finding words directly and automatically in scanned images of manuscript pages. We will illustrate and evaluate this method as applied to an Old Swedish manuscript.

Word spotting offers a path directly into the manuscripts, one that does not go through an edition. The number of unedited manuscripts, at least from the late medieval times, is large, and this tool could be of great help to get orientated in that material. Through word spotting, relevant passages in unedited manuscripts can be identified and word forms be excerpted.

Applications of this kind can be of use to linguists in many ways. The most obvious one is as a search tool, allowing us to search for words directly in manuscript images, without requiring an ordinary digital text. It will also be useful for automatic or computer-assisted transcription of manuscripts. Even if error-free

*Fredrik Wahlberg, Uppsala University, Sweden. Email: fredrik.wahlberg@it.uu.se. Mats Dahllöf, Uppsala University, Sweden. Email: mats.dahllof@lingfil.uu.se. Lasse Mårtensson, University of Gävle, Sweden. Email: lasse.martensson@hig.se. Anders Brun, Uppsala University, Sweden. Email: anders.brun@cb.uu.se.

automatic transcription is out of reach, a noisy transcription could be used for statistical analysis of word frequencies and co-occurrences of words. Furthermore, the HTR technology will support research relying on more fine-grained image data. It thus points in the direction of digital palaeography, which might address issues relating to e.g. detailed analysis of letter forms, scribe identification, and dating.

The script used in the manuscript providing data for our word spotting experiments can be characterized as *cursiva recentior* (see below). Through digital image analysis the writing can be characterized by a sequence of numerical features. Two stretches of writing can then be compared mathematically and their degree of similarity assigned a value. Word spotting uses this value to find new instances of a word that has been selected by a user. The difficulty with this is, of course, that handwritten instances of the same word are never identical. Needless to say, human readers process images and linguistic data with a unique ability to find the significant aspects among a wealth of useless information. The challenge consequently lies in finding a mathematical model of similarity that agrees with the linguistic facts (as provided by a linguist's analysis). The model must, so to speak, be designed to be sensitive to the semantically distinguishing features of the writing, while not allowing the linguistically insignificant 'noise' to influence decisions. The computational model must to that extent agree as much as possible with a competent reader's perception of which stretches are instances of a certain word. Our experiments on *cursiva recentior* will illustrate how the word spotting modelling takes the palaeographically significant features into consideration when looking for a word, and also the shortcomings of these models.

Automatic analysis of handwritten documents – challenges and possibilities

Physical and digitized linguistic data

Of course, language existed long before digital computers, and the primary traces of linguistic performance are sounds produced by speaking and images produced by handwriting. Today, the computer is the main vehicle for text access and production for many people. Modern text-based communication is consequently often digital all along the way, essentially representing text as sequences of codes identifying graphemes. This makes it easy for us, using tools like Google's search engine, to search through enormous masses of text. The old forms of linguistic performance, handwriting and speech, can be digitized as images and sounds, but these digital representations do not give us a direct representation of the linguistic units, e.g. letters and words. Rather, in order to find them, digital tools for recognizing such units must be used. Speech and handwriting reflect, as products of human behaviour, all the complexities and sensitivities of the human mind and body, and of the physical media involved. Furthermore, as sounds and images they contain far more information than that carried by the letters and words alone. Each linguistic type can, because of this, be realized in innumerable ways. Word recognition for speech or handwriting is thus a process of interpretation having to operate on very rich signals, whose significance depends on a variety of factors.

Handwriting – sources of variation

A technology for automatic recognition of letters and words in handwriting has to deal with all the sources of variation found in that mode of communication. Many aspects of variation are intentional, reflecting that writers follow certain

conventions. For instance, the craftsmanship of medieval scribes consisted in producing writing and books that met certain formalized standards. Many kinds of everyday handwriting are by contrast more informal, but they are still the products of schooling and individual choices.

The basic linguistic properties of a piece of handwriting are which language and which writing system it represents. Then there are different script types. In medieval manuscripts written in the gothic script we find types like *textualis*, *cursiva*, and *hybrida*. The choice between these types was not based on a whim, but depended to a large extent on the context in which the script was to be used. For a while, *cursiva* was confined to charters, etc., whereas *textualis* primarily was a book script. Later on, *cursiva* extended its domains and also became used in books (Derolez 2003: 128–130). All types share the basic characteristics of the gothic script originating from the gradual change of its predecessor, the Carolingian minuscule. Still, each script type represents a certain graphonomic design, with some variation regarding the typology of certain letterforms. Variants of letters might be used systematically, depending on context, as well as ligatures. This means that words cannot simply be understood as sequences of letters. The use of abbreviations further complicates the relation between writing elements and words. The abbreviation system existed in most variants of the gothic script, but generally abbreviations were more frequent and sophisticated in Latin works than in the vernacular languages (one exception being the Icelandic script).

Furthermore, a piece of writing has its own design and layout characteristics, e.g. in the way different sizes or colours are used for the letters (i.e. in rubrics, which were often written in red), or in line spacing. Another complication is that a manuscript may hold different pieces of writing, e.g. marginal notes added a long time after the main text was written. There are examples of a corrector going over the text afterwards and making changes in the text, deleting some word elements and adding others (Lindell 2000: 72–80).

There are also many sources of accidental or unintended variation. There is obviously personal variation. A scribe executes the script type in a way that is to some extent unique, at least on a micro-palaeographic level; the performance may be influenced by any kind of bodily, mental, or external circumstance. This individual variation can manifest itself in e.g. the execution of the elements constituting the letterforms, regarding minor features such as the proportions between ascenders and minims, the inclination of the ascenders and minims and so forth. Handwriting is irregular in the way any human motoric process is.

The properties of the writing material can also be a cause of further variation. A sheet of paper or parchment has to some extent an irregular texture, and the pen (sharpness, softness, etc.) also influences the shape of the writing. Different kinds of ink can also produce different results.

Orthographical variation was a rule in most medieval orthographies. The way in which words are rendered as sequences of letters (i.e. spelling) may vary. In medieval orthographies, for instance, for which no official orthographic norm existed, a scribe could choose to write the same word form in different ways. The marking of long vowels was optional, and this means that, for instance, the word *lif* ('life') in Old Swedish could be written 'liif' or 'lif' due to the scribe's preference or choice in the specific instance.

There is also variation in the morphology. Old and new word forms could exist in parallel for quite some time. This is, for instance, the case in the Old Swedish of the late Middle Ages, when the language was in a state of flux and the whole case system changed considerably. Also, a scribe could copy an exemplar written in a different dialect than his/her own, or copy older language

forms from the exemplar, which were not a part of his/her own language. The result is that some manuscripts contain a great deal of language variation, some of which were from the scribe of the manuscript (if a younger and an older form were used in parallel), some of which were older remnants from the exemplar and some due to geographic variation.

There are also more physical sources of variation facing linguists and HTR applications alike. One is that the physical objects carrying the handwriting often change over time. Physical defects are common. Parchment might have darkened, been stained, or damaged. Ink might have changed its colour, or bled through the sheets. Processes like this often lead to a loss of relevant information and make the data noisier. Another physical source of variation is that we generally work with reproductions of manuscripts. They have been photographed, printed, and digitized, and there will be losses of information along the way, e.g. as regards image resolution and the way in which colours are rendered.

Automatic analysis of handwritten documents

With a handwritten document there are many questions a researcher in the humanities or a librarian might want to know the answer to. Each of these questions might be difficult. It might even be difficult to find the right expert who would be able answer them.

- Is this a piece of handwriting at all? (Or is it a kind of print? A piece of art?)
- In which writing system (e.g. Latin, Greek, Cyrillic, Devanagari) is this written?
- In which language (language stage) is this written?
- In which script type (e.g. Carolingian minuscule, textualis, cursiva) is this written?
- Who wrote this? (Is the writer identical to or distinct from some individual we have some knowledge of, e.g. the scribe behind some other manuscript?)
- When was this written?
- What does it say? (What constitutes a correct transcription of the text, given some set of orthographic conventions?) This is a question that is typically in focus for a linguist. It also defines the purpose of HTR systems.
- The further analysis of the text may take us further into any subfield of the humanities, e.g. linguistics, history, theology, etc.

Needless to say, the linguistic questions can only be answered with reference to knowledge of the language and the writing system. As in all linguistic interpretation the process is circular: hypotheses about the minimal units (e.g. letter instances) and the complex units (e.g. words and sentences) are evaluated in parallel. A writing segment might, e.g. if we look at it in isolation, be read as 'm' or 'ni', but the further linguistic context might decide which reading is reasonable. Even if these questions in some cases are extremely difficult even for an expert, let alone for a layman, they represent the kind of interpretation that people are typically very good at. Reading handwriting is often easy, if we know the language and writing system. By contrast, this kind of interpretation is a good example of a kind of processing that is difficult for computational models. This is true for all kinds of linguistic understanding, which is characterized by a two-fold complexity: first, linguistic messages are interpreted by being connected to the sender's language, beliefs, and intentions. Second, such understanding has to deal with unpredictable and noisy signals. The sounds and visual

impressions reaching an interpreter are, in addition to a sender's communicative intentions, also a product of all the other factors influencing human behaviour and various kinds of background noise. The excellence of human perception and cognition in this regard is illustrated by reading: many kinds of everyday handwriting that are easy for people to read are impossible to transcribe using state-of-the-art HTR technology. This makes HTR an interesting research area for image analysis, computational linguistics, and linguistics. What kind of modelling of images and language is needed to solve the theoretical and technical challenges of HTR?

Handwritten text recognition (HTR)

Handwritten text recognition (HTR) can be defined as the process of automatically identifying text elements, such as letters or words, in given digital images of manuscript pages. The process of finding words in handwritten documents is obviously related to standard optical character recognition (OCR), which in its most well-known forms operates on modern printed text. To be more precise, this article deals with the *off-line* kind of HTR, which is the only one relevant for historical data. There is also *on-line* HTR, whose input is a recording of the writer's movements and the pressure of the pen tip. Here, HTR will refer to off-line HTR only. The simplest HTR problem is then to find instances of one text element, such as a letter or a word. The latter problem is known as word spotting. (The term *letter spotting* is not as common, but the task is of course a central one in HTR and OCR systems.) Identification of all text elements, together with determining their place in the textual flow, gives us transcription. This can be seen as the full-fledged HTR application. In all of these cases, the design of an HTR system involves deciding what to look for and how to do it. Documents of that kind are uniform, exhibiting little variation in the way e.g. letters appear. An HTR system, on the other hand, has to find letters and words in images where the writing is subject to considerable variation caused by a variety of factors, as discussed above. The range of variation that has to be covered depends on how versatile the system is intended to be.

When it comes to the basic units to be identified, letters, words, or letter sequences shorter than words seem to be reasonable choices. Special elements such as ligatures or scribal abbreviation signs require special care in a letter-oriented framework. The most straightforward approaches in HTR are based on image matching. The search question is defined by an image of an instance of the unit we are looking for. The technical issue is then to compare this image with subareas of the manuscript and decide how similar they are. This similarity is recorded as a number, and areas above a certain threshold value will be delivered as search hits. This means that a user might decide (using a low threshold) to look at many hits, accepting false ones, but being unwilling to allow true instances to escape recognition. Or, we might only want to look at the most certain hits (high threshold), accepting that the system does not find all instances. Of course, there are many ways in which images might be similar or dissimilar. In HTR image matching we want a comparison method that is sensitive to features of writing that are semantically distinguishing, but disregards features that are a matter of linguistically insignificant variation. The experiments reported in this article are concerned with word image matching (using a comparison method known as *dynamic time warping*).

From a linguistic point of view, a template image represents one way of writing the letters or one spelling variant of a specific word form. The image

matching method is designed to capture the important visual features of writing, but is not informed by deeper information about the linguistic structure. To make a full search for a word form one has to use images covering all spelling and allographic variants. Still, one can use this method to map the word content in an unedited manuscript, at least roughly, and this is useful for many groups of researchers, especially those interested in words and word forms (language historians, lexicographers, etc.).

A more sophisticated approach to HTR is to start with the insight that letters and words come in many different shapes, and to build a system that generalizes from richer sets of annotated data. The basic components are in that case not about directly comparing two images, but a matter of classifying them according to a model that has been produced by machine learning from the many given examples (annotated images). These classifier components might target letters, other subword units, or whole words. For instance, Fischer et al. (2012) propose a letter-based but context-sensitive system intended for word spotting with arbitrary search words.

The basic building blocks of an HTR system are image-oriented in the sense that the information they use is what can be found in the images of the written shapes. As the spotting of textual elements is a matter of similarity measurements, there will be a high degree of uncertainty. Some segments will be ambiguous, others impossible to recognize at all. As the image-based components of an HTR system will typically deliver alternative hypotheses, to each of which some probability is attached, a model of linguistic probabilities will help the HTR system deliver better proposals. Given a corpus covering a language it is possible to compute the probabilities of the different letters or different words given their context. In this way it is possible to design HTR applications that take both image data and linguistic structure into consideration.

As we see, an HTR system is based on data from known manuscripts and languages. The intended use of a system is a crucial question. State-of-the-art research typically involves one manuscript at a time. The HTR analysis builds on data taken from one part of the manuscript and is tested on other parts. The whole process is concerned with one writer and one language. HTR components of this kind could be used to explore the text of a manuscript and form the core of a system for computer-aided or partially automatic transcription.

HTR as a component in a system for digital palaeography

To be useful for linguists, an HTR component must be part of a set of tools supporting systematic procedures for analysing manuscripts in a digital framework. First, there is the issue of how the physical manuscripts are digitized. As far as possible, this should be done in a standardized way, ensuring that the images are of a high and documented quality. The handling of metadata should also be integrated in the digital workflow. Then there is the issue of how to represent the results of the linguist's analysis, e.g. a transcription or a deeper linguistic analysis. Again, an important advantage of using digital tools is that they support standardized representations (e.g. as defined by the Text Encoding Initiative, www.tei-c.org) and systematic practices. An HTR component can be used as a tool providing a very raw transcription in a situation where a linguist is responsible for a finalized full transcription. It can also be used as a search tool in a research procedure where the text is not transcribed. In both cases, the analyses provided by a researcher can be associated with specific and precise locations in the digitized manuscripts. This makes it possible for anyone (given that the data

are made public) to trace the interpretive processes back to the very words they concern and in the form in which they have been handed down to us.

An HTR component typically relies on certain preprocessing steps, as we will see in the section detailing a word spotting experiment. The digital master image produced at a library is often a full spread. The pages then have to be separated. This can normally not be done by just cutting the image in half but requires some more processing to identify the actual blocks of text. The main blocks have to be separated from illustrations, margin notes, illuminated characters, etc. Many image matching methods used in HTR require the text lines to be separated. This is in itself a far from trivial problem: overlapping between lines, notes, and help lines often obscures the line structure.

Experiment: Word spotting applied to an Old Swedish manuscript

In this section we report the outcome of applying the method of word spotting through image matching to an Old Swedish manuscript. In this experiment, we compare results for two letter sequences that are similar from a graphic perspective, but represent different language units. In this way we evaluate the word spotting method in a specific linguistic and palaeographic context.

Old Swedish manuscripts

The complete corpus of Old Swedish manuscripts consists of approximately 270 books (Åström 1993: 230). Some of these contain only a few leaves, whereas other books consist of what were originally different works that were bound together at a later stage. Great effort has been put into editing Old Swedish texts, although not to the same extent as in the case of the West Norse ones, especially those in Old Icelandic. The society *Svenska Fornskriftsällskapet* (The Swedish Society for Old Texts), whose main goal was (and still is) to edit medieval texts originating from Sweden, was founded in 1843, and from then on, a large number of editions of Old Swedish texts have emerged.

Despite the long and periodically rather intensive activity in the production of editions, the study of Old Swedish medieval manuscripts faces large infrastructural challenges. Although a great number of editions have been published, a large portion of the manuscripts are still unedited, which means that a great deal of linguistic data has not been accounted for in the study of the Old Swedish language. In most editions, the focus is on editing a *text*, not a *manuscript*, and the manuscript that is regarded as the best from the perspective of textual criticism forms the base, and other manuscripts containing the same text are accounted for in the variant apparatus. So-called synthetic editions, common in editions of Latin texts, are rare among the Old Swedish editions. Only the main manuscript can be said to be fully available for the reader of the edition, as the variant apparatus usually contains only variants that are of significance from the perspective of textual criticism. (And not all users of the edition are studying textual criticism.) Another major obstacle for the study of the Old Swedish manuscripts is the absence of a catalogue, which makes it difficult to get a good picture of the construction and content of the manuscripts in their entirety. Thus, despite many years of research, the Old Swedish manuscripts are not nearly as well mapped as for instance their West Norse counterparts. Consequently, there is a wealth of Old Swedish material in need of philological

investigation. This is an area in which the use of HTR technology would support a systematic empirical approach.

The manuscript Codex Upsaliensis C 61

The present study has been conducted on the manuscript Codex Upsaliensis C 61, preserved at the University Library in Uppsala. This small quarto manuscript is dated to the beginning of the 16th century. Today, it consists of 565 leaves, divided into 70 gatherings. It comprises a number of the Revelations of St Bridget of Vadstena in Old Swedish translation, but also other material concerning this saint. It was produced in the monastery of Vadstena, but it left the monastery at an early stage, and as early as 1536 it was to be found in Stockholm (Morris 1991: 4–7).

The codex is the work of three different scribes. The hand that wrote pp. 539–1104 provides the data for the present study. Morris (1991: 9) describes the script as being of a typical Vadstena type, but she also claims that it is very distinctive. According to Morris, the script differs from other Vadstena manuscripts in that it is less angular and more curved, and that the letters are rather spacious. Other researchers, for instance G.E. Klemming (1883–1884: 156), have described the script in negative terms, calling it awkward.

As has been stressed in later research (Åström 2010: 121; Dverstorp 2010: 154), the script used in Vadstena can be seen as a part of the general development of the gothic script in northern Europe. The script in C 61, admittedly having some clearly distinct traits, as pointed out by Morris, carries the characteristics of the script type *cursiva recentior*, which was used in large parts of medieval Europe. Albert Derolez (2003: 142) gives the following criteria for this type:

- (1) Single-compartment 'a'
- (2) 'b', 'h', 'k', 'l' with loops to the right of the ascenders
- (3) 'f' and straight 's' extending below the baseline

The script in C 61 meets all these criteria, as can be seen in Figure 1.

Regarding the level of execution, the script can be categorized as *currens*, or perhaps *libraria*. C 61 is by no means a *de luxe* manuscript, as the scribe seems to have produced the script rapidly. From the perspective of image analysis, this means that the variation in the script is quite great. The template chosen for the

Figure 1. Examples of the letters 'a', 'h', 'k', 'f', and 's'.

word spotting process can thus be expected to deviate considerably from the other instances of the same letter sequence occurring at other places.

The image matching method

Search by image matching consists in finding parts of a larger image that are similar to a given template image. In our case the larger image is a manuscript page and the template an image representing a stretch of writing. The method tried here presupposes that the manuscript image is cut up in text lines. This means that the search can be performed in a linear fashion, along the baselines of the text, eliminating the need to proceed with a computationally more expensive two-dimensional search. (However, the two-dimensional search procedure could be useful for manuscripts where text lines are difficult to discern.) Line segmentation (i.e. the process of separating an image into segments, each containing a text line) is in itself a challenging problem for image analysis as applied to handwritten manuscripts. A variety of techniques can be found in the literature, addressing different types of degradation problems and scripts. We have separated the lines using the graph optimization method from Wahlberg and Brun (2012), seen in Figure 2. We should also add that no further segmentation is necessary: the matching method does not need to be given any information about letter and word boundaries. This is a good thing, since separation of linguistic units like letters and words is yet another difficult task.

There are a number of ways to make the comparison between a template and a text line found in the literature. Among the more common ones is the use of 'sliding window features'. When using this method the images are not compared directly. Rather, a number of numerical features are extracted from them, and they provide the data structures that are more directly compared. In an optimal design of this method, these features should reflect all aspects of the writing that are significant to linguistic recognition. The advantage of this method lies in the fact that they only partially represent the information of the images, making the

Figure 2. Example of text line separation from the automatic line segmentation algorithm. The text is from C 61 blended with coloured regions signifying the area of each line.

comparison process computationally more efficient. A number of different ways of extracting these features have been proposed. We have used a set of features that have proved successful in earlier empirical studies like Rath & Manmatha (2003a) and Fischer et al. (2012).

The feature modelling is based on a splitting of the text line image into short segments. We use the shortest segments defined by the image representation, i.e. pixel-wide ones. Each feature associates each pixel column with a number capturing some information present in the text line at that column. We have here used the following features:

- F1–F3: The total number of foreground pixels (pixels ‘filled’ with ink) in each pixel column, called the projection profile. In addition, two partial projection profiles are computed in the column above and below an estimated middle line.
- F4, F5: The upper and the lower contour height of the text. These will be used in the following examples, shown in Figures 3 and 5, since they are easily illustrated.
- F6, F7: The derivative, i.e. the directional tendencies, of the upper and lower contour.
- F8, F9: The average of the two statistical measures (weight centre) and variance of the foreground pixels.
- F10: The number of transitions between foreground and background.
- F11: The fraction of the pixel column between the upper and lower contours belonging to the foreground.

Given that we work with several features simultaneously, each pixel column along the text line is associated with a sequence of numerical values – in mathematical terminology, a *feature vector*. The problem of finding word images is then reformulated as a problem of finding sequences of feature vectors that are similar. This way of doing the search on features instead of using the actual image data directly might seem crude, but has proved both fast and accurate for word spotting. Ideally, the features should be invariant to noise and irrelevant aspects of the text line and yet carry enough information to discriminate between different word graphs.

This procedure is based, firstly, on column-wise comparison of the feature vectors, and secondly, on comparing sequences of such feature vectors. For instance, the columns in the middle of an ‘i’ (presupposing that it is a perfectly vertical one) will have a high projection profile weight (capturing the vertical amassment of ink). Two such columns should be similar, whereas each of them should be different from a column in the middle of an ‘o’ or in a space. As letter sequences or components of letters cannot be expected to have exactly the same widths, the second aspect of the comparison, where sequences are compared, cannot simply proceed column by column at the same pace for the template and the manuscript image. Rather, it must allow a narrow letter instance to correspond to a wider instance of the same letter and vice versa. For instance, the matching procedure should allow a 3-pixel-wide ‘i’ followed by a 6-pixel-wide space to be aligned with similar sequence with 4 ‘black’ columns followed by 9 ‘blank’ columns. We have used a method for this called dynamic time warping (DTW) as explained in Rath & Manmatha (2003b). The point of the metaphor behind this term can be explained by looking at reading as a process proceeding one column at a time. Some letters are then quick (narrow) and others are slow (wide).

The process of feature extraction, matching, and the associated (dis)similarity function is illustrated in Figures 3, 4, and 5. In this example, two instances of the

Figure 3. Two instances of the word 'och' ('and') with two feature curves. The superimposed curves are the upper and lower contour features (F4, F5). These curves are, among a number of other feature curves, the only data used when doing word matching. Note that the lower contours of the ascenders of the last letters in both images are not fully captured due to ink degradation.

Figure 4. Image of a text line (upper) with the corresponding weight matrix (lower). The line through the weight matrix shows a warping path when looking for the word 'och' in the text line. One instance of the word sought for is found, i.e. the best match given the template. If the match is good enough the remaining part of the text line is searched again to find additional instances.

Figure 5. Three images illustrating the matching process. Left: The upper and lower contours from the images of the word 'och' ('and') from Figure 3. Middle: The difference between the curves is filled in. The area of the difference can be used as the dissimilarity measure. Right: The curves, with the difference filled in, after dynamic time warping has been applied to 'stretch' the curves for a better match. (Note that the difference area of the rightmost image is much smaller than that of the middle image.)

word ‘och’ (‘and’) are matched. The features shown in this simple example do not capture all relevant aspects of the letter shape. Using only the upper and lower contours (F4, F5) will make the letters ‘c’ and ‘e’ look the same, from the computer’s point of view. For this reason, more features (the eleven described above, F1–F11) have been used for the experiments below.

When searching for the template selected by the user, a measure of similarity or dissimilarity between sequences of feature vectors has to be constructed. Here it is roughly the area between the curves that is summed up and used as a measure of dissimilarity. The dissimilarity between two pixel columns is defined as the square Euclidean distance (Equation 1) between the two feature vectors associated with the columns. All differences can then be summed to create a global dissimilarity measure.

$$d^2(\vec{x}, \vec{y}) = \sum_{i=1}^N (x_i - y_i)^2 \quad (1)$$

The numerical features are normalized by subtracting the mean value μ (Equation 3) from each element and then dividing by the standard deviation σ (Equation 2) of the full sequence of features in the document (Wahlberg et al. 2011).

$$\sigma(\vec{x}) = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu(\vec{x}))^2} \quad (2)$$

$$\mu(\vec{x}) = \frac{1}{N} \sum_{i=1}^N x_i \quad (3)$$

Finding the optimal warping with respect to the dissimilarity measure can be guaranteed using a technique from computer science called *dynamic programming* (Cormen et al. 2001). It efficiently finds the best alignment of the template word feature vectors to the text line feature vectors, expressed as the optimal warping of the template word to fit the text line. Using the dissimilarity function $d^2(i, j)$, where i and j are associated with pixel column numbers in the text line image and the template image (and their respective feature vectors), the warping matrix D is computed (Equation 4).

$$D(i, j) = \min \left\{ \begin{array}{l} D(i-1, j) \\ D(i, j-1) \\ D(i-1, j-1) \end{array} \right\} + d^2(\text{template}(i), \text{textline}(j)) \quad (4)$$

This strategy yields a rectangular matrix with the dimensions equal to the lengths of the template and text line (as shown in Figure 4). The next step is finding a path through this matrix from the upper to the lower edge, where the accumulated cost of each element passed should be minimal. Given the definition in Equation 4 this can be done by ‘backtracking’ from the lower edge to the upper. It can be shown that the given path represents the warping with the minimal dissimilarity cost.

In the rightmost image in Figure 5 the curves have been warped, and the difference area is much smaller than in the middle image. Note that the

warping could have been more exact if the upper and lower contours were considered separately. When using the full set of 11 features, warping each separately raises the computational complexity significantly, and it is unknown if it would yield desirable improvements. In the warped image the area between the curves is considerably smaller, and this is used as the DTW-improved dissimilarity measure. An important difference between the example and the real implementation is that the dissimilarity between two feature vectors is squared, punishing larger differences significantly more than smaller differences.

The matches generated by a search will fit very well with the template in the feature domain, that is, the feature vectors match well with respect to the dissimilarity measure. The word matching might, however, not be as good. Some calibration is almost always necessary. Our search software was implemented using Matlab with extensions in C. A full search for one word takes about a minute on a 2.3GHz processor.

Searching for 'och' and 'ath' in C 61

In this study, we have performed searches for two separate graph words, namely 'och' and 'ath' in 20 pages of C 61. These contained a total of 159 and 80 instances of 'och' and 'ath', respectively. The total number of words is approximately 3,000.

The first one, 'och', is in C 61 the most common spelling variant for the word form *ok* ('and'). Other spelling variants exist, also on these specific pages of C 61, e.g. 'ok', which is often used at the end of lines, probably because it is a shorter sequence and thus fits better into the end of the line when the space is limited. The graph word 'ath' may represent several lemmas, i.e. the infinitive marker, a subordinate conjunction, and a preposition.

The two words provide good targets for a search experiment, as they are very frequent. The contours of different instances of the two words are not extremely divergent. However, their similarity is a challenge to the present method, as it increases the probability that the two graph words will be confused. Both sequences begin with two low letters that have a similar form (see further below) and end with an 'h'. The letter 'a' in this script type is of a single-compartment type, consisting of an oval circle, placed on the baseline, and a stroke down to the right side of this. The last stroke is the only element that separates 'a' from 'o' in this style, and in fact it only marginally affects the contours of the two letters, especially if they are placed in a letter sequence, and another letter is attached to it on the right side. To some extent, the oval circle in 'a' slopes slightly more than in 'o', but only marginally, and this can furthermore be very different within different examples of 'a' and 'o'.

The letters 't' and 'c' are also often similar here, and in cursiva generally (Derolez 2003: 144). They have a very similar ductus, as both are executed in two strokes, one horizontal and one vertical, which at the baseline turns into a hook to the right. The trait distinguishing 't' from 'c' is rather the lengthening of the vertical stroke over the headline; in 't' this line crosses the horizontal one, whereas in 'c' it stops at this place. In practice this is not consistently carried out, as 't' sometimes lacks the lengthening and 'c' sometimes has it. These borderline cases often make it difficult to tell 'c' and 't' apart when seen in isolation.

Results, discussion, and conclusions

The performance of the word spotting method

Our word spotting method is able to find instances of selected words with an accuracy far above the chance-level baseline. Two important evaluation performance measures for a search are the true positive rate (TPR) and the false positive rate (FPR). TPR is the proportion of (correct) hits among actual instances of the target, e.g. the number of ‘och’ instances found divided by the actual number of instances. FPR is the proportion of (incorrect) hits among actual non-instances of the target type, e.g. the number of incorrect instances proposed divided by the actual number of non-instances. The two measures are plotted in so-called ROC curves in Figure 6. By choosing the threshold for registering a hit, we can decide whether to avoid false positives, paying the price in a loss of true positives, or to avoid false negatives, having to deal with an increasing number of false positives. A perfect procedure would produce TPR 100% and FPR 0%, whereas one giving random verdicts (i.e. being completely incompetent, so to speak) would perform in a way that makes the two measures identical, i.e. an example and a non-example would be as likely to be produced as search hits. This corresponds to the diagonals in the ROC curves (and different points there to different thresholds). The word spotting curves are in between the perfection of the upper left axis and the incompetence of the diagonal. We see that the word spotting generates a certain amount of false hits (false positives) and misses instances in the manuscripts (false negatives). From the curves we can conclude that if we accept a false positive rate of 30%, we will find all of the true positives.

However, we do not imagine that the word spotting method should be used in isolation as in the experiments reported here. Rather, in a context where the system is set up to search for a considerable number of words (or other linguistic units), it can benefit from using information about the probability of different linguistic types in different context, as estimated by a language model generated from a corpus representing the language of the document. Furthermore, if we search for a large number of words we can hope that many potentially false positives for the other words become true positives for the right word. We may also adopt a manual or semi-automatic post-classification of the results, to boost accuracy to a level that is needed for a given application. We know today from Internet technology that searching is an interactive process, where the user

Figure 6. ROC curves for ‘och’ and ‘ath’.

quickly refines the query to steer the result. Manual correction is also feasible when we search for moderately rare words, where false positives can quickly be spotted by an expert.

An interesting question, which is in need of further investigation, is in what way the script type influences the performance of word spotting by image matching. At first glance, one might expect that textualis, to take an example that is important from the point of view of medieval palaeography, would prove easier to handle for our word spotting method than the cursiva script that we have studied here. Textualis is typically used as a more carefully executed book script, while cursiva often represents a less tidy charter script. The difference between textualis and cursiva mainly manifests itself in whether certain redundant features are present or not. The most prominent of these are the loops on the letterforms with ascenders and the extension under the baseline on straight 's' and 'f'. A third feature is the morphology of 'a', as textualis normally has a two-compartment 'a' and cursiva a one-compartment form (Derolez 2003: 142). Cursiva also has some other redundant features, for instance frequent extensions on the final minim in such letters as 'm' and 'h' (Derolez 2003: 147, 148). Because of the features used, which include upper and lower contours, the image matching process used in this study will be sensitive to components like ascenders and lengthenings of the minims, as in the case of cursiva 'h'. This is reflected in the false hits that the matching gave: most of these sequences contained an 'h' as the final letter, whereas the initial two of them could vary, as long as they were 'low' letters (containing no ascender). The lengthenings of the minims in e.g. cursiva 'h' provide a distinct lower contour, and also contribute to the projection (vertical amount of ink), which are among the features we use for matching. Structurally redundant elements can therefore be of considerable value for the model, and as these are more prominent in a style like cursiva than in textualis, we might conjecture that word spotting by means of our image matching method performs better on cursiva manuscripts. On the other hand, the often faster execution behind cursiva makes the variation between the individual letters, the graphs, greater.

Prospects for future work in HTR

Basic image matching can only provide a partial solution to the problem of word spotting, as the template image only represents one graph word, in one spelling, and in one kind of writing style. A more general system for word spotting must be able to deal with arbitrary search words. Of course, it would be possible in principle to extract template images for all graph words manually in advance, but this would be a time-consuming process. In order not to rely on such a procedure, the system must be able to handle queries by performing searches based on subword units, e.g. letters or letter sequences. Perhaps we also want a system that is versatile in the sense of being able to perform searches for a variety of manuscript types. This desideratum is, however, beyond the reach of the current state of the art in HTR, but it points in the direction of methods based on modern machine learning and richer data sets. A possible architecture for doing this would be to first identify, through an analysis of the manuscript as a whole, which script type is involved, and then to apply a search procedure taking the script type into consideration. Such an approach could also be used to construct the query in word spotting from a typed sequence of characters rather than a word graph.

Research in HTR is at present in a state of exploring basic solutions and experimenting with different data. Data mining of manuscripts and collections is

an open field of research, involving everything from well-preserved and carefully printed to damaged and hastily written texts, where the technical challenges range from tractable to nearly impossible. However, we do believe HTR components integrated in software tools for automatic or computer-aided transcription and analysis will support deeper and more systematic empirical research concerning handwritten documents.

REFERENCES

Manuscript

Cod. Ups. C 61, University Library in Uppsala.

Printed Sources

- Åström, Patrik. 1993. Källorna till den fornsvenska språkhistorien: en översikt över det bevarade bokbeståndet. *Studier i svensk språkhistoria* 3: 229–243.
- Åström, Patrik. 2010. Myten om Vadstenastilen. *Arkiv för nordisk filologi* 125: 99–126.
- Cormen, Thomas H., Clifford Stein, Ronald L. Rivest & Charles E. Leiserson. 2001. *Introduction to algorithms*. 2nd edn. New York: McGraw-Hill Higher Education.
- Derolez, Albert. 2003. *The palaeography of Gothic manuscript books. From the twelfth to the early sixteenth century*. Cambridge Studies in Palaeography and Codicology 9. Cambridge: Cambridge University Press.
- Dverstorp, Nils. 2010. Skrivaren och skriften. Om skrift- och handskriftsproduktion i Vadstena kloster. Ph.D. dissertation, University of Oslo.
- Fischer, Andreas, Andreas Keller, Volkmar Frinken & Horst Bunke. 2012. Lexicon-free handwritten word spotting using character HMMs. *Pattern Recognition Letters* 33(7), 934–942.
- Klemming, G.E. (ed.). 1883–1884. *Heliga Birgittas uppenbarelser*. Samlingar utgivna av Svenska Fornskriftsällskapet H1.14:5. Stockholm: Svenska fornskriftsällskapet.
- Lindell, Inger (ed.). 2000. *Heliga Birgittas uppenbarelser bok 7 efter Cod. Ups. C 61. Diplomatisk utgåva med kommenterande inledning*. Samlingar utgivna av Svenska fornskriftsällskapet H1.84. Uppsala: Svenska fornskriftsällskapet.
- Morris, Bridget (ed.). 1991. *Book V of St Birgitta's Uppenbarelser edited from Ms Cod. Ups. C 61*. Samlingar utgivna av Svenska fornskriftsällskapet H1.80. Stockholm: Svenska fornskriftsällskapet.
- Rath, Toni M. & R. Manmatha. 2003a. Features for word spotting in historical manuscripts. *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, vol. 1: 218–222.
- Rath, Toni M. & R. Manmatha. 2003b. Word image matching using dynamic time warping. *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, vol. 2: 521–527.
- Wahlberg, Fredrik, Mats Dahllöf, Lasse Mårtensson & Anders Brun. 2011. Data mining medieval documents by word spotting. *Proceedings of the 2011 Workshop on Historical Document Imaging and Processing (HIP '11)*, ACM, New York: 75–82.
- Wahlberg, Fredrik & Anders Brun. 2012. Graph based line segmentation on cluttered handwritten manuscripts. *Proceedings of the 21st International Conference on Pattern Recognition, (ICPR)*, 11–15 Nov. 2012, 1570–1573.